A Comprehensive Framework for Rendering Layered Materials

Wenzel Jakob
ETH Zürich

Eugene d'Eon
Weta Digital

Otto Jakob
Atelier Otto Jakob

Steve Marschner
Cornell University

Figure 1: *All materials in this interior scene were generated and rendered using the techniques described in this paper. The insets on the left and right reveal the corresponding structural descriptions that were used as inputs to our system.*

Abstract

We present a general and practical method for computing BSDFs of layered materials. Its ingredients are transport-theoretical models of isotropic or anisotropic scattering layers and smooth or rough boundaries of conductors and dielectrics. Following expansion into a directional basis that supports arbitrary composition, we are able to efficiently and accurately synthesize BSDFs for a great variety of layered structures.

Reflectance models created by our system correctly account for multiple scattering within and between layers, and in the context of a rendering system they are efficient to evaluate and support texturing and exact importance sampling. Although our approach essentially involves tabulating reflectance functions in a Fourier basis, the generated models are compact to store due to the inherent sparsity of our representation, and are accurate even for narrowly peaked functions. While methods for rendering general layered surfaces have been investigated in the past, ours is the first system that supports arbitrary layer structures while remaining both efficient and accurate.

We validate our model by comparing to measurements of real-world examples of layered materials, and we demonstrate an interactive visual design tool that enables easy exploration of the space of layered materials. We provide a fully practical, high-performance implementation in an open-source rendering system.

CR Categories: I.3.3 [Computer Graphics]: Three-Dimensional Graphics and Realism—Color, shading, shadowing, and texture

Keywords: layered materials, linear transport theory, albedo problem, importance sampling, BRDF, BSDF

Links: DL PDF WEB VIDEO CODE

1 Introduction

The notion of scattering at a surface is central to rendering. It is appropriate, and computationally essential, to treat thin structures as surfaces when the scale of the scene allows: from a metal surface with microscopic imperfections to a rough ocean viewed from space, surface-scattering models are often the correct representation.

Metal or dielectric interfaces with microscopic roughness are the simplest scattering surfaces, but the vast majority of nonmetallic materials are not transparent; they reflect light diffusely by subsurface scattering. The most commonly used reflectance models, with a diffuse and a specular component, are based on the idea of a dielectric boundary above an optically dense scattering medium.

Surfaces can also be more complex, with layers that might not fully hide the material below: glaze over ceramic, wall paint over primer, colored car paint with a clear coat, vitreous enamel in gold and silver jewelry, and layered biological structures like leaves, flower petals, or skin. All these can be described in terms of a stack of layers of scattering and/or absorbing media, separated by interfaces that might be smooth or rough. At the bottom the stack could be an opaque interface (such as a metal) or a transparent one. Layers and interfaces provide a language that is useful for describing a wide range of surfaces, and which already underlies most BRDF models.

Expressing a surface explicitly as a layered system with physical parameters also allows it to be treated consistently across scales. For close views where a given structure can't be treated as a thin surface, the structure can be rendered directly, and with an accurate

reduction to a surface model, it can be freely switched to a surface for farther views.

However, it is important to realize that we do not have accurate computational models for scattering from any systems more complex than a single layer or a single interface. Even the simplest nontrivial system, of a single medium bounded by a smooth interface, is only roughly approximated by standard BRDF models. Many other models have been proposed for layered or coated structures of one kind or another, but all are limited to particular special cases or lacking in terms of accuracy. Of course, it is possible to instantiate any kind of layered structure explicitly in a rendering system, relying on a general purpose rendering algorithm to resolve interactions between the layers; however, this approach is usually impractical due to the difficulty that a such methods face in reliably finding light paths through a material stack.

In this paper we provide a complete solution for accurately simulating scattering from layered surfaces. Our system handles any isotropic surface that can be expressed in terms of layers and interfaces; in a sense we provide a computational language for describing surface structure. The building blocks of our system—the grammar of the language, so to speak—are basis expansions, the adding equations, and the adding/doubling method: computational tools from radiative transport that have seen occasional use before in graphics. In this paper we develop the vocabulary needed to express the kinds of surfaces that are useful for graphics: we show how to use microfacet models for transmission and reflection, measured BRDFs, and media with the Henyey-Greenstein and von Mises-Fisher phase functions in this framework. We also provide efficient algorithms for evaluation and exact importance sampling at render time.

Our system is expressive but also practical. A precomputation step expands all the BSDFs and phase functions in a Fourier basis, which causes the plane-parallel multiple scattering problem to decompose into separate problems per azimuthal frequency. The multiple scattering problem is solved using adding/doubling for scattering layers and the adding equations for the sequence of layers and interfaces. Finally the resulting BSDF is stored to be used for rendering.

Our implementation scales efficiently to high orders, handling very rough to nearly mirrorlike interfaces, and isotropic to strongly peaked phase functions, while maintaining accuracy and avoiding ringing or other artifacts. The precomputed BSDF is stored in a sparse format, requiring a few kilobytes to a few megabytes, depending on the expansion order. Precomputation time is on the order of seconds, and render-time performance is close to that of dedicated implementations of BSDF models—in fact, the quality of our importance sampling technique means that precomputed representations of standard BSDF models sometimes outperform the corresponding analytic models in equal-time comparisons.

Achieving this level of accuracy and performance requires considerable care in designing representations, algorithms, and numerical techniques used in both precomputation and rendering. In this paper we discuss the design of the system and the most important issues in making it work, and the full details are available in the supplementary report. We will release a complete open-source implementation as part of the Mitsuba renderer [Jakob 2010].

This paper has the following contributions:

- We develop efficient and robust numerical methods for projecting a range of transport-theoretic reflectance models into a basis that supports arbitrary composition.
- We recognize the inherently sparse nature of the problem to be solved and develop a system that is able to exploit this property to scale to a wide range of layered structures including very challenging input. To our knowledge it constitutes the first practical solution for modeling the interactions of multiple rough boundaries without simplifying assumptions.
- We show how to efficiently evaluate and importance sample the resulting tabulations and propose an extrapolation method that accelerates precomputation and rendering times further when some approximations are acceptable.
- We propose a multiple scattering term for microfacet BRDFs that avoids an energy loss issue of these models.
- We develop heuristics that automatically select a basis of sufficient resolution to guarantee bounds on representation error.

2 Prior work

Light transport in graphics is typically done in the framework of linear transport theory (wave and quantum effects are neglected except, possibly, in deriving scattering phase functions). Predicting reflectance from layered materials at this level of accuracy is the subject of the *albedo problem* [van de Hulst 1980] in one-dimensional linear transport theory—the problem of determining the emergent distributions from plane-parallel media subject to only external sources.

Exact closed-form albedo problem solutions are rare. In graphics, Blinn [1982] used single-scattering solutions, which were extended by Hanrahan and Krueger [1993] to include index of refraction changes, multiple layers, and a Monte Carlo method for computing multiple scattering. These were some of the first graphics models to work in terms of layered media, but single-scattering solutions have limited applicability and the more general Monte Carlo method is unfortunately slow to converge—though the convergence rate can be improved in some cases by the use of scattering equations in integral form [Pharr and Hanrahan 2000]. Our work pursues the same goal with a focus on efficiency, accuracy, and generality.

Exact solutions that include all orders of scattering are limited to infinitely-thick materials with isotropic or linearly-anisotropic scattering and smooth, indexed-matched boundaries [Chandrasekhar 1960]. The semi-infinite solution for isotropic scattering involving the H-function was used by Premože [2002] to render dusty surfaces. Solutions for finite layers [Das 2010] or for semi-infinite media with Fresnel reflection [Williams 2006] require solving integral equations by iteration or other methods.

A wide variety of approximate deterministic methods are available for computing BSDFs for general layered materials, including discrete ordinates (S_N) [Thomas and Stamnes 2002], the transfer matrix method [Aronson 1971], the FN method [Siewert 1978], adding-doubling [van de Hulst 1980], analytical discrete ordinates [Siewert 2000], and the singular eigenfunction method [McCormick and Kuscer 1973]. These methods all have different accuracy/cost tradeoffs and stability issues [Chalhoub et al. 2003] and in principle any of them could be used to compute transport operators within our general framework.

In graphics, our approach is most similar to that of Stam [2001]. In his paper, he proposes the first BSDF model for rough dielectrics and uses it via the S_N method to compute a BSDF that models skin as an anisotropic scattering layer with rough dielectric boundaries. Our work continues in the same vein as Stam's: we also model surfaces in terms of layers and boundaries and we work in terms of a similar directional basis. However, our system goes considerably beyond Stam's earlier work, providing a complete, modular solution that can handle any layered structure accurately and efficiently and maintains good performance for high order expansions. This

requires fundamentally different ways of computing basis expansions, representing them compactly, and solving the resulting equations robustly in the face of challenging input. Also, to the best of our knowledge, we are the first to consider importance sampling, von Mises–Fisher scattering, and an approximate multiple surface-scattering correction for rough interfaces using this approach.

Several specialized BRDF models have been proposed that approximate specific types of layered structures using coupled diffuse and specular lobes [Wolff et al. 1998; Shirley et al. 1997; Kelemen and Szirmay-Kalos 2001], including asperity scattering [Koenderink and Pont 2003], and modified microfacet distributions for rough slabs [Dai et al. 2009]. Our framework subsumes these models and is generally also more accurate as a consequence of correctly accounting for all interactions between layers. A variety of other approximate analytic methods have been proposed for rendering layered materials, such as Kubelka-Munk [Dorsey and Hanrahan 1996], diffusion-based methods [Donner and Jensen 2005], or specialized approximations for leaves [Wang et al. 2005]. A flexible family of layered BRDFs was proposed by Weidlich and Wilkie [2007]. Their approach is simple and efficient but doesn't achieve many desirable properties that our model does, including reciprocity, energy conservation, and accurate accounting for multiple scattering within and among layers.

Neglecting polarization effects, we consider only scalar radiative transfer in this paper. Extension of our methods to handle vector radiative transfer is straightforward [Garcia 2012]. There is also the possibility of inhomogeneous scattering layers—layers where the single-scattering albedo is a function of depth, which are solvable using known methods [Yanovitskij 1997a]. Flourescence [Wilkie et al. 2006], anomalous dispersion [Weidlich and Wilkie 2009], and thin film interference effects [Hirayama et al. 2001; Icart and Arquès 2000; Ershov et al. 2001] have been considered in reflectance models and could be included as components in our framework but our foremost focus here is on a comprehensive geometrical optics framework.

3 Background

This section describes the computational framework of our system, which builds upon a number of prior works in transport theory that we review here. We will work under a plane-parallel, or 1D transport, assumption, meaning that illumination and surface properties are assumed to be invariant across the surface, so that all functions depending on position can be modeled as depending on depth alone. Practically speaking, this means that the we are deriving a BSDF rather than a BSSRDF model.

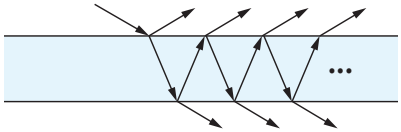
The system works in terms of radiance functions expanded in a directional basis, so that the BSDFs of layers and interfaces are represented by matrices known as *scattering matrices*.

We begin by introducing the adding equations, which are used to compute the scattering matrix of a composite layer given the scattering matrices of its constituent layers. Next, we describe the directional basis that underlies these matrix representations, and we show how to use it to discretize the radiative transfer equation and boundary conditions. Finally, we describe how to solve the radiative transfer equation using the adding-doubling method, which is named in this way due to its reliance on the adding equations to repeatedly double a layer until it has the necessary size.

