A Full-Wave Reference Simulator for Computing Surface Reflectance

YUNCHEN YU, Cornell University, USA
MENGQI XIA, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
BRUCE WALTER, Cornell University, USA
ERIC MICHIELSSEN, University of Michigan, USA
STEVE MARSCHNER, Cornell University, USA

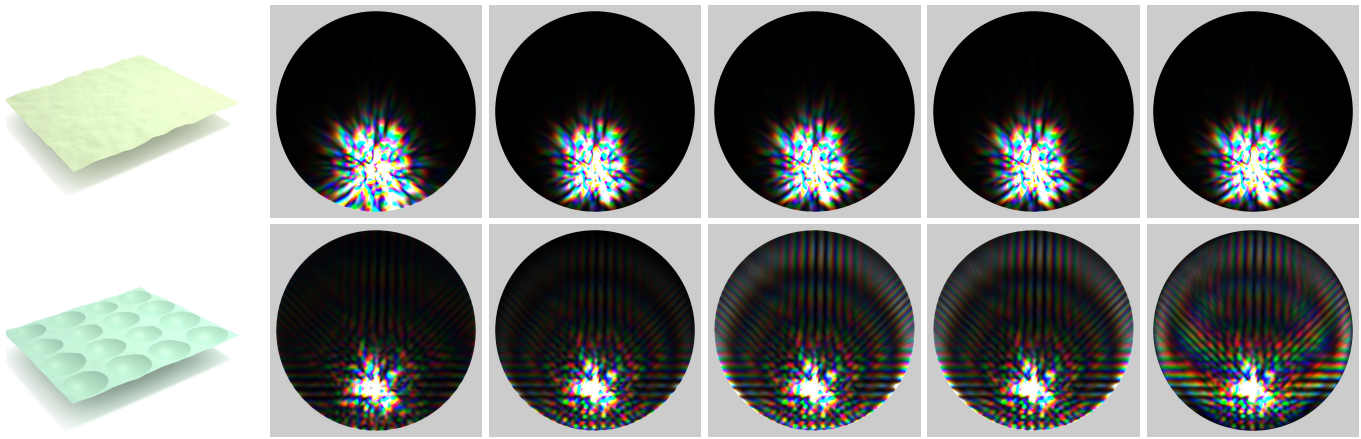

Fig. 1. Wavelength-dependent BRDFs of two surfaces, computed using the original Harvey-Shack model, the generalized Harvey-Shack model, the Kirchhoff model, the tangent plane method in [Xia et al. 2023], and our simulation. The incident direction is given by $\theta_i = 36^\circ, \phi_i = 90^\circ$, and BRDF values are visualized for a large collection of outgoing directions. The models predict similar BRDFs for the smooth surface with isotropic bumps, while for the surface with spherical pits, our simulation predicts brighter color bands in the BRDF, which likely result from interference between reflection of different orders.

Computing light reflection from rough surfaces is an important topic in computer graphics. Reflection models developed based on geometric optics fail to capture wave effects such as diffraction and interference, while existing models based on physical optics approximations give erroneous predictions under many circumstances (e.g. when multiple scattering from the surface cannot be ignored). We present a scalable 3D full-wave simulator for computing reference solutions to surface scattering problems, which can be used to evaluate and guide the development of approximate models for rendering. We investigate the range of validity for some existing wave optics based reflection models; our results confirm these models for low-roughness surfaces but also show that prior rendering methods do not accurately predict the scattering behavior of some types of surfaces.

Our simulator is based on the boundary element method (BEM) and accelerated using the adaptive integral method (AIM), and is implemented to execute on modern GPUs. We demonstrate the simulator on domains up to $60 \times 60 \times 10$ wavelengths, involving surface samples with significant height variations. Furthermore, we propose a new system for efficiently computing BRDF values for large numbers of incident and outgoing directions at once, by combining small simulations to characterize larger areas. Our simulator will be released as an open-source toolkit for computing surface scattering.

CCS Concepts: • **Computing methodologies** → **Reflectance modeling**.

Additional Key Words and Phrases: wave optics, material appearance, scattering, diffraction

ACM Reference Format:
Yunchen Yu, Mengqi Xia, Bruce Walter, Eric Michielssen, and Steve Marschner. 2023. A Full-Wave Reference Simulator for Computing Surface Reflectance. *ACM Trans. Graph.* 42, 4, Article 1 (August 2023), 17 pages. https://doi.org/10.1145/3592414

Authors' addresses: Yunchen Yu, yy735@cornell.edu, Cornell University, USA; Mengqi Xia, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland; Bruce Walter, Cornell University, USA; Eric Michielssen, University of Michigan, USA; Steve Marschner, Cornell University, USA.

0730-0301/2023/8-ART1 $15.00
https://doi.org/10.1145/3592414

1 INTRODUCTION

Accurately modeling light reflection from surfaces, which arises from the interaction of light with various kinds of small-scale geometric structures, is important for rendering realistic and visually compelling images. A surface reflection model in computer graphics typically starts with some assumptions about the target surface structure, and then applies an approximate model of optics to derive a bidirectional reflectance distribution function (BRDF). Many reflection models are based entirely on geometric optics and assume light

travels as rays, and are thus inaccurate for surfaces with feature sizes comparable to visible light wavelengths. Some other models involve wave optics and use approximate diffraction theories, including the Beckmann-Kirchhoff theory [Beckmann and Spizzichino 1987] and the Harvey-Shack family of models [Krywonos 2006].

Most models provide smooth mean BRDFs corresponding to distributions of surfaces, predicting reflection in terms of large-area averages. Some models use either ray or wave optics to compute spatially-varying solutions with high-resolution details [Yan et al. 2016, 2018], but even when modeling the propagation of light as electromagnetic waves, these models still contain significant approximations. Given this large collection of reflection models, it is difficult to determine which models are accurate in a specific context, as a ground truth solution is not available.

The goal of this paper is to provide a reference simulation tool that faithfully models the propagation of light as described by Maxwell's equations and computes optical scattering from small areas of surface microgeometry with no approximation other than numerical discretization, to compute surface BRDFs with high angular and spatial resolution. To account for interference effects, the simulation should scale up to as large as the coherence area of common light sources like sunlight, or tens of microns across.

Reaching this scale, which is large for full wave simulations, requires careful attention to memory usage and performance. Our simulator represents surfaces as height fields, allowing a regular-grid discretization into quadrilateral basis elements, which leads to important simplifications and efficient storage of coefficients. We solve for scattered fields using the boundary integral formulation, an established method in computational electromagnetics, accelerated by the adaptive integral method (AIM), an approach based on 3D Fast Fourier Transform that was originally proposed for accelerating radar calculations. Our accelerated simulations are implemented to run efficiently on modern GPUs, and we demonstrate simulations on domains up to $60 \times 60 \times 10$ wavelengths.

We also propose a new system for solving the graphics-specific problem of efficiently computing BRDF values for a large collection of incident and outgoing directions to completely characterize the bidirectional scattering behavior of a surface. While our simulation is not usually directly applicable to rendering, its high accuracy and broad applicability make it a useful research tool that can evaluate the accuracy and range of validity of approximate reflection models. Our open-source simulation code can be accessed at https://github.com/blaire9989/BEMsim3D.

2 RELATED WORK

In this section we discuss existing surface reflection models as well as relevant methods in computational electromagnetics.

2.1 Wave Optics Based Reflection Models

In computer graphics, most surface scattering models are based on geometric optics (e.g. [Burley 2012; Cook and Torrance 1982; Oren and Nayar 1994; Walter et al. 2007]) due to its low cost and complexity. Prior methods using wave optics have generally relied on physical optics approximations such as Beckmann-Kirchoff [Beckmann and Spizzichino 1987] and Harvey-Shack [Harvey 1979; Krywonos 2006; Krywonos et al. 2011], which are first-order (single scatter), scalar approximations to the full wave equations. These have been used to estimate spatially averaged BRDFs for a variety of surface types, such as Gaussian random [He et al. 1991; Lanari et al. 2017], stationary periodic [Dhillon et al. 2014; Stam 1999; Toisoul and Ghosh 2017], multi-level planar [Levin et al. 2013], tabulated [Dong et al. 2016], and scratched [Werner et al. 2017]. Physical optics has also been used to estimate the spatially varying appearance of tabulated [Yan et al. 2018] and statistically random surfaces [Steinberg and Yan 2022]. Hybrid surface models apply physical optics models for certain surface components, such as small scale roughness [Falster et al. 2020; Holzschuch and Pacanowski 2017], thin films, [Belcour and Barla 2017], or suspended platelets [Guillén et al. 2020], while using simpler geometric optics models for larger scales. One of the goals of our work is to test the accuracy of some physical optics approximations.