3.1 Adding equations

Some of the earliest theoretical work on layered materials was conducted by Stokes [1860], who analyzed the combined reflection and transmission properties of a stack of glass plates. It will be instructive to review the mathematics underlying the simplest case of his analysis involving only a single plate.

The top interface of a glass plate illuminated by a ray of unit power reflects a portion R of the light and transmits another portion T into the material, where it goes on to encounter the bottom interface, reflecting back and forth with a fraction escaping at each event:



Due to reciprocity, the reflection and transmission coefficients at the bottom interface are also equal to R and T. By summing over all light paths, it is then possible to compute the total reflectance and transmittance of the plate as a whole. The resulting geometric series have a very simple explicit form:

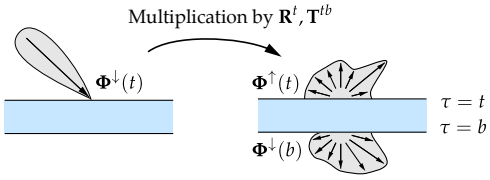
$$\begin{aligned} \tilde{R} &= R + TRT + \ \ldots \quad &&= R + \frac{RT^2}{1-R^2}, \\ \tilde{T} &= TT + TR^2T + \ \ldots \ &&= \frac{T^2}{1-R^2}. \end{aligned} \tag{1}$$

These equations show us how to compute the scalar reflectance and transmittance of the two interfaces together from the reflectance and transmittance of the two separate interfaces.

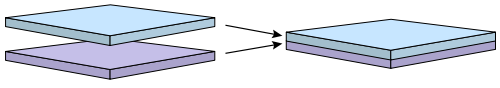
With this example in mind, let us move to a more general case: Rather than a smooth interface illuminated from a single direction, consider a slab of arbitrary composition illuminated by a radiance distribution expressed in some basis. (We leave the underlying discretization unspecified for now.) The linearity of light transport then allows us to write the scattered illumination projected into the same basis using a matrix-vector product:

$$\begin{aligned} \mathbf{\Phi}^{\uparrow}(t) &= \mathbf{R}^t \mathbf{\Phi}^{\downarrow}(t) + \mathbf{T}^{bt} \mathbf{\Phi}^{\uparrow}(b), \\ \mathbf{\Phi}^{\downarrow}(b) &= \mathbf{R}^b \mathbf{\Phi}^{\uparrow}(b) + \mathbf{T}^{tb} \mathbf{\Phi}^{\downarrow}(t), \end{aligned} \tag{2}$$

where $\mathbf{\Phi}^{\uparrow}(\tau)$ and $\mathbf{\Phi}^{\downarrow}(\tau)$ are vectors describing the upwards and downwards radiance at depth τ with respect to the basis. The depths t and b correspond to the top and bottom surface, and the square matrices $\mathbf{R}^t$, $\mathbf{R}^b$ and $\mathbf{T}^{tb}$, $\mathbf{T}^{bt}$ describe the reflection and transmission for light arriving at the top and bottom, respectively:



The analogous question to the glass plate example is: given the scattering matrices of two distinct layers (i.e. $\mathbf{R}_1^t, \mathbf{R}_2^t, \mathbf{T}_1^{tb}, \mathbf{T}_2^{tb}$, etc.), what are the scattering matrices of the two layers stacked together?



The solution is similar, but with the $\mathbf{R}$ and $\mathbf{T}$ matrices replacing the scalars R and T. As in the scalar case we sum over all possible sequences of reflections and transmissions, replacing geometric series by their closed-form solutions. Attention must be paid to the ordering of multiplications, since the matrices generally do not

Term	Meaning
$\Phi(\tau,\mu,\phi)$	Continuous radiance function
μ	Cosine of the elevation angle
ϕ	Azimuth angle
τ	Optical depth within a layer
n	Number of discretizations in μ
m	Number of Fourier basis functions
δ_{ij}	Kronecker delta
l	Index used for Fourier expansions
α	Beckmann roughness of a layer
σ_t	Extinction coefficient of a layer
p	Phase function of a layer
f	BSDF of a boundary between layers
$\Phi_l(\mu), \mathbf{\Phi}_l$	Fourier expansion of ϕ and μ-discretization ($\mathbb{R}^n$)
$p_l(\mu,\mu'), \mathbf{P}_l$	Fourier expansion of p and μ-discretization ($\mathbb{R}^{n\times n}$)
$f_l(\mu,\mu'), \mathbf{F}_l$	Fourier expansion of f and μ-discretization ($\mathbb{R}^{n\times n}$)
$\mathbf{W}$	Integration weights of the quadrature scheme ($\mathbb{R}^{n\times n}$)

Table 1: *Notation used in this paper*

commute. For a rigorous discussion of this derivation, we refer the reader to [Grant and Hunt 1969]. The final result of this computation, analogous to (1), are the so-called *adding equations*:

$$\begin{aligned}
\tilde{\mathbf{R}}^t &= \mathbf{R}_1^t + \mathbf{T}_1^{bt}(\mathbf{I}-\mathbf{R}_2^t\mathbf{R}_1^b)^{-1}\mathbf{R}_2^t\mathbf{T}_1^{tb}\\
\tilde{\mathbf{R}}^b &= \mathbf{R}_2^b + \mathbf{T}_2^{tb}(\mathbf{I}-\mathbf{R}_1^b\mathbf{R}_2^t)^{-1}\mathbf{R}_1^b\mathbf{T}_2^{bt}\\
\tilde{\mathbf{T}}^{tb} &= \mathbf{T}_2^{tb}(\mathbf{I}-\mathbf{R}_1^b\mathbf{R}_2^t)^{-1}\mathbf{T}_1^{tb}\\
\tilde{\mathbf{T}}^{bt} &= \mathbf{T}_1^{bt}(\mathbf{I}-\mathbf{R}_2^t\mathbf{R}_1^b)^{-1}\mathbf{T}_2^{bt}
\end{aligned} \tag{3}$$

The same approach works equally well to compute the effect of a rough interface at the top or bottom of a layer or at the boundary between two layers.

The adding equations are a key ingredient of our system, since they permit accurate computation of the scattering properties of stacks of layers. We compute the scattering matrix of a layered material by repeatedly using Equation (3) to combine layers and boundaries based on a structural description of the material. This requires knowing the matrices of the interior of layers, which we discuss in Section 3.4 and of the boundaries, which is covered in Section 3.4.1.

3.2 Problem statement

Before describing the basis we use to represent scattering functions and the methods used to compute the matrices used in the adding equations, we first formalize the global plane-parallel radiative transfer problem to be solved. The geometric situation is illustrated in Figure 2: we have a sequence of layers, each containing a homogeneous medium, separated by interfaces that are described by BSDFs. Because of the 1D transport assumption, the radiance depends only on direction and depth. For simplicity we define the dimensionless optical depth at z, measured from the top, as

$$\tau(z) \coloneqq \int_0^z \sigma_t(z)\,\mathrm{d}z \quad \text{(dimensionless)}$$

where $\sigma_t(z)$ denotes the *extinction coefficient* at depth z inside the layered material (having units of $1/\text{distance}$).

Radiance is then denoted as $\Phi(\tau,\mu,\phi)$, where $\mu=\cos\theta\in[-1,1]$ is the cosine of the elevation angle, and $\phi\in[0,2\pi]$ is the azimuth. Using this parameterization, the radiative transport equation inside

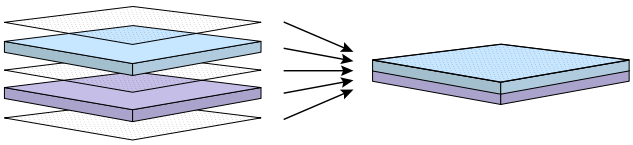


Figure 2: *Our framework combines interface and layer operators to form a combined material BSDF.*

a layer takes on the following form [Chandrasekhar 1960]:

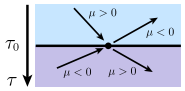
$$\begin{aligned}
\mu\frac{\mathrm{d}\Phi(\tau,\mu,\phi)}{\mathrm{d}\tau} &= -\Phi(\tau,\mu,\phi)\\
&\quad + \int_0^{2\pi}\int_{-1}^{1}\Phi(\tau,\mu',\phi')p(\mu',\phi',\mu,\phi)\,\mathrm{d}\mu'\mathrm{d}\phi',
\end{aligned} \tag{4}$$

where $p(\mu,\phi,\mu',\phi')$ is the phase function of the layer, which is a function of the angle between the directions (μ,ϕ) and (μ',ϕ'). Note that p implicitly accounts for the albedo of scattering interactions and will generally integrate to a value less than one.

The boundary conditions of this equation are the BSDFs at the interfaces between layers. To describe interactions of light with these layer boundaries, we must distinguish between incident and exitant radiance, since a boundary at some depth τ_0 generally introduces a discontinuity in Φ:

$$\Phi^i(\tau_0,\mu,\phi) = \begin{cases}\Phi(\tau_0^-,\mu,\phi), & \mu\geq 0\\ \Phi(\tau_0^+,\mu,\phi), & \mu<0\end{cases}$$

$$\Phi^o(\tau_0,\mu,\phi) = \begin{cases}\Phi(\tau_0^+,\mu,\phi), & \mu\geq 0\\ \Phi(\tau_0^-,\mu,\phi), & \mu<0\end{cases}$$



With these definitions, the surface illumination integral at a layer boundary (accounting for both reflection and transmission) takes the form:

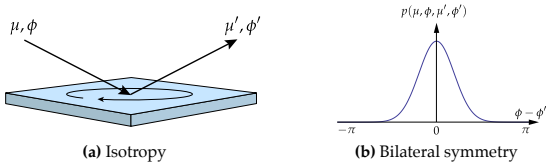
$$\Phi^o(\tau_0,\mu,\phi) = \int_0^{2\pi}\int_{-1}^{1}\Phi^i(\tau_0,\mu',\phi')f(\mu',\phi',\mu,\phi)\,|\mu'|\,\mathrm{d}\mu'\mathrm{d}\phi', \tag{5}$$

where f is the BSDF of the boundary.

3.3 Directional basis

Before we can proceed to use the adding equations to solve actual problems, we must decide on a basis that is used to represent the space of radiance functions. To discretize Φ in direction, we rely on a basis originally proposed by Chandrasekhar [1960]. It represents the light distribution at depth τ using a Fourier series in the azimuth angle ϕ and point samples in the elevation angle cosines μ.

We restrict our analysis to surfaces that are isotropic, in the sense of invariance with rotation around the normal (a), and add the further reasonable assumption of bilateral symmetry (b):



(a) Isotropy **(b)** Bilateral symmetry

Together these imply that all relevant quantities only depend on μ, μ', and $|\phi–\phi'|$. We expand the components of the model in an even

real Fourier expansion with respect to $\phi - \phi'$. In scattering volumes we expand the radiance distribution Φ and phase function p:

$$\Phi(\tau, \mu, \phi) = \sum_{l=0}^{\infty} \Phi_l(\tau, \mu) \cos l\phi \tag{6}$$

$$p(\mu, \phi, \mu', \phi') = \sum_{l=0}^{\infty} p_l(\mu, \mu') \cos(l(\phi - \phi')), \tag{7}$$

and at an interface at depth τ_0, we define analogous expansions of the incident and scattered radiance functions, $\Phi_l^i(\tau_0, \mu)$ and $\Phi_l^o(\tau_0, \mu)$, and the boundary BSDF $f_l(\mu, \mu')$. Substituting these definitions into the scattering integral in (4), and simplifying using the convolution property of the cosine basis, puts the inscattered radiance (the final term in (4)) in the form of a Fourier series,

$$\sum_{l=0}^{\infty} \int_{-1}^{1} \Phi_l(\tau, \mu')\, p_l(\mu', \mu)\, (\pi(\cos(l\phi) + \delta_{0l}))\, \mathrm{d}\mu'$$

where δ_{0l} is the Kronecker delta. Equating coefficients yields:

$$\mu \frac{\mathrm{d}\Phi_l(\tau, \mu)}{\mathrm{d}\tau} = -\Phi_l(\tau, \mu) + \pi(1 + \delta_{0l}) \int_{-1}^{1} \Phi_l(\tau, \mu')\, p_l(\mu', \mu)\, \mathrm{d}\mu'. \quad (l = 0, \ldots) \tag{8}$$

A similar transformation of the surface reflection equation reads:

$$\Phi_l^o(\mu) = \pi(1 + \delta_{0l}) \int_{-1}^{1} \Phi_l^i(\mu')\, f_l(\mu', \mu)\, |\mu'|\, \mathrm{d}\mu'. \quad (l = 0, \ldots) \tag{9}$$

This form reveals a key benefit of the Fourier expansion—the problem of computing radiance as a function of μ, μ', and $\phi - \phi'$ has been reduced to a sequence of *separate* problems each involving only μ and μ'. In practice, the coefficients p_l and f_l decay as $l \to \infty$ so that only problems up to some maximum index m need to be solved. This is dictated by the lowest frequency factor in the integral—e.g., for a diffuse layer only the $l = 0$ term is necessary regardless of the frequency of the illumination. For brevity, we will from now on omit the $(l = 0, \ldots)$ suffix with the understanding that each equation with an unbound l is a sequence of equations.

Fourier series are often treated with reluctance when dealing with functions that may contain narrow peaks, but they offer important benefits in this case: the separation into independent problems for each azimuthal mode significantly reduces the difficulty of the problem that must be solved. This is a consequence of the convolution theorem, which is unique to the Fourier family of basis functions. By taking advantage of inherent symmetries in the underlying scattering functions, their dimension is furthermore reduced from 4D to 3D. As we will see later, in conjunction with sparsity, these properties will allow us to go to high order expansions to represent even mirror-like reflectors accurately. Finally, projecting existing material models onto this space involves a sequence of one-dimensional integrals, which can be evaluated cheaply using recurrences (details in Section 5). Our system could in principle also be realized using another basis, but this would involve complications due to the loss of at least one of the above properties.

3.3.1 Discretization over elevation angles

We now turn to the μ dependence: here, a quadrature scheme with integration nodes $\{\mu_1, \ldots, \mu_n\} \subset [-1, 1] \setminus \{0\}$ and weights $\{w_1, \ldots, w_n\}$ is used to discretize the integration variable μ'. This turns the Fourier-space surface illumination integral (9) into a sum:

$$\Phi_l^o(\mu) = \pi(1 + \delta_{0l}) \sum_{i=1}^{n} w_i\, \Phi_l^i(\mu_i)\, f_l(\mu_i, \mu)\, |\mu_i|\,, \tag{10}$$

and using matrix notation, we can write this equation as

$$\mathbf{\Phi}_l^o(\tau_0) = \pi(1 + \delta_{0l})\, \mathbf{F}_l \mathbf{W} \overline{\mathbf{M}} \mathbf{\Phi}_l^i(\tau_0), \tag{11}$$

where $\mathbf{\Phi}_l(\tau) \coloneqq (\Phi_l(\tau, \mu_k))_k$, and similarly for $\mathbf{\Phi}_l^i$ and $\mathbf{\Phi}_l^o$, which are n-vectors, and $\mathbf{F}_l \coloneqq (f_l(\mu_i, \mu_j))_{ij}$ and the diagonal $\mathbf{W} \coloneqq (\delta_{ij} w_i)_{ij}$ and $\overline{\mathbf{M}} \coloneqq (\delta_{ij} |\mu_i|)_{ij}$, which are $n \times n$ matrices. An analogous process applied to the equation of transfer (8) leads to the matrix differential equation

$$\mathbf{M} \frac{\mathrm{d}\mathbf{\Phi}_l(\tau)}{\mathrm{d}\tau} = -\mathbf{\Phi}_l(\tau) + \pi(1 + \delta_{0l}) \mathbf{P}_l \mathbf{W} \mathbf{\Phi}_l(\tau), \tag{12}$$

where $\mathbf{P}_l$ is defined similarly to $\mathbf{F}_l$ and $\mathbf{M} \coloneqq (\delta_{ij} \mu_i)_{ij}$.

A wide range of quadrature schemes are applicable in this setting; in our implementation, we use Gauss-Lobatto points [Chandrasekhar 1960], which include the endpoints ± 1 and maximize the order of exactly integrable polynomials subject to this constraint.

3.4 Solving for the scattering matrices

Recall that in our system, we plan to use the adding equations (Section 3.1), whose input are scattering matrices $\mathbf{R}^t, \mathbf{R}^b, \mathbf{T}^{tb}, \mathbf{T}^{bt}$ describing the response of a layer to incident illumination. An important consequence of our transition to a frequency representation in azimuth is that each layer now in fact has a *sequence* of these matrices labeled $\mathbf{R}_l^t, \mathbf{R}_l^b, \mathbf{T}_l^{tb}, \mathbf{T}_l^{bt}$ $(l = 0, \ldots, m)$, which describe its response to illumination with different azimuthal Fourier modes. In this section, we show how to find these scattering matrices, starting with the simpler case of interaction with a boundary.

3.4.1 Scattering matrices of layer boundaries

For layer boundaries, it is straightforward to extract the scattering matrices directly from the Fourier-projected BSDF $\mathbf{F}_l$ based on Equation (11). For this, let us partition the matrix $\mathbf{F}_l$ into the four sub-blocks $\mathbf{F}_l^t, \mathbf{F}_l^b, \mathbf{F}_l^{tb}$ and $\mathbf{F}_l^{bt}$ corresponding to the different modes of reflection and transmission at the top and bottom surfaces. Then we have

$$\left[\begin{array}{c|c} \mathbf{T}_l^{tb} & \mathbf{R}_l^b \\ \hline \mathbf{R}_l^t & \mathbf{T}_l^{bt} \end{array}\right] = \pi(1 + \delta_{0l}) \left[\begin{array}{c|c} \mathbf{F}_l^{tb} & \mathbf{F}_l^b \\ \hline \mathbf{F}_l^t & \mathbf{F}_l^{bt} \end{array}\right] \mathbf{W} \overline{\mathbf{M}}. \tag{13}$$

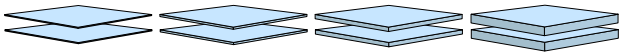
In Section 5 we look at specific scattering models to find $\mathbf{F}_l$.

3.4.2 Scattering matrices of layers

Several different techniques exist that can be used to solve for the scattering matrices of medium layers. In our system, we implemented the *discrete ordinates method* and *adding-doubling* techniques, and found the adding-doubling method to be generally preferable due to its robustness[1]. Adding-doubling builds on the property that, as a function of optical depth, multiple scattering is a higher-order effect that can be neglected for sufficiently thin layers. On the other hand, scattering matrices of layers with *at most* a single scattering event are easily obtained, since they admit an analytic

[1] The discrete ordinates method is discussed in the supplement; it enables us to handle layers whose parameters are continuous functions of depth.

solution. The idea of adding-doubling then is as follows: after computing the scattering matrices of a very thin layer (thin enough that multiple scattering can be neglected), the results of Section 3 are used to find the scattering matrices of a layer twice the thickness, by joining two identical layers. The layer is repeatedly doubled until it has the desired thickness.



Since τ increases exponentially, even very thick layers can be processed rapidly.

In our implementation, we start with a depth of $\mathrm{d}\tau = 2^{-15}$. To obtain layers with thicknesses other than powers of two times $\mathrm{d}\tau$, we first represent the number $\lfloor (b-t)/\mathrm{d}\tau \rfloor$ in base 2 (where $b-t$ is the desired thickness). A simple loop then iterates through the digits in increasing magnitude and adds a layer of thickness $2^i \mathrm{d}\tau$ on top of the partially generated layer whenever the i-th bit is 1.

Returning again to the projected and discretized radiative transfer equation (12), note that it describes how one "unit" of optical depth of the ambient medium locally influences the radiance function:

$$\frac{\mathrm{d}\mathbf{\Phi}_l(\tau)}{\mathrm{d}\tau} = \mathbf{M}^{-1}\left(-\mathbf{I} + \pi(1+\delta_{0l})\mathbf{P}_l\mathbf{W}\right)\mathbf{\Phi}_l(\tau).$$

To obtain the scattering matrices of a thin layer with optical depth $\mathrm{d}\tau$, we can then simply use the above differential equation as the linear term of a Taylor expansion. With some liberty in notation, this can be interpreted as

$$\mathbf{\Phi}_l^{\text{after}} = \mathbf{\Phi}_l^{\text{before}} + \mathrm{d}\tau\left[\mathbf{M}^{-1}\left(-\mathbf{I} + \pi(1+\delta_{0l})\mathbf{P}_l\mathbf{W}\right)\right]\mathbf{\Phi}_l^{\text{before}}$$

More formally, let us partition the matrix $\mathbf{P}_l$ into the four sub-blocks $\mathbf{P}^t, \mathbf{P}^b, \mathbf{P}^{tb}$ and $\mathbf{P}^{bt}$ corresponding to the different modes of reflection and transmission at the top and bottom surfaces. Then we have

$$\left[\begin{array}{c|c} \mathbf{T}_l^{tb} & \mathbf{R}_l^b \\ \hline \mathbf{R}_l^t & \mathbf{T}_l^{bt} \end{array}\right] = \mathbf{I} + \mathrm{d}\tau\,\mathbf{M}^{-1}\left(-\mathbf{I} + \pi(1+\delta_{0l})\left[\begin{array}{c|c} \mathbf{P}_l^{tb} & \mathbf{P}_l^b \\ \hline \mathbf{P}_l^t & \mathbf{P}_l^{bt} \end{array}\right]\mathbf{W}\right)$$

which tells us how to find the scattering matrix of the initial layer of depth $\mathrm{d}\tau$, which is required to start up the adding and doubling process discussed above, from the phase function matrices $\mathbf{P}$.

So far, we have explained the basic mathematical framework of our system, which consists of a directional basis and the requisite methods to turn the matrix representations of phase functions and BSDFs in that basis into a scattering matrix for any layered material. The remainder of the paper is structured as follows: Section 4 gives an overview of the core algorithm and ties together later sections. Section 5 shows how to find the matrix representations $\mathbf{P}_l$ and $\mathbf{F}_l$ of relevant phase functions and BSDF models that we required in Sections 3.4.1-3.4.2. Section 5.3 fixes an energy loss problem in traditional microfacet models that causes difficulties in layered BRDFs, and Section 6 explains how to efficiently evaluate and importance sample our model in a renderer. Finally, Section 7 demonstrates applications of our model and Section 8 concludes the paper.