Physical optics has also been used to approximate longer range, inter-surface effects [Cuypers et al. 2012; Steinberg et al. 2022]. Wave optics based scattering methods, including the Lorenz-Mie theory and the T-matrix method [Bohren and Huffman 2008; Mishchenko et al. 2002], have also been applied to volumetric scattering [Frisvad et al. 2007; Guo et al. 2021]. Moreover, generalized ray tracing techniques involving complex values have been applied to rendering natural phenomena and structural colors [Sadeghi et al. 2012; Shimada and Kawaguchi 2005]. Inter-surface and volumetric scattering currently lie beyond the scope of our work.

2.2 Computational Electromagnetics and Graphics

The most widely used computational methods in electromagnetics include the finite-difference time-domain (FDTD) method, the finite element method (FEM), and the boundary element method (BEM). FDTD simulations (e.g. [Oskooi et al. 2010]), which solve the differential form of Maxwell's equations based on time stepping and 3D spatial discretization, have been used to predict the appearance of wavelength-scale structures [Auzinger et al. 2018; Musbach et al. 2013], but its cost increases rapidly with domain size, limiting it to 2D domains or 3D domains much smaller than those we use. FEM, which also requires 3D discretization, is similarly limited in simulation size. BEM formulates a scattering problem into integral equations on the surface of the scattering object [Gibson 2021], which lowers the dimension of the discretization and provides better scalability. BEM was previously applied to fibers modeled as generalized cylinders [Xia et al. 2020], but this relied on using translational symmetry to reduce the problem to 2D.

Varied numerical techniques have been used to accelerate BEM, including the fast multipole method (FMM) [Liu and Nishimura 2006; White and Head-Gordon 1994], the adaptive integral method (AIM) [Bleszynski et al. 1996], and the sparse-matrix canonical grid method [Liao et al. 2016; Pak et al. 1997]. Our work uses BEM because of its scalability, and uses AIM to accelerate the computation thanks to its efficiency for domains that are small along one axis.

3 FULL-WAVE SIMULATION

In this work, we present our 3D full-wave simulations for computing spatially-varying surface BRDFs with high resolution details. Our

Table 1. List of symbols.

ω	Angular frequency of light
ε	Permittivity of a homogeneous region
μ	Permeability of a homogeneous region
$\mathbf{E}$	Electric field
$\mathbf{H}$	Magnetic field
$\mathbf{J}$	Electric current density
$\mathbf{M}$	Magnetic current density
λ	Wavelength of the light in vacuum
η	Refractive index of a particular medium
k	Wavenumber of the light in a particular medium
$\mathbf{r}, \mathbf{r}'$	Position of points in the 3D space
$G(\mathbf{r}, \mathbf{r}')$	Green's function
w	Waist of a Gaussian beam
θ	Divergence of a Gaussian beam
$\mathbf{f}_m, \mathbf{f}_n$	Vector basis functions
ψ_m, ξ_n	Scalar components of basis functions
Λ	Point source approximation coefficient matrix
ω_i, ω_o	Incident and outgoing directions
$f_r(\omega_i, \omega_o)$	BRDF values

overall method starts with a surface described by a height field and its material properties (complex refractive index), together with a target point on the surface. To compute the BRDF for a particular incident direction, we define an incident field that propagates towards the target point from the given direction. The incident field is then used in our surface scattering simulation, which solves for the resulting scattered electromagnetic field. The scattered field is evaluated in the far-field limit to produce a full hemisphere of BRDF values for all outgoing directions.

Our simulation is based on the 3D boundary element method (BEM), which is discussed in detail by Gibson [Gibson 2021]; Xia et al. [Xia et al. 2020] provide a summary of the method in a 2.5D setting. In this section we state the principles of BEM in our context. In Section 4 we describe the computational methods for implementing BEM efficiently, and in Section 5 we describe how to efficiently combine multiple simulations to synthesize BRDFs that are densely sampled in both incident and scattered directions. Important symbols used in these sections are summarized in Table 1.

3.1 Boundary Element Method: The Basics

BEM targets single-frequency scattering problems, which involve incident electric and magnetic fields of a given angular frequency ω and a scattering object whose boundary divides the space into two homogeneous regions. The constitutive parameters of the region that contains the incident fields are given by (ε_1, μ_1), and those of the other region are given by (ε_2, μ_2). Here, $\varepsilon_1, \varepsilon_2$ represent the permittivity and μ_1, μ_2 represent the permeability.