4 Algorithm overview

Algorithm 1 implements the central component that computes the Fourier modes of the reflection and transmission matrices for each layer, which are subsequently combined using the adding equations. Inputs to this algorithm are a structural model of the material consisting of the phase functions and optical depth of layers, the BSDFs

Algorithm 1 Solve for the scattering matrices of a layered material

```
1  function SOLVE-SCATTERING-MATRICES(structure, n, m)
2      for l ← 0 to m do                                   ▷ For each Fourier mode
3          R̂_l^t, R̂_l^b ← 0,  T̂_l^tb, T̂_l^bt ← I          ▷ Initialize as clear layer
4          for L in structure do                           ▷ Iterate over structure
5              if L is a boundary layer then
6                  Compute F_l                             ▷ Section 5.2
7                  Extract R_l^t, R_l^b, T_l^tb, T_l^bt    ▷ Section 3.4.1
8              else if L is a medium layer then
9                  Compute P_l                             ▷ Section 5.1
10                 Solve for R_l^t, R_l^b, T_l^tb, T_l^bt  ▷ Section 3.4.2
11             Use the adding equations to merge           ▷ Section 3.1
               R_l^t, R_l^b, T_l^tb, T_l^bt into R̂_l^t, R̂_l^b, T̂_l^tb, T̂_l^bt
12     return R̂_l^t, R̂_l^b, T̂_l^tb, T̂_l^bt (l = 0, ..., m)
```

of boundaries, as well as the targeted discretization n in μ and μ' and the number of Fourier expansion coefficients m for the azimuth difference angle $\phi - \phi'$. The discretization is related to the accuracy of the model, and in Section 5.4 we present conservative bounds on these parameters.

This entire computation is an offline process in the sense that it is executed before rendering starts. The output of this process is a set of $n \times n$ matrices $\mathbf{R}_l^t, \mathbf{R}_l^b, \mathbf{T}_l^{tb}, \mathbf{T}_l^{bt}$ $(l = 0, \ldots, m)$ that characterize the material's response to illumination. Section 6 explains how this representation can be used by a rendering algorithm.

Note that the high level structure of Algorithm 1 itself is not new, being a combination of standard techniques in 1D linear transport theory. The key contributions of this paper follow in the subsequent sections and focus on making each of the steps sufficiently general and scalable to tackle relevant problems in rendering.

Sparsity: At this point, we make an observation with important implications regarding the scalability of our approach:

1. To represent very peaked reflectance functions, we require many Fourier coefficients, but this is only the case for a small set of elevation pairs (μ, μ'). For instance, in the case of specular reflection, the expansions have high frequencies only when $\mu \approx \mu'$ and low frequencies or even zeroes elsewhere.

2. Smoother reflectance functions are nonzero over many pairs (μ, μ'), but they are low frequency in the azimuth difference angle $\phi - \phi'$, and thus their Fourier series decay rapidly.

Scattering matrices of high frequency materials expressed in Chandrasekhar's [1960] directional basis are *sparse*; our system therefore relies on sparse linear algebra techniques, allowing us to go to very high orders to represent even mirror-like materials without ringing or other artifacts (Figure 3), while generating BSDF representations that require comparably little storage. We have not found any references of this property in the literature and believe that we are the first to exploit it.

We now turn to the computation of $\mathbf{P}_l$ and $\mathbf{F}_l$ on lines 6 and 9.

5 Scattering models

Our system supports several kinds of scattering functions: for layer boundaries of dielectrics and conductors, we use the microfacet model proposed by Walter et al. [2007], and medium layers are modeled using linear combinations of the Henyey-Greenstein [1941] phase function and the von Mises-Fisher lobe proposed by Gkioulekas et al. [2013]. The main difficulty in deriving the matri-

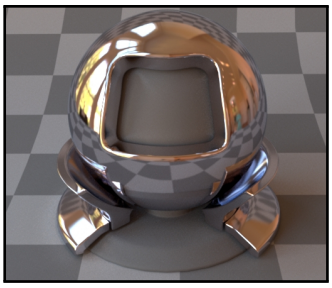


Figure 3: *Frequency-based representations are usually impractical when trying to represent specular materials like this chrome object with Beckmann roughness $\alpha = 0.01$. We projected this material into a basis with $m = 9763$ Fourier series terms and $n = 503$ discretizations in μ_i and μ_o, which would normally produce about 28 GiB of dense coefficient data. By exploiting sparsity and our efficient Fourier projections, we require only 51.3 MiB of coefficients (0.19%) computed in 9.6 seconds; rendering took 1.4 minutes.*

ces $\mathbf{P}_l$ and $\mathbf{F}_l$ lies in finding Fourier expansions

$$p_l(\mu,\mu') = \mathcal{F}_l\left[p(\mu,\mu',\cdot)\right] \coloneqq \frac{2-\delta_{0l}}{\pi}\int_0^\pi p(\mu,\mu',\phi)\cos(l\phi)\mathrm{d}\phi,$$

for phase functions p and analogous for BSDFs f. The dependence on the elevations μ and μ' is simple as it is handled by discretization. We begin by looking at expansions of the phase function.

5.1 Phase functions

5.1.1 Henyey-Greenstein phase function

The Henyey-Greenstein (HG) phase function is a widely used one-dimensional family of phase functions with a single parameter $g \in [-1, 1]$, which specifies the medium's disposition of scattering light into the backward or forward direction. As a function of γ, the angle between the incident and outgoing directions, it is defined as

$$p(\cos\gamma) = \frac{1-g^2}{4\pi\left(1+g^2-2g\cos\gamma\right)^{3/2}}. \tag{14}$$

To compute the Fourier expansions of p, Stam [2001] numerically evaluated a power series for each coefficient to be computed. We found this series to converge very slowly, particularly for high g. Instead, we use a technique by Yanovitskij [1997b] to rapidly and accurately compute the coefficients in one pass using recurrences (code is provided in the supplementary technical report).

We wish to highlight a curious aspect of Yanovitskij's recurrence because it will resurface in the next section. His derivation shows that any three successive Fourier coefficients of the HG model relate as

$$(2l+1)p_{l+2} = (l+1)\,\rho\,p_{l+1} - (2l+3)p_l, \tag{15}$$

where $\rho \in \mathbb{R}$ does not depend on l. This suggests the following simple implementation: after computing p_0 and p_1 using an arbitrary numerical or analytic approach, simply perform one sweep for $l = 2, \ldots, m$, each time using Equation (15) to find the next p_l from p_{l-1} and p_{l-2}. However, this does not work: after a few iterations, severe cancellation and error amplification in Equation (15) lead to errors that are larger than the magnitude of the correct answer. Yanovitskij therefore uses the recurrence *the other way*, going from p_{l+2} and p_{l+1} to p_l, where it is numerically well-behaved. This leads to the somewhat unusual situation of having to start a recurrence at the *end* of a desired range rather than its beginning, but the resulting implementation is fast and numerically stable.

5.1.2 von Mises–Fisher phase function

Due to its inherent simplicity, the HG model is a common default choice when simulating anisotropic volumetric scattering, but in a recent study, Gkioulekas et al. [2013] analyzed the perceptual significance of different phase function spaces and recommended switching to a larger space containing linear combinations of the HG model and lobes of the von Mises–Fisher distribution defined as

$$p_{\mathrm{vMF}}(\gamma) = \frac{\kappa}{4\pi\sinh\kappa}e^{\kappa\cos\gamma} \tag{16}$$

We derive a recurrence for this phase function model which turns out to be useful a second time to compute Fourier expansions of rough boundaries in Section 5.2. We begin by noting that $\cos\gamma$ expressed in terms of μ, ϕ and μ', ϕ' is equal to

$$\cos\gamma = \mu\mu' + \sqrt{(1-\mu^2)\,(1-\mu'^2)}\cos(\phi-\phi'). \tag{17}$$

After rewriting Equation (16) in terms of the azimuthal difference angle $\phi_d = \phi - \phi'$ via (17) and applying an identity of the hyperbolic sine we have

$$p_{\mathrm{vMF}}(\phi_d) = \frac{\kappa}{2\pi(1-e^{-2\kappa})}e^{A+B\cos\phi_d} \tag{18}$$

where $A = \kappa(\mu\mu' - 1)$ and $B = \kappa\sqrt{(1-\mu^2)(1-\mu'^2)}$ do not depend on ϕ_d and can thus be considered constants for the purpose of this derivation.

Using the Jacobi-Anger expansion (Abramowitz and Stegun, 9.6.34), we turn the trigonometric exponential in (18) into a Fourier series involving modified Bessel functions of the first kind:

$$e^{A+B\cos\theta} = e^A\left[I_0(B) + 2\sum_{l=1}^{\infty} I_l(B)\cos(l\theta)\right]. \tag{19}$$

Technically, this completes the derivation of the Fourier projection, but in practice Bessel functions are very costly to evaluate, and in our system, they can be invoked with arguments that lead to overflows in floating point arithmetic due to their fast growth (consider that $I_0(1000) \approx 2.5 \cdot 10^{432}$); this is particularly an issue with uses of this expansion later in the paper. To address these issues, we start by noting that Equation (19) actually takes on very reasonable values in the interval $[0, 1]$ since $A + B\cos\phi_d \leq 0$. To always remain in the representable number range, it is thus key that we incorporate the exponential scaling by e^A into the computation of the coefficients. We use the following expression for computing the l-th scaled Bessel function:

$$e^A I_l(B) = \underbrace{e^{-B}I_0(B)}_{=\,I_0^e(B)}\; e^{A+B}\prod_{j=1}^{l}\frac{I_j(B)}{I_{j-1}(B)} \tag{20}$$

The first term $I_0^e(B)$ is the exponentially scaled modified Bessel function of order zero which does not suffer from overflow and can be found in standard numerical libraries. The last term is a product of *ratios* of Bessel functions, which are close to 1 and easily representable. These ratios satisfy a recurrence relation which we use not only to evaluate (20) but also to efficiently find all series terms of (16) for a pair of angles μ, μ' (i.e. for constant A and B) in one sweep. The recurrence is given by

$$\frac{I_{l+1}(B)}{I_l(B)} = \frac{I_{l-1}(B)}{I_l(B)} - \frac{2l}{B}. \tag{21}$$

Interestingly, similar to the Henyey-Greenstein recurrence in Section 5.1.1 this relation also suffers from catastrophic cancellation

and error magnification when used in the upward sense (i.e. computing the ratio for $l+1$ from the previous one at l), while being stable when used in the *downward* sense:

$$\frac{I_l(B)}{I_{l-1}(B)} = \frac{B}{2l + B\frac{I_{l+1}(B)}{I_l(B)}} \tag{22}$$

Our implementation then starts with the maximum expansion order $l = m$, and computes the ratios I_l/I_{l-1}, I_{l-1}/I_{l-2} down to I_1/I_0 using recurrences. However, this requires knowing the *last* ratio I_l/I_{l-1} to bootstrap the recurrence formula: for this purpose, we use a quickly converging continued fraction representation based on Gauss' work on ratios of hypergeometric functions (see Gautschi and Slavik [1978] for a general discussion):

$$\frac{I_l(B)}{I_{l-1}(B)} = \cfrac{1}{\frac{2}{B}l + \cfrac{1}{\frac{2}{B}(l+1) + \cfrac{1}{\frac{2}{B}(l+2) + \cdots}}} \quad (l > 0) \tag{23}$$

Combining the continued fraction representation, the recurrence over ratios and the exponentially scaled function I_0^e, we obtain an efficient and stable method for computing azimuthal Fourier series of the von Mises–Fisher distribution.

5.2 Boundaries between layers

To model boundaries of dielectrics and conductors, we use the microfacet model proposed by Walter et al. [2007]. Microfacet models describe the interaction of light with random surfaces composed of microscopic dielectric or conducting facets that are oriented according to a microfacet distribution. Integration over this distribution then leads to simple analytic expressions that describe the expected reflection and transmission properties at a macroscopic scale. Validations against real-world measurements have shown that microfacet models compare favorably against other families of parametric BRDF models [Ngan et al. 2005].

We begin with a short review of the microfacet model by Walter et al., specifically the variant that uses the Beckmann distribution derived from Gaussian random surfaces. This model consists of a reflection and a transmission term

$$f(\mu_i, \phi_i, \mu_o, \phi_o) = f_r(\mu_i, \mu_o, \phi_i - \phi_o) + f_t(\mu_i, \mu_o, \phi_i - \phi_o). \tag{24}$$

The reflection term f_r is defined as

$$f_r(\mu_i, \mu_o, \phi_d) = \frac{F(\mu_h)D(\mu_h)G(\mu_i, \mu_o)}{4|\mu_i \mu_o|}$$

where F specifies the Fresnel reflectance, D is the Beckmann distribution, G is a shadowing-masking term, and μ_h denotes the cosine of the angle between the normal and the half-direction vector of the incident and outgoing directions. The refractive case is defined analogously but involves a modified half direction vector and different weighting terms. Note that while the paper by Walter et al. only concerns itself with dielectrics, it is straightforward to generalize the model to also handle conductors by replacing F with the general unpolarized Fresnel reflectance term for boundaries with a complex-valued relative index of refraction and setting $f_t = 0$. We refer to the original papers for many further details and a full discussion of the the refraction term f_t.

To compute the azimuthal modes of f we must again integrate

$$\mathcal{F}_l\left[f(\mu, \mu', \cdot)\right] = \frac{2 - \delta_{0l}}{\pi} \int_0^\pi f(\mu, \mu', \phi_d) \cos(l\phi_d)\, \mathrm{d}\phi_d. \tag{25}$$

However, we were not able to find an analytic expression for this integral due to the significant complexity of the integrand. Forced to turn to numerical integration, we encountered further problems:

- f varies greatly in magnitude with respect to ϕ_d: usually, only a tiny portion of the of the domain contributes to the integral.
- Higher-order Fourier basis functions are oscillatory, i.e. they have a large number of lobes of opposing signs, which almost (but not quite) cancel each other.

The combination of these properties makes both general-purpose and special oscillatory integration methods impracticably slow. To sidestep these difficulties, we propose a semi-analytic integration routine tailored to this specific problem.

We note that, although the full expression of the BSDF f has several parts that depend on ϕ_d, the exponential term in the Beckmann microfacet distribution is the one that, by far, dominates its behavior. Splitting the integrand of (25) into this exponential term and a function f_{rem} containing the remaining terms produces

$$\mathcal{F}_l\left[f(\mu_i, \mu_o, \cdot)\right] = \frac{2 - \delta_{0l}}{\pi} \int_0^\pi f_{\mathrm{rem}}(\phi_d)\, e^{A + B\cos\phi_d} \cos(l\phi_d)\, \mathrm{d}\phi_d$$

where in the reflective case A and B are defined as

$$A = \frac{\mu_i^2 + \mu_o^2 - 2}{\alpha^2(\mu_i - \mu_o)^2}, \quad B = \frac{2\sqrt{1-\mu_i^2}\sqrt{1-\mu_o^2}}{\alpha^2(\mu_i - \mu_o)^2}. \tag{26}$$

In the refractive case, they are given by

$$A = \frac{\eta_i^2\left(\mu_i^2 - 1\right) + \eta_o^2\left(\mu_o^2 - 1\right)}{\alpha^2(\eta_i\mu_i - \eta_o\mu_o)^2},\ B = \frac{2\eta_i\eta_o\sqrt{1-\mu_i^2}\sqrt{1-\mu_o^2}}{\alpha^2(\eta_i\mu_i - \eta_o\mu_o)^2}. \tag{27}$$

We observe a striking similarity to the exponential-of-cosine Fourier series in Equation (19) encountered during our treatment of the von Mises–Fisher phase function, with the main change being the extra term $f_{\mathrm{rem}}(\phi_d)$ which contains normalization and shadowing-masking terms as well as the Fresnel coefficient. Suppose for a moment that we were able to obtain series representations of these pieces *separately*, i.e. coefficients a_l and b_l such that

$$e^{A + B\cos\phi_d} = \sum_{l=0}^{\infty} a_l \cos(l\phi_d) \ \text{ and } \ f_{\mathrm{rem}}(\phi_d) = \sum_{l=0}^{\infty} b_l \cos(l\phi_d)$$

The Fourier series we actually wanted to obtain is the pointwise multiplication of these two series. At this point, we note that as a consequence of the convolution theorem, the frequency space analog to a pointwise multiplication in the ϕ_d argument is the *discrete convolution* of the number sequences (a_n) and (b_n). If a_l and b_l are easily obtained, and if one of them decays quickly enough, this yields an attractive way of finding the coefficients of the product. We found that this is indeed the case: for conductors, f_{rem} is generally smooth enough to representable with relative error of 10^{-6} using only six series terms (we conservatively use 12). For dielectrics, more coefficients are needed in certain cases (details in the supplemental material), but the average is still very low, around 14-20 depending on the parameters. Importantly, changing the roughness α does not change the frequency content of f_{rem}, and we can therefore depend on the robustness of our exponential-of-cosine expansion to handle low-roughness cases.

Our approach then is to compute *two* Fourier series and find their discrete convolution: one of the high frequency exponential, where we simply use the algorithm already developed in Section 5.1.2, and another of $f_{\mathrm{rem}}(\phi_d)$, which is handled using a traditional numerical method for Fourier integrals (we use Filon quadrature [1928]).

Measured materials: It is also possible to import measured materials into our system to compose them with other layers. We have implemented this for materials in the database of Matusik et al. [2003], for which we approximately Fourier-project the BSDF for each pair μ_i, μ_o by densely sampling it in azimuth $[0, 2\pi]$ and applying a Fast Fourier Transform. This is relatively fast but only an approximation; also the required resolution depends on the material's specularity and must be determined manually. We leave better integration schemes for such "black box" data as a future work.

5.3 Multiple scattering term

One issue with currently used microfacet models is that they only account for a single scattering event at the microgeometry level; light that interacts with multiple facets is effectively ignored. This means that these models incur some energy loss which grows steadily as the roughness of the interface is increased. In a simulation of layered materials with multiple rough internal boundaries, this loss is incurred many times due to interactions between layers, potentially removing significant amounts of energy.

We propose an additive correction term to the microfacet model, which reintroduces any energy lost to multiple scattering. This term is approximate—in particular, we assume that, following multiple interactions within the surface microgeometry, the scattered radiation emerges with an angular distribution that is close to diffuse. The main constraint on this correction is that it should be reciprocal like other parts of our system. The discrete analog of continuous reciprocity [Veach 1997] in our setting is a set of matrix equations:

$$\mathbf{F}_l^t = \mathbf{F}_l^{tT}, \quad \mathbf{F}_l^b = \mathbf{F}_l^{bT}, \quad \eta_t^2 \mathbf{F}_l^{tb} = \eta_b^2 \mathbf{F}_l^{btT} \qquad (l = 0, \ldots, m)$$

The assumption of multiply scattered illumination emerging with an approximately diffuse profile allows us to restrict the correction to the 0^{th} order Fourier mode (i.e. $l = 0$), and to preserve reciprocity we use the following rank-1 update of $\mathbf{F}$:

$$\widehat{\mathbf{F}}_0^t = \mathbf{F}_0^t + \mathbf{r}^t \mathbf{r}^{tT}, \qquad \widehat{\mathbf{F}}_0^{tb} = \mathbf{F}_0^{tb} + \frac{\mathbf{t}^{bt} \mathbf{t}^{tbT}}{\eta_b^2}$$

and analogous for $\mathbf{F}_0^{tb}$ and $\mathbf{F}_0^{bt}$. The vectors $\mathbf{r}^t$, $\mathbf{r}^b$, $\mathbf{t}^{tb}$ and $\mathbf{t}^{bt}$ are chosen appropriately so that the final representation has no energy loss. We derived expressions for these vectors for rough conductors and dielectrics. Our approach is related to work by Kelemen et al. [2001] who derived a continuous correction term for a specular-matte BRDF. The derivation of these expressions is fairly technical, hence we refer the reader to the supplementary material for details. Figure 4 highlights the effect of our correction on dielectrics and conductors of different roughness values.

5.4 Error analysis

To use Algorithm 1 in practice we need a way of selecting an appropriate discretization n in elevation, and number m of Fourier expansion terms. How these parameters should be set is not obvious; overly high values will cause unnecessary overheads while a too low value may introduce unacceptable errors. Intuitively, their choice relates to the "peakedness" of the ingredients, i.e. the parameter α for boundaries, and g or κ for scattering layers.

We propose a heuristic for each kind of layer in the supplemental material that specifies appropriate n and m if that layer were be represented *in isolation*. These heuristics are simple functions that map the relevant parameter to a discretization with less than 1% relative L^2 representation error. We created these heuristics by bounding n and m values found via a brute force search (Figure 5).

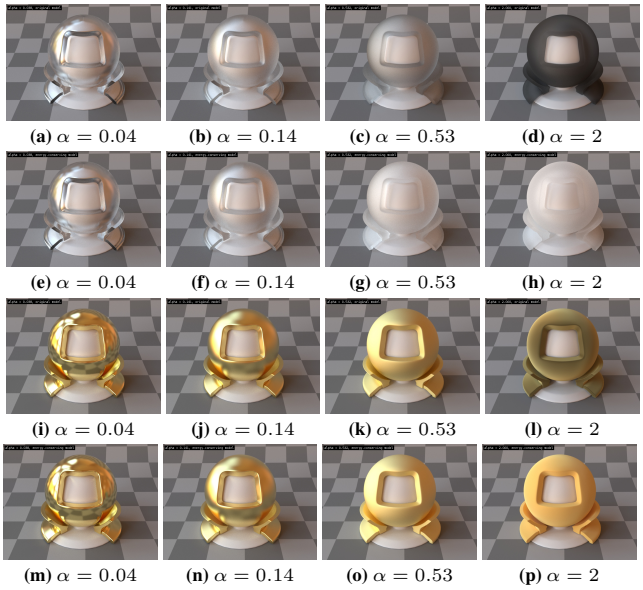


Figure 4: *Traditional microfacet models for dielectrics and conductors (rows 1 and 3) suffer from energy loss that is particularly problematic in simulations of layered materials. We propose a multiple scattering term that reintroduces this energy (rows 2 and 4).*

For a material made of multiple layers $i = 1, \ldots, N$ these heuristics will generally recommend a set of incompatible parameter values $(n_1, m_1), \ldots, (n_N, m_N)$. However, for Algorithm 1 to work we must decide on *a single* value of n and m that are used for the entire computation. For the n parameter, we simply set $n = \max_i n_i$. In the parameter m we do not need to be as conservative: each layer i can be thought of as a low-pass filter that removes azimuthal frequency content that is higher than order m_i. Therefore, all interior layers are "filtered" by the top and bottom layers, and hence we set $m = \max\{m_1, m_N\}$.

6 Evaluation and sampling

All elements of our system described until now tie into Algorithm 1 in one way or another. This algorithm runs as an offline process prior to rendering and produces a sequence of scattering matrices $\mathbf{R}_l^t, \mathbf{R}_l^b, \mathbf{T}_l^{tb}, \mathbf{T}_l^{bt} (l{=}0, \ldots, m)$. The last part of this pipeline is discussed below; its purpose is to convert the resulting data into a format that is more convenient for rendering before writing it to disk.