Notably, we work with complex-valued field quantities which carry both amplitude and phase information. The field quantities are all assumed to be time-harmonic, with their time dependence given by $e^{j\omega t}$. The term $e^{j\omega t}$ is suppressed throughout the text.

Maxwell's Equations and Surface Currents. Solving wave scattering problems fundamentally relies on Maxwell's equations. The time-harmonic Maxwell's equations are given by

$$\begin{aligned} \nabla \times \mathbf{E} &= -\mathbf{M} - j\omega\mu\mathbf{H} \\ \nabla \times \mathbf{H} &= \mathbf{J} + j\omega\varepsilon\mathbf{E} \end{aligned} \tag{1}$$

where $\mathbf{E}$ and $\mathbf{H}$ are the electric and magnetic fields, and $\mathbf{J}$ and $\mathbf{M}$ are the time-harmonic electric and magnetic current densities.

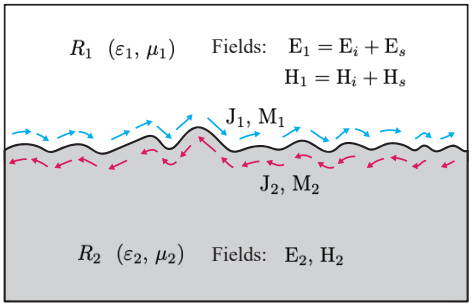

Fig. 2. An overview of the boundary element method. The space is divided into two regions by the scattering surface, and fictitious currents are introduced on both sides of the surface.

Solving a scattering problem using BEM involves dividing the space into two regions using the boundary of the scattering object and introducing fictitious surface currents on both sides of the object boundary, as illustrated in Fig. 2. Maxwell's equations characterize the relationships between the fields and currents in each region, and the two regions are connected through boundary conditions. These relationships and constraints give rise to equations that determine the surface currents, effectively converting the job of computing field values in a 3D space into the simpler task of solving for current densities on a 2D boundary.

Source-Field Relationships. The total fields in the regions R_1, R_2 are denoted by $(\mathbf{E}_1, \mathbf{H}_1)$ and $(\mathbf{E}_2, \mathbf{H}_2)$, and the incident fields in R_1 are given by $\mathbf{E}_i$ and $\mathbf{H}_i$, as shown in Fig. 2. The fields $\mathbf{E}_s$ and $\mathbf{H}_s$ are the introduced scattered fields, which propagate outward from the scattering surface and satisfy the relationships

$$\mathbf{E}_1 = \mathbf{E}_i + \mathbf{E}_s; \qquad \mathbf{H}_1 = \mathbf{H}_i + \mathbf{H}_s \tag{2}$$

The currents $(\mathbf{J}_1, \mathbf{M}_1)$ and $(\mathbf{J}_2, \mathbf{M}_2)$ in Fig. 2 can be considered the sources that radiate the fields $(\mathbf{E}_s, \mathbf{H}_s)$ and $(\mathbf{E}_2, \mathbf{H}_2)$. The fields $\mathbf{E}_s$ and $\mathbf{H}_s$ can be seen as generated by the currents $\mathbf{J}_1, \mathbf{M}_1$, radiating in a homogeneous medium with constitutive parameters (ε_1, μ_1). $\mathbf{E}_2$ and $\mathbf{H}_2$ can be seen as generated by the currents $\mathbf{J}_2, \mathbf{M}_2$, radiating in a homogeneous medium with constitutive parameters (ε_2, μ_2).

The source-field relationships for homogeneous media are derived from Maxwell's equations and are given by

$$\begin{aligned} \mathbf{E}(\mathbf{r}) &= -j\omega\mu(\mathcal{L}\mathbf{J})(\mathbf{r}) - (\mathcal{K}\mathbf{M})(\mathbf{r}) \\ \mathbf{H}(\mathbf{r}) &= -j\omega\varepsilon(\mathcal{L}\mathbf{M})(\mathbf{r}) + (\mathcal{K}\mathbf{J})(\mathbf{r}) \end{aligned} \tag{3}$$

where

$$\begin{aligned} (\mathcal{L}\mathbf{X})(\mathbf{r}) &= [1 + \frac{1}{k^2}\nabla\nabla\cdot] \int_V G(\mathbf{r}, \mathbf{r}')\mathbf{X}(\mathbf{r}')d\mathbf{r}' \\ (\mathcal{K}\mathbf{X})(\mathbf{r}) &= \nabla \times \int_V G(\mathbf{r}, \mathbf{r}')\mathbf{X}(\mathbf{r}')d\mathbf{r}' \end{aligned} \tag{4}$$

Here the domain of integration V is the homogeneous medium, and when $\mathcal{L}, \mathcal{K}$ are applied to surface current densities, V reduces to the boundary of the medium—in our case, the scattering surface. k is the wavenumber, given by $k = 2\pi\eta/\lambda$, where η is the refractive index of the medium and λ is the source field's wavelength in vacuum. $G(\mathbf{r}, \mathbf{r}')$ is the 3D Green's function for the scalar Helmholtz equation

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jkr}}{4\pi r} \qquad \text{where } r = |\mathbf{r} - \mathbf{r}'|. \tag{5}$$

Eq. 3 can be applied in both R_1 and R_2, giving rise to

$$\begin{aligned} \mathbf{E}_s(\mathbf{r}) &= -j\omega\mu_1(\mathcal{L}_1\mathbf{J}_1)(\mathbf{r}) - (\mathcal{K}_1\mathbf{M}_1)(\mathbf{r}) \\ \mathbf{H}_s(\mathbf{r}) &= -j\omega\varepsilon_1(\mathcal{L}_1\mathbf{M}_1)(\mathbf{r}) + (\mathcal{K}_1\mathbf{J}_1)(\mathbf{r}) \end{aligned} \tag{6}$$

and

$$\begin{aligned} \mathbf{E}_2(\mathbf{r}) &= -j\omega\mu_2(\mathcal{L}_2\mathbf{J}_2)(\mathbf{r}) - (\mathcal{K}_2\mathbf{M}_2)(\mathbf{r}) \\ \mathbf{H}_2(\mathbf{r}) &= -j\omega\varepsilon_2(\mathcal{L}_2\mathbf{M}_2)(\mathbf{r}) + (\mathcal{K}_2\mathbf{J}_2)(\mathbf{r}) \end{aligned} \tag{7}$$

Since $\mathcal{L}, \mathcal{K}$ assume different forms in different media, we use the subscripts 1 and 2 to distinguish between these forms.

Boundary Conditions. The tangential components of the electric and magnetic fields need to be continuous across the boundary surface, giving us

$$\mathbf{n} \times (\mathbf{E}_1 - \mathbf{E}_2) = \mathbf{0}; \qquad \mathbf{n} \times (\mathbf{H}_1 - \mathbf{H}_2) = \mathbf{0} \tag{8}$$

In addition, the net current densities on the boundary are required to be zero, meaning that

$$\mathbf{J} = \mathbf{J}_1 = -\mathbf{J}_2; \qquad \mathbf{M} = \mathbf{M}_1 = -\mathbf{M}_2 \tag{9}$$

Integral Equations. Combining Eq. 6, 7, 8 and 9 gives us

$$\begin{aligned} [j\omega\mu_1(\mathcal{L}_1\mathbf{J})(\mathbf{r}) + j\omega\mu_2(\mathcal{L}_2\mathbf{J})(\mathbf{r}) + (\mathcal{K}_1\mathbf{M})(\mathbf{r}) + (\mathcal{K}_2\mathbf{M})(\mathbf{r})]_{\tan} \\ = [\mathbf{E}_i(\mathbf{r})]_{\tan} \\ [(\mathcal{K}_1\mathbf{J})(\mathbf{r}) + (\mathcal{K}_2\mathbf{J})(\mathbf{r}) - j\omega\varepsilon_1(\mathcal{L}_1\mathbf{M})(\mathbf{r}) - j\omega\varepsilon_2(\mathcal{L}_2\mathbf{M})(\mathbf{r})]_{\tan} \\ = -[\mathbf{H}_i(\mathbf{r})]_{\tan} \end{aligned} \tag{10}$$

which hold at every point $\mathbf{r}$ on the surface. Here, the "tan" notation refers to the component of the field that is tangent to surface at $\mathbf{r}$. These equations are the PMCHWT [Gibson 2021] electric field integral equation (EFIE) and magnetic field integral equation (MFIE).