Recall Equation (13), which we previously used to turn a projected BSDF model $\mathbf{F}_l$ into scattering matrices suitable for computations involving the adding equations. We now apply this equation once more in the reverse direction to convert the final set of matrices $\mathbf{R}_l^t, \mathbf{R}_l^b, \mathbf{T}_l^{tb}, \mathbf{T}_l^{bt}$ back into a projected BSDF $\mathbf{F}_l$ $(l = 0, \ldots, m)$. We then transpose this tensor so that the Fourier coefficients $(\mathbf{F}_l)_{i,j}$ are contiguous with respect to the index l, matching the order in which they will be accessed sequentially later on. To exploit sparsity, we also determine a minimal effective expansion order $m(i, j)$ separately for each pair of elevation angles, satisfying $(\mathbf{F}_l)_{i,j} = 0$ for $l > m(i, j)$. Only coefficients up to this order are stored.

The remainder of this section discusses how to use this information in a rendering algorithm. In our experiments, we used a standard Monte Carlo path tracer, which requires the ability to evaluate the BSDF and importance sample directions proportional to it; we show how to implement both of these operations efficiently.

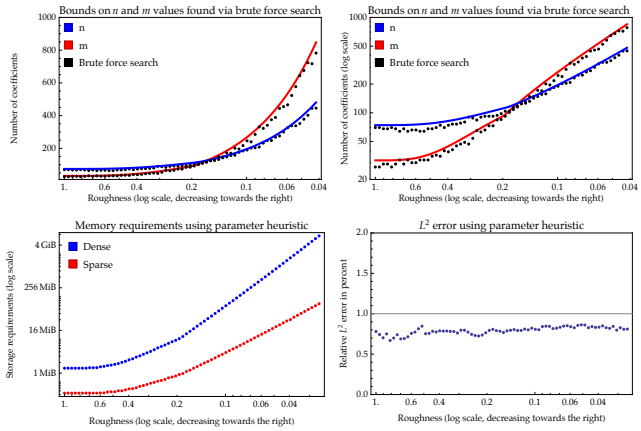


Figure 5: *To develop a good heuristic for selecting the discretization and expansion order (n, m), we instantiated the supported layer types of our system over a range of parameters and ran a search for the cheapest representation with less than 1% error. We bounded the resulting values from above using a simple analytic expressions that capture the asymptotics (top row: bounds in normal and log domain for a rough dielectric with $\eta = 1.5$, bottom row: storage requirements and relative error achieved by our bound).*

6.1 Model evaluation

To evaluate the model for a pair of directions μ_i, ϕ_i and μ_o, ϕ_o, we determine the location in the sparse file that contains the azimuthal Fourier expansion for the given elevations (μ_i, μ_o) and fetch the associated coefficients $a_0, \ldots, a_k$. The BRDF value is then simply

$$f(\mu_i, \phi_i, \mu_i, \phi_o) = \sum_{l=0}^{k} a_l \cos(l(\phi_i - \phi_o)). \tag{28}$$

This works when μ_i and μ_o are part of the set of discretized elevations, but this is generally not the case. For intermediate values, we interpolate the Fourier coefficients from those of nearby elevation angles on the fly using the 4×4 interpolation stencil of a standard 2D tensor product Catmull-Rom spline.

A naïve implementation of Equation (28) can lead to serious performance issues during rendering due to very many trigonometric function calls per BSDF evaluation. To reduce these to a minimum, we use the multiple angle formula for cosines

$$\cos(n\phi) = 2\cos(\phi)\cos((n-1)\phi) - \cos((n-2)\phi) \tag{29}$$

as a recurrence while iterating through the terms of the sum. To accelerate computations even further, we exploit instruction level parallelism when evaluating (28). All sum terms are independent from each other, and hence we can load and process groups of several coefficients at a time, using iterated forms of (29) to find the cosine values needed for the entire group in one step.

Texturing: To texture an arbitrary parameter of a layer (e.g. its optical depth σ_t), we compute several BSDF models corresponding to samples throughout the targeted parameter domain. To evaluate the BSDF for a particular parameter value we simply interpolate between these models. For parameters with a linear dependence, two samples suffice; higher numbers are necessary to capture nonlinear effects. Multiple parameters can be textured simultaneously, though eventually the exponential growth in the required number of samples becomes burdensome. This scheme is approximate but works very well in practice for many types of parameters, including diffuse or phase function albedos or optical depth. Other parameters are less suitable (e.g. index of refraction and boundary roughness) and may require many samples when textured in this way. This is a limitation of our system and an interesting avenue for future work.

6.2 Importance sampling

The spline interpolation scheme introduced in Section 6.1 induces continuous extensions of the BSDF matrices $\mathbf{F}_l$ to piecewise polynomial functions $F_l : [-1, 1] \times [-1, 1] \to \mathbb{R}$. For any two angles (μ_i, μ_j) from the initial discretization, $F_l(\mu_i, \mu_j)$ simply returns entry (i, j) from the underlying matrix; for arguments that lie between samples of the discretization, nearby entries are interpolated.

Note the special significance of the 0-th order function F_0, which describes the average value of the BSDF integrated over azimuth:

$$F_0(\mu_i, \mu_o) = \frac{1}{\pi} \int_0^\pi f(\mu_i, \mu_o, \phi_d)\, \mathrm{d}\phi_d.$$

This means that the dependence on azimuth is effectively "marginalized" out of the model, a fact which we use to build a simple and efficient importance sampling method that is split into separate elevation and azimuth angle sampling steps. Because both can be implemented as smooth mappings from a single uniform variate to an outgoing angle, they are particularly well-suited for use with certain structured point sets, such as Sobol or Halton sequences, leading to reduced variance in renderings that use them.

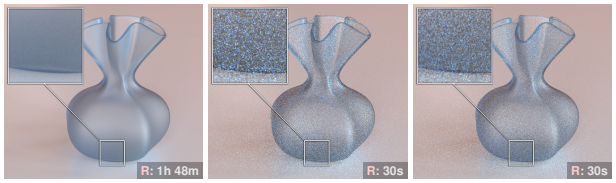
Both steps involve a standard numerical inversion method for integrals, which we review for completeness: to draw samples proportional to a 1-dimensional density function $q(x) \geq 0$ on an interval $[a, b]$, we generate a uniform variate ξ and solve for x in $Q(x) = \xi$, where $Q(x) = \int_a^x q(x')\, \mathrm{d}x'$. Note that Q is continuous and monotonically increasing; since $Q(a) = 0$ and $Q(b) = 1$, this guarantees that a unique solution exists; $[a, b]$ is known as a bracketing interval. We rely on a hybrid Newton-Bisection method to invert Q, which maintains a bracketing interval as an invariant, while using Newton-Raphson steps whenever possible to benefit from quadratic convergence rates near the solution.

Elevation sampling: Given a fixed incident elevation μ_i, this step picks an outgoing elevation with a probability proportional to the piecewise cubic spline function $F_0(\mu_i, \cdot)$. For simplicity, let us briefly assume that μ_i is equal to one of the elevation angles from the discretization. Let $I_{i,k}$ denote the definite integral of $F_0(\mu_i, \cdot)$ up to the k-th spline segment, i.e.

$$I_{i,k} := \int_{\mu_1}^{\mu_k} F_0(\mu_i, \mu')\, \mathrm{d}\mu' \quad (k = 1, \ldots, n).$$

We first use a $\mathcal{O}(\log n)$ binary search to map a uniform variate $\xi_1 \in [0, 1)$ to the segment k satisfying $I_{i,k} \leq \xi_1 I_{i,n} < I_{i,k+1}$. Following this, we apply the discussed Newton-Bisection method to robustly invert the integral of F_0 (a quartic) on $[\mu_k, \mu_{k+1}]$, producing the desired outgoing elevation μ_o. The entries of the matrix $I_{i,k}$ have analytic solutions that can be precomputed ahead of time. Elevations μ_i that lie between samples of the discretization are easy to handle due to the linearity of this setup: we can simply interpolate the precomputed integrals $I_{i,k}$ using the same Catmull-Rom basis.

Azimuth sampling: To sample the azimuthal component conditioned on (μ_i, μ_o), we fetch the associated interpolated Fourier coefficients $a_l := F_l(\mu_i, \mu_o)$ from the sparse coefficient storage and pick an azimuth difference angle ϕ_d proportional to the resulting Fourier expansion $q(\phi_d)$. Finally, we set $\phi_o = \phi_i + \phi_d$. As before, the main sampling operation involves the inversion of a definite integral Q using the hybrid Newton-Bisection approach. This integral



(a) *Analytic (8K spp)* **(b)** *Analytic (38 spp)* **(c)** *Ours (26 spp, eq. time)*

Figure 6: *Reflectance models generated by our system support exact importance sampling, which can make them interesting even in cases where an analytic reflectance model exists. The images on the middle and right show equal-time renderings of a sand-blasted vase modeled using a standard microfacet BSDF. Note improved convergence in* ***(c)*** *despite the lower number of samples per pixel.*

fortunately has a simple explicit form:

$$Q(\phi_d) = \int_0^{\phi_d} \sum_{l=0}^{m} a_l \cos(l\phi'_d)\,\mathrm{d}\phi'_d = a_0\phi_d + \sum_{l=1}^{m} l^{-1} a_l \sin(l\phi_d)$$

The recurrence from Section 6.1 can also be applied here to avoid costly trigonometric function evaluations. Instruction level parallelism offers further acceleration opportunities.

The proposed importance sampling scheme is *exact* in the sense that it produces a constant importance weight ≤ 1 over all scattered directions when holding the incident direction fixed. Additional terms (e.g. the cosine of the surface illumination integral) are easily absorbed into the data to account for them as well.

This is a rare situation, since BSDF sampling techniques are almost never exact. This effectively means that our approach can be interesting even when an analytic (and potentially faster to evaluate) reflectance model is available. Figure 6 shows a rough glass vase ($\eta = 1.5, \alpha = 0.3$) rendered with the analytic evaluation and sampling of Walter et al. [2007] and the approach presented in this section. Renderings were performed using Quasi Monte-Carlo integration, specifically scrambled Halton points.

The analytic sampling technique only samples one factor of the model (the microfacet distribution) and relies on a transformation from microfacet normals to outgoing directions, which distorts the equidistribution of the Halton points. In comparison, our technique is able to sample *all* terms of the BSDF directly on the sphere of outgoing directions.

6.3 Harmonic extrapolation

We found that for the special case of scattering and absorbing dielectric slabs with smooth or rough boundaries, the azimuthal dependence of the resulting BSDFs was well approximated by a simple 3-parameter family of functions created by mixing a uniform and a wrapped Gaussian distribution on the circle, i.e.

$$g_{\delta,\beta,\sigma^2}(\phi) \coloneqq \frac{\delta}{2\pi} + \beta \sum_{k=-\infty}^{\infty} G(\phi + 2\pi k, \sigma^2), \qquad (30)$$

for constants δ, β, and σ^2. For such cases, our framework supports what we refer to as *harmonic extrapolation* to obtain additional performance improvements in preprocessing and rendering time.

This works by equating the computed Fourier modes of the scattering matrices to the modes of (30) and solving for the parameters δ, β, and σ^2. Afterwards, the Fourier coefficients can be discarded and (30) is used instead. Since this approach extrapolates the entire behavior from the first three harmonics, Algorithm 1 can be stopped after $m = 2$, which significantly reduces the precomputation time.

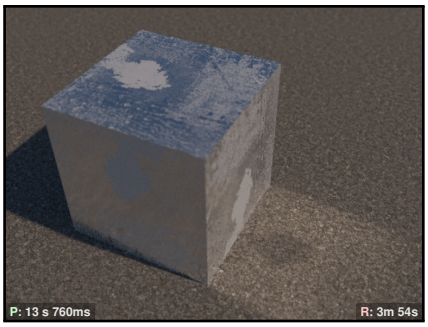


Figure 7: *Gu et al. [2007] provide a database of measured surface contaminants. We use one of their optical thickness maps to render a shiny chrome cube with dried deposits due to salt water exposure. Note how the salt blocks part of the caustic on the ground.*

7 Results

We implemented this technique in C++ on top of the Mitsuba renderer [Jakob 2010], relying on the Eigen library [Guennebaud et al. 2010] for sparse linear algebra computations. BSDF sampling and evaluation exploit instruction parallelism using Intel's AVX instruction set. To solve the sparse linear systems in the adding Equations (Section 3.1), we use supernodal LU decompositions [Li 2005].