Solving for Current Densities. To solve for the current densities $\mathbf{J}, \mathbf{M}$, we discretize the object surface into a collection of basis elements and locally assign low-order basis functions to each basis element. The surface currents can then be written as

$$\mathbf{J}(\mathbf{r}) = \sum_{m=1}^{N} I_{\mathrm{J}m}\mathbf{f}_m(\mathbf{r}); \qquad \mathbf{M}(\mathbf{r}) = \sum_{n=1}^{N} I_{\mathrm{M}n}\mathbf{f}_n(\mathbf{r}) \tag{11}$$

where N is the total number of basis functions. To solve for the unknown coefficients in Eq. 11, we apply a Galerkin-type method and combine the EFIE and MFIE into a linear system:

$$\begin{bmatrix} A_{\mathrm{EJ}} & A_{\mathrm{EM}} \\ A_{\mathrm{HJ}} & A_{\mathrm{HM}} \end{bmatrix} \begin{bmatrix} I_{\mathrm{J}} \\ I_{\mathrm{M}} \end{bmatrix} = \begin{bmatrix} V_{\mathrm{E}} \\ V_{\mathrm{H}} \end{bmatrix} \tag{12}$$

where

$$A_{\mathrm{EJ}}^{mn} = \int_S \mathbf{f}_m(\mathbf{r}) \cdot [j\omega\mu_1(\mathcal{L}_1\mathbf{f}_n)(\mathbf{r}) + j\omega\mu_2(\mathcal{L}_2\mathbf{f}_n)(\mathbf{r})]d\mathbf{r} \tag{13}$$

$$A_{\mathrm{EM}}^{mn} = \int_S \mathbf{f}_m(\mathbf{r}) \cdot [(\mathcal{K}_1\mathbf{f}_n)(\mathbf{r}) + (\mathcal{K}_2\mathbf{f}_n)(\mathbf{r})]d\mathbf{r} \tag{14}$$

$$A_{\mathrm{HJ}}^{mn} = \int_S \mathbf{f}_m(\mathbf{r}) \cdot [(\mathcal{K}_1\mathbf{f}_n)(\mathbf{r}) + (\mathcal{K}_2\mathbf{f}_n)(\mathbf{r})]d\mathbf{r} \tag{15}$$

$$A_{\mathrm{HM}}^{mn} = -\int_S \mathbf{f}_m(\mathbf{r}) \cdot [j\omega\varepsilon_1(\mathcal{L}_1\mathbf{f}_n)(\mathbf{r}) + j\omega\varepsilon_2(\mathcal{L}_2\mathbf{f}_n)(\mathbf{r})]d\mathbf{r} \tag{16}$$

and

$$V_{\mathrm{E}}^{m} = \int_S \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}_i(\mathbf{r})d\mathbf{r} \tag{17}$$

$$V_{\mathrm{H}}^{m} = -\int_S \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{H}_i(\mathbf{r})d\mathbf{r} \tag{18}$$

One might have noticed that the EFIE and MFIE assume very similar forms, while we have applied an extra factor of -1 to both sides of the MFIE, resulting in a factor of -1 in Eq. 18. We use this specific form of the linear system thanks to its symmetry.

In fact, observing Eq. 13 and 16 as well as the $\mathcal{L}$ operator given in Eq. 4, we notice that

$$A_{\mathrm{EJ}} = A_{\mathrm{EJ}}^T; \qquad A_{\mathrm{HM}} = A_{\mathrm{HM}}^T \tag{19}$$

Inspecting Eq. 14 and 15 and the $\mathcal{K}$ operator in Eq. 4 gives us

$$A_{\mathrm{EM}} = A_{\mathrm{EM}}^T; \qquad A_{\mathrm{HJ}} = A_{\mathrm{HJ}}^T; \qquad A_{\mathrm{EM}} = A_{\mathrm{HJ}} \tag{20}$$

Eq. 19 and 20 show that our linear system in Eq. 12 is symmetric, allowing us to evaluate fewer matrix elements and use an iterative solver that requires less memory. After solving for the surface current densities, we can again apply Eq. 6 to compute the scattered fields that propagate outward from the scattering surface.

3.2 Rough Surface Scattering: The Specifics

We now introduce our specific approach to represent and discretize the surfaces involved in our simulations.

Rough Surface Samples. The general illustration in Fig. 2 represents the scattering surface as infinitely large, dividing the space into two half spaces. Each simulation, however, needs to be limited to a finitely large computational domain. We make this possible by our specific choice of incident fields.

Each finitely large surface sample is represented as a 2D height field. For a surface sample with size $L_x \times L_y$, we choose a step size d and further define the sequences $\{x_s\}, \{y_t\}$ such that

$$\begin{aligned} x_s &= s \cdot d \qquad s = 0, 1, ..., N_x \\ y_t &= t \cdot d \qquad t = 0, 1, ..., N_y \end{aligned} \tag{21}$$

where $N_x = L_x/d$ and $N_y = L_y/d$. For all pairs of indices (s, t), height data are provided as $z_{s,t} = h(x_s, y_t)$ from some height field function h. See Fig. 3 for an illustration.

Note that the height variations on the surface are in very small scale—typically within a few microns (a few wavelengths of visible light), and the surface is considered globally flat, with the positive $\mathbf{z}$ direction as its normal on the macro-scale.

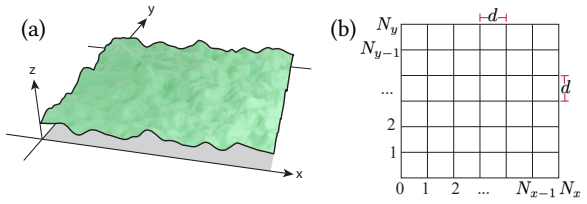

Fig. 3. A rough surface sample represented as a 2D surface height field, and discretized into quadrilateral basis elements.

Basis Elements and Functions. Since our rough surface is represented as a 2D height field defined on an xy-grid, we take advantage of the grid structure and use quadrilateral basis elements to discretize the surface. As shown in Fig. 3 (b), the surface is tiled into $N_x \times N_y$ elements, each with an *xy footprint* of size $d \times d$. Despite the "8-10 steps per wavelength" convention for basis element size, the xy footprints of our quadrilateral elements may need to be smaller in some cases, as the rough surface contains nonzero slopes that lengthen the actual basis elements. In most of our simulations, we normally use $d \approx \lambda/16$ for our minimum wavelength.

We adopt the first-order vector basis functions proposed in [Djordjevic and Notaros 2004]. Each basis element is parametrized with respect to two variables u, v, both in the range of $[-1, 1]$. The shape of the basis element is represented analytically by a bilinear function $\mathbf{r}(u, v)$, which for the element with index (s, t) is given by

$$\begin{aligned}\mathbf{r}(u,v) = &\frac{(1-u)(1-v)}{4}\mathbf{p}_{s-1,t-1} + \frac{(1-u)(1+v)}{4}\mathbf{p}_{s-1,t} \\ &+ \frac{(1+u)(1-v)}{4}\mathbf{p}_{s,t-1} + \frac{(1+u)(1+v)}{4}\mathbf{p}_{s,t}\end{aligned} \tag{22}$$

where $\mathbf{p}_{s,t} = (x_s, y_t, z_{s,t})$. Four basis functions are defined on each quadrilateral element in the following form:

$$\begin{aligned}\mathbf{f}_1(u,v) &= \frac{(1-u)}{J(u,v)}\frac{\partial \mathbf{r}(u,v)}{\partial u}; \quad \mathbf{f}_2(u,v) = \frac{(1+u)}{J(u,v)}\frac{\partial \mathbf{r}(u,v)}{\partial u}; \\ \mathbf{f}_3(u,v) &= \frac{(1-v)}{J(u,v)}\frac{\partial \mathbf{r}(u,v)}{\partial v}; \quad \mathbf{f}_4(u,v) = \frac{(1+v)}{J(u,v)}\frac{\partial \mathbf{r}(u,v)}{\partial v}\end{aligned} \tag{23}$$

where the Jacobian $J(u, v)$ is defined as:

$$J(u,v) = \left| \frac{\partial \mathbf{r}(u,v)}{\partial u} \times \frac{\partial \mathbf{r}(u,v)}{\partial v} \right| \tag{24}$$

As illustrated in Fig. 4, the basis function $\mathbf{f}_i$ represents the flux of the surface currents across the edge e_i of the basis element, for $i = 1, 2, 3, 4$. It can be shown that among the four local basis functions, $\mathbf{f}_i$ is the only basis function with nonzero flux of the currents across the edge e_i. To ensure continuity of the flux, the two basis functions related to each edge (defined on the two neighboring elements that share this edge) share the same coefficient.

Gaussian Beam Incidence. Since we limit our simulation domain to a finite area, we want to similarly constrain the size of the incident field. This way, the current densities induced on the surface can be considered as only having nonzero magnitudes within a finite region, which is usually slightly larger than the illuminated area. Thus, Gaussian beams are great candidates for the incident fields.