To ensure the correctness of our system, we routinely performed validation tests—these included checking that projection of scattering models into the directional basis is accurate, comparing against reference data including angular reflectance data of index-matched isotropic and anisotropic slabs (Tables 12 and 35 of [van de Hulst 1980]) and albedo values for isotropic half-spaces with dielectric boundaries [Williams 2006]. Finally, we also ran comparisons against Monte Carlo renderings with explicit layer structure (these comparisons are in the supplement). In all cases, our validations showed excellent agreement, usually matching reference data in the radiative transfer literature to four significant digits.

We now present a series of results produced using our system. Precomputation and rendering times are reported for an Intel Xeon E5-2660 machine with 16 physical (32 hyper-threaded) cores.

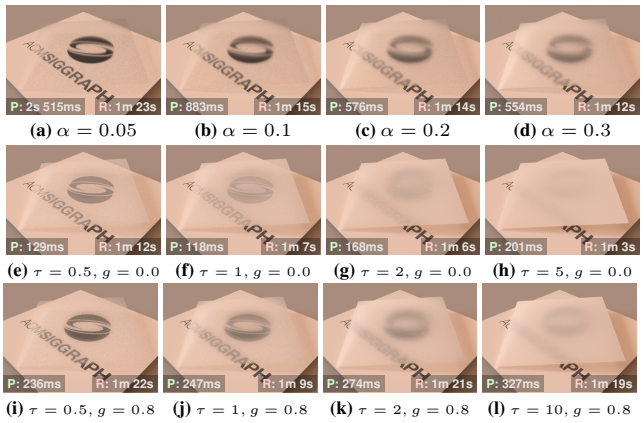
Interior: Figure 1 depicts a range of layered materials generated by our model. The corresponding layer structures are shown on the sides, and detailed statistics regarding processing time and storage requirements are presented in Table 2. The wooden kitchen counter material is a coated diffuse surface with texture-mapped albedo; to capture the material's non-linear dependence on albedo we use four BRDF samples over the admissible parameter range $[0, 1]$. Following precomputation, the image was path traced at with a resolution of 1800×1080 pixels using 2048 samples per pixel to resolve glossy interreflections between objects, which took 1 hour and 23 minutes.

Contaminants maps: Figure 7 shows a shiny chrome cube with dried salt water deposits rendered under daylight illumination. We modeled this as a shiny chrome base ($\alpha = 0.03$) with an index-matched anisotropic layer ($g = 0.7$) whose optical thickness is textured by a contaminant map of Gu et al. [2007]. To capture the nonlinear dependence in σ_t, we precomputed six BRDF samples over the parameter range. The combined BRDF with discretization $n = 158$ and expansion order $m = 2540$ required 21.6 MiB as opposed to 4.3 GiB for a dense version (a reduction to 0.5%) and was computed in 13.7 s. Rendering using Manifold Exploration MLT [Jakob 2013] using 64 mutations/pixel took 3m 54s.

Surface vs. volumetric scattering: The two classes of transport operators in our framework—surface and volume—act on the incoming light fields in fundamentally different ways. It is instruc-

Material	n	m	Time	Storage	Sparsity
Coffee table	394	671	3.34s	68.4MiB	5.7%
Red vase	196	267	2.09s	11.9 MiB	9.9%
Coated copper vase	196	267	2.52s	3 MiB	2.5%
Large purple bowl	196	267	4.31s	5.2 MiB	4.3%
Measured BRDF	200	200	1.69s	22.7MiB	24%
Coated measured BRDF	200	267	3.49s	3.2 MiB	2.5%
White brick wall	196	267	0.96s	1.3 MiB	3.2%
Kitchen sink	502	890	7.52s	9.6 MiB	1.1%
Kitchen counter (wood)	394	671	6.5s	55.5 MiB	1.2%
Kitchen counter (metal)	236	4194	1.41s	10.1 MiB	0.38%

Table 2: *Discretization, preprocessing time, storage requirements, and the percentage of nonzero coefficients of the objects in Figure 1.*



(a) $\alpha = 0.05$ **(b)** $\alpha = 0.1$ **(c)** $\alpha = 0.2$ **(d)** $\alpha = 0.3$

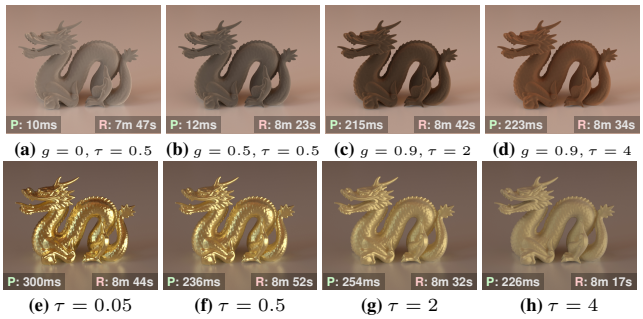
(e) $\tau = 0.5, g = 0.0$ **(f)** $\tau = 1, g = 0.0$ **(g)** $\tau = 2, g = 0.0$ **(h)** $\tau = 5, g = 0.0$

(i) $\tau = 0.5, g = 0.8$ **(j)** $\tau = 1, g = 0.8$ **(k)** $\tau = 2, g = 0.8$ **(l)** $\tau = 10, g = 0.8$

Figure 8: *Comparing the visual loss in clarity as caused by surface roughness versus volumetric scattering inside a layer. The top row ((a)-(d)) shows the transmission through a dielectric plate with identical roughness α on both faces. The lower two rows demonstrate index-matched plates with volumetric imperfections simulated by HG scattering (which may be appropriate for material with air pockets or other inclusions). A variety of appearances is evident as the thickness and anisotropy are varied ((e)-(l)). The rough dielectric scattering produces a blurring effect, while the volumetric scattering leads to a hazing of the transmitted signal.*

tive to analyze their behaviors in isolation. Figure 8 compares the different loss of clarity caused by transmission through layers with rough surfaces versus layers with smooth surfaces, but with internal volumetric scattering. Roughening both sides of the glass plate causes a uniformly blurred transmission of the background scene, whereas forward-peaked volumetric scattering inside plates with smooth surfaces causes more of a hazed appearance. The arbitrary combination of this wide variety of operators yields a very flexible system for building BSDFs.

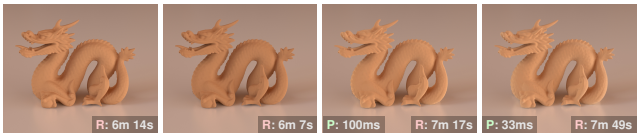
Asperity Scattering and Diffuse Materials: Adding a volumetric scattering layer to a base material can be used for asperity scattering [Koenderink and Pont 2003], useful for simulating fine layers of dust or fibers on a surface. We illustrate this in Figure 9 by visualizing various forms of asperity layerings on a black base layer, and then to simulate dust on a gold statue. Note that these renderings resemble those of Pharr and Hanrahan [2000], whose nonlinear scattering equations provide an alternative way of solving for the interaction between layers. However, their approach does not lead to a reflectance model in explicit form: during the rendering process, every interaction with a layered material triggers a nested Monte Carlo integration over layers. Our system is the first to account for all orders of interreflection between an asperity and underlying layer, and to include multiple scattering in the asperity layer without requiring a costly integration over layers at render time.



(a) $g = 0, \tau = 0.5$ **(b)** $g = 0.5, \tau = 0.5$ **(c)** $g = 0.9, \tau = 2$ **(d)** $g = 0.9, \tau = 4$

(e) $\tau = 0.05$ **(f)** $\tau = 0.5$ **(g)** $\tau = 2$ **(h)** $\tau = 4$

Figure 9: *Top: asperity scattering using a volumetric scattering layer on top of a black surface. Various thicknesses τ and HG phase functions (with mean cosine g) are compared. The single-scattering albedo is slightly orange in all cases. Because more scattering events are required to scatter a similar amount of light for $g = 0.9$, the result is more saturated. Bottom: applying various thicknesses τ of grey HG ($g = 0.9$) dust to a rough gold statue.*



(a) *Lambert* **(b)** *Oren-Nayar* **(c)** $g = 0$ **(d)** $g = 0.7$

Figure 10: *A wide variety of diffusive reflectance models are possible by combining volumetric scattering layers and Lambertian reflectors. Note the different silhouettes and overall subtle appearance differences between the volumetric models in (c) and (d) vs. the traditional diffuse BRDFs in (a) and (b). Parameters in (c), (d) were chosen to yield the same albedo under normal illumination.*

Volumetric scattering layers can yield a wide variety of new diffusive BRDFs that complement that standard analytic models. Figure 10 compares Lambert and Oren-Nayar BRDFs to volumetric scattering materials with isotropic or forward-peaked Henyey-Greenstein scattering. The use of an isotropic scattering material (c) yields a very dusty look, which is lessened in (d) by layering a HG scattering layer $g = 0.8, \tau = 5$ on top of a 90% Lambertian reflector. The volumetric-scattering renders were no more than 30% slower, and the BSDFs were compactly stored in 3.5 KiB.

Approximate layered BRDFs: The model of Weidlich and Wilkie [2007] supports texturable layered materials with anisotropic interfaces but does not account for all orders of interreflection between layers. This leads to incorrect albedos from simple combinations, such as a smooth dielectric interface over a Lambertian reflector. Figure 11 compares the total albedo of such a model with refractive index 1.5 as a function of diffuse reflectance k_d of the underlying Lambertian surface. Due to internal scattering, the total reflected energy has an inherently nonlinear dependence on k_d. Weidlich and Wilkie's model discards such internal reflection light paths and uses an heuristic scaling factor to compensate for the resulting loss of energy. However, the nonlinear behavior of the layers is not captured regardless of whether or not the heuristic is used. Another limitation of this approach is that it cannot faithfully simulate the interaction of multiple rough boundaries, which requires a full integration over internal reflection paths.

Fitting to measured data: Figure 13 shows transmission and reflectance measurements for a milky light blue layer of vitreous enamel. The black dots specify radiance recorded along the material's normal direction while a light source performs a 180° sweep around the layer (the left half of the plot being transmission). We replicated this lighting and viewing geometry in our framework and performed an automated search for a layer that best fits the observed

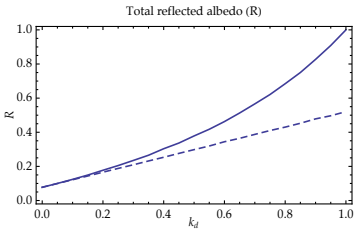


Figure 11: *Comparing the total reflected energy R of a smooth dielectric $\eta = 1.5$ on top of a Lambertian surface with reflectance k_d. A recent approximate model (dashed) proposes a linear relation between k_d and total reflectance, which is not seen in the exact solution that considers all orders of interreflection.*

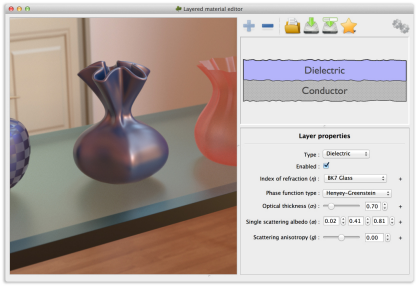


Figure 12: *We developed an interactive visual design tool to explore the space of layered BRDF models; the supplementary video shows an editing session using it.*

data. By fixing index of refraction and roughness (both approximately known), a three parameter family of models remained, involving albedo, optical depth, and anisotropy. The search found the fit shown in blue, which is in good agreement with the data.

Visual layer design tool: To conveniently assemble materials from layers and render them, we developed a visual design tool shown in Figure 12. The supplementary video contains a screen recording of an interactive material editing session using this tool.

8 Conclusion

We have presented a new framework for rendering layered materials. Our system allows arbitrary layered combinations of any known analytic or measured isotropic BSDF and produces new energy-conserving, reciprocal BSDFs with importance sampling. All interreflections between layers are computed at the same level of accuracy that is used for the primary light transport throughout the scene. This allows seamless level-of-detail transitions for practically authoring complex scenes as well as allowing design of layered materials in a fashion that is closely tied to their physical constituents, which is useful for rapid authoring of heterogeneous or temporally-varying materials and essential for predictive rendering. Combining smooth and rough interface BSDFs with volumetric scattering layers produces a large variety of new layered BSDFs, including expressive generalizations of diffuse surfaces and asperity scattering layers. We have demonstrated the practicality of our approach, which hinges critically on novel methods for treating rough surfaces and scattering events in a stable fashion, a new method for conserving energy at rough interfaces, and also sparse methods and extrapolation techniques for compact storage and accelerated computation of our BSDFs. Our system remains practical over a large range of model parameters, scaling even to narrowly peaked reflectance functions, but cannot represent perfect mirrors or ideally smooth glass, which are Dirac delta functions in direction. Ultimately, analytic solutions exist for such cases that will be preferable to numerical approaches like ours.

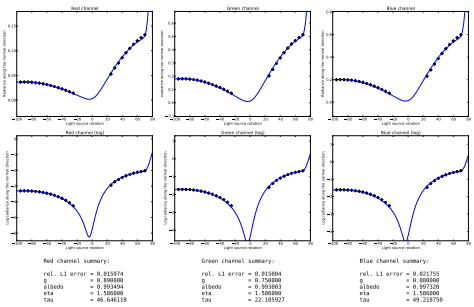


Figure 13: *Our method can be used to determine suitable scattering parameters for measured materials; here we fitted it to transmission and reflectance measurements of a 1 mm layer of a milky light blue vitreous enamel used in jewelry design. This dielectric had a known index of refraction and was very smooth, leaving only a three parameter family of models to be explored involving albedo, optical depth, and anisotropy. An automatic search found the fit shown as a blue curve, which is in good agreement with the data (left to right: RGB, top: radiance, bottom: log radiance).*

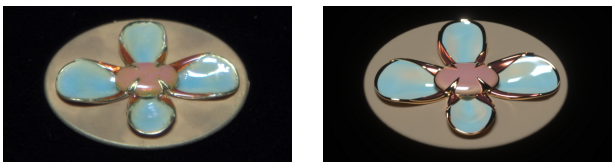


Figure 14: *Left: photograph of a flower art piece manufactured by casting a 3D printed wax model into gold and manually coating it with jewelry enamel. The petals use the same enamel also measured in Figure 13. Note the subtle hue shifts due to varying layer thickness. Right: a rendering using our framework. For details, please refer to the supplemental material.*

9 Acknowledgments

The authors are indebted to Olesya Isaenko, who kindly designed the scene shown in Figure 1, as well as the shape of the art piece in Figure 14. We thank Albert Liu for assistance with measurement calibrations and Norm McCormick for helpful discussions. Funding for this work was provided by NSF grant 1011919 and the Intel Science and Technology Center for Visual Computing. Wenzel Jakob was supported by an ETH/Marie Curie fellowship. This project was started during an internship at Weta Digital.

References

ARONSON, R. 1971. Relation between the transfer matrix method and Case's method. *Transport Theory and Stat. Physics 1*, 3.

BLINN, J. F. 1982. Light reflection functions for simulation of clouds and dusty surfaces. In *Proc. SIGGRAPH 1982*, vol. 16.

CHALHOUB, E., CAMPOS VELHO, H. D., GARCIA, R., AND VILHENA, M. 2003. A comparison of radiances generated by selected methods of solving the radiative-transfer equation. *Transport Theory and Statistical Physics 32*, 5-7, 473–503.

CHANDRASEKHAR, S. 1960. *Radiative Transfer*. Dover Publications.

DAI, Q., WANG, J., LIU, Y., SNYDER, J., WU, E., AND GUO, B. 2009. The dual-microfacet model for capturing thin transparent slabs. In *Computer Graphics Forum*, vol. 28.

DAS, R. 2010. Solution of Riemann–Hilbert problems for determination of new decoupled expressions of Chandrasekhar's X-and

Y-functions for slab geometry in radiative transfer. *Astrophysics and Space Science 326*, 1, 91–103.

DONNER, C., AND JENSEN, H. W. 2005. Light diffusion in multi-layered translucent materials. *Transactions on Graphics 24*, 3.

DORSEY, J., AND HANRAHAN, P. 1996. Modeling and rendering of metallic patinas. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM.

ERSHOV, S., KOLCHIN, K., AND MYSZKOWSKI, K. 2001. Rendering pearlescent appearance based on paint-composition modelling. *Computer Graphics Forum 20*, 3, 227–238.

FILON, L. 1928. On a quadrature formula for trigonometric integrals. In *Proc. Roy. Soc. Edinburgh*, vol. 49, 38–47.

GARCIA, R. 2012. Radiative transfer with polarization in a multi-layer medium subject to Fresnel boundary and interface conditions. *Journal of Quant. Spectroscopy and Rad. Transf. 115*, 0.

GAUTSCHI, W., AND SLAVIK, J. 1978. On the computation of modified bessel function ratios. *Mathematics of Computation 32*, 143, 865–875.

GKIOULEKAS, I., XIAO, B., ZHAO, S., ADELSON, E. H., ZICKLER, T., AND BALA, K. 2013. Understanding the role of phase function in translucent appearance. *ACM Trans. Graph. 32*, 5.

GRANT, I., AND HUNT, G. 1969. Discrete space theory of radiative transfer. i. fundamentals. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences 313*, 1513.

GU, J., RAMAMOORTHI, R., BELHUMEUR, P., AND NAYAR, S. 2007. Dirty glass: Rendering contamination on transparent surfaces. In *Proceedings of EGSR '07*, 159–170.

GUENNEBAUD, G., JACOB, B., ET AL., 2010. Eigen v3. http://eigen.tuxfamily.org.

HANRAHAN, P., AND KRUEGER, W. 1993. Reflection from layered surfaces due to subsurface scattering. In *Proceedings of ACM SIGGRAPH 1993*, 164–174.

HENYEY, L. G., AND GREENSTEIN, H. L. 1941. Diffuse radiation in the galaxy. *Astrophysical Journal 43*, 70–83.

HIRAYAMA, H., KANEDA, K., YAMASHITA, H., AND MONDEN, Y. 2001. An accurate illumination model for objects coated with multilayer films. *Computers & Graphics 25*, 3, 391–400.

ICART, I., AND ARQUÈS, D. 2000. A physically-based BRDF model for multilayer systems with uncorrelated rough boundaries. In *Proceedings of EGWR '00*.

JAKOB, W., 2010. Mitsuba renderer. http://www.mitsuba-renderer.org.

JAKOB, W. 2013. *Light Transport on Path-Space Manifolds*. PhD thesis, Cornell University.

KELEMEN, C., AND SZIRMAY-KALOS, L. 2001. A microfacet based coupled specular-matte BRDF model with importance sampling. In *Eurographics Short Present.*, vol. 25, 34.

KOENDERINK, J., AND PONT, S. 2003. The secret of velvety skin. *Machine Vision and Applications 14*, 4, 260–268.

LI, X. S. 2005. An overview of SuperLU: Algorithms, implementation, and user interface. *ACM Trans. Math. Softw. 31*, 3.

MATUSIK, W., PFISTER, H., BRAND, M., AND MCMILLAN, L. 2003. A data-driven reflectance model. *ACM Transactions on Graphics 22*, 3 (July), 759–769.

MCCORMICK, N., AND KUSCER, I. 1973. Singular eigenfunction expansions in neutron transport theory. *Advances in Nuclear Science and Technology 7*, 181–282.

NGAN, A., DURAND, F., AND MATUSIK, W. 2005. Experimental analysis of BRDF models. In *Proceedings of EGSR '05*, Eurographics Association, 117–226.

PHARR, M., AND HANRAHAN, P. 2000. Monte carlo evaluation of non-linear scattering equations for subsurface reflection. In *Proceedings of SIGGRAPH '00*, 75–84.

PREMOŽE, S. 2002. Analytic light transport approximations for volumetric materials. In *Proc. Pacific Graphics '02*, 48–57.

SHIRLEY, P., SMITS, B., HU, H., AND LAFORTUNE, E. 1997. A practitioners' assessment of light reflection models. In *Computer Graphics and Applications*, IEEE, 40–49.

SIEWERT, C. 1978. The FN method for solving radiative-transfer problems in plane geometry. *Astrophysics and Space Science 58*, 1, 131–137.

SIEWERT, C. 2000. A concise and accurate solution to Chandrasekhar's basic problem in radiative transfer. *Journal of Quantitative Spectroscopy and Radiative Transfer 64*, 2, 109–130.

STAM, J. 2001. An illumination model for a skin layer bounded by rough surfaces. In *Rendering Techniques*, 39–52.

STOKES, G. G. 1860. On the intensity of the light reflected from or transmitted through a pile of plates. *Proceedings of the Royal Society of London 11*, pp. 545–556.

THOMAS, G., AND STAMNES, K. 2002. *Radiative transfer in the atmosphere and ocean*. Cambridge Univ Press.

VAN DE HULST, H. 1980. *Multiple light scattering*. Academic Press.

VEACH, E. 1997. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford University.

WALTER, B., MARSCHNER, S. R., LI, H., AND TORRANCE, K. E. 2007. Microfacet models for refraction through rough surfaces. In *Proceedings of EGSR '07*.

WANG, L., WANG, W., DORSEY, J., YANG, X., GUO, B., AND SHUM, H.-Y. 2005. Real-time rendering of plant leaves. *ACM Trans. Graph. 24*, 3 (July), 712–719.

WEIDLICH, A., AND WILKIE, A. 2007. Arbitrarily layered microfacet surfaces. In *Proceedings of GRAPHITE '07*, 171–178.

WEIDLICH, A., AND WILKIE, A. 2009. Anomalous dispersion in predictive rendering. In *Computer Graphics Forum*, vol. 28.

WILKIE, A., WEIDLICH, A., LARBOULETTE, C., AND PURGATHOFER, W. 2006. A reflectance model for diffuse fluorescent surfaces. In *Proceedings of GRAPHITE '06*, 321–331.

WILLIAMS, M. 2006. The albedo problem with Fresnel reflection. *Journal of Quant. Spectroscopy and Radiative Transfer 98*, 3.

WOLFF, L. B., NAYAR, S. K., AND OREN, M. 1998. Improved diffuse reflection models for computer vision. *International Journal of Computer Vision 30*, 1, 55–71.

YANOVITSKIJ, E. G. 1997. *Light scattering in inhomogeneous atmospheres*. Springer Verlag.

YANOVITSKIJ, E. 1997. A recurrence formula for computing Fourier components of the Henyey-Greenstein phase function. *Journal of Quant. Spectroscopy and Radiative Transfer 57*, 1.