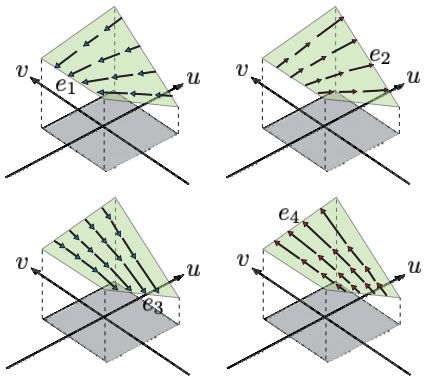

Fig. 4. Four basis functions are defined on each quadrilateral basis element, each associated with the flux of the surface currents across one edge.

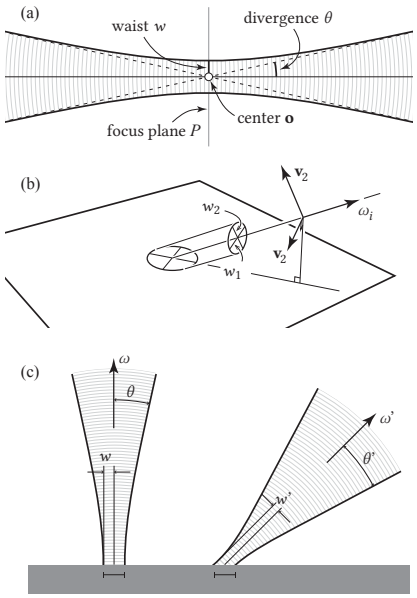

Fig. 5. Illustration of Gaussian beams. (a) A Gaussian beam and its focus plane, center, waist, and divergence. (b) A Gaussian beam that illuminates a surface at some incident angle, with different waists w_1, w_2 along the two transverse directions $\mathbf{v}_1, \mathbf{v}_2$. (c) Gaussian beams incident on the surface from different directions have the same illumination area, as the beam waists are chosen according to incident directions.

A Gaussian beam is an electromagnetic wave whose amplitude is a 2D Gaussian function in the plane perpendicular to its propagation direction [Paschotta 2008]. As illustrated in Fig. 5 (a), a Gaussian beam is characterized by a focus plane P, a center $\mathbf{o}$, and a beam waist w. At each point $\mathbf{q}$ on the focus plane P, the field amplitude is proportional to e^{-r^2/w^2}, where $r = |\mathbf{q} - \mathbf{o}|$. Thus, at points whose distance to $\mathbf{o}$ is greater than $2.5w$, the field intensity (squared amplitude) can be considered negligible.

A Gaussian beam can be used to model nearly collimated illumination, while it also has a divergence θ, as shown in Fig. 5 (a), which

is inversely proportional to w. For a beam with wavelength λ that propagates in a medium with refractive index η, we have:

$$\theta = \frac{\lambda}{\pi \eta w} \tag{25}$$

Some Gaussian beams have different waists along two transverse directions, giving them an elliptical cross section on the focus plane and different divergences on the two axes. In our work, we use elliptical Gaussian beams for modeling non-normal incidence. As shown in Fig. 5 (b), to model an incident direction ω_i (which points away from the surface), a Gaussian beam that propagates along $-\omega_i$ can be used. The vector $\mathbf{v}_1$ is defined to be perpendicular to the surface normal and ω_i, and $\mathbf{v}_2$ is perpendicular to ω_i and $\mathbf{v}_1$. We use a Gaussian beam with different waists w_1, w_2 along the two transverse directions $\mathbf{v}_1, \mathbf{v}_2$

$$w_1 = w; \qquad w_2 = w \cos\theta_i \tag{26}$$

where θ_i is the zenith angle characterizing ω_i and w is the waist of the Gaussian beam normally incident on the surface. Since our simulations are used for deriving BRDFs, we may model illumination from many incident directions. For the same surface, we want similar illumination area from Gaussian beams for all the directions, and the $\cos\theta_i$ factor in Eq. 26 guarantees this, as illustrated in Fig. 5 (c). We refer to w as the *primary waist*.

Lastly, the field generated by a Gaussian beam has an exact form as a linear combination of plane waves, while it also has a paraxial approximation that makes the field much easier to evaluate. We find the paraxial approximation sufficiently accurate in our context, and use the general form of paraxial Gaussian beams in [Kiselev and Plachenov 2019] to model circular and elliptical Gaussian beams.

4 IMPLEMENTATION AND ACCELERATION

Without any acceleration, full-wave simulations in 3D are computationally expensive, even for very small problems. In this section, we discuss our acceleration method and implementation details.

4.1 The Adaptive Integral Method

Our 3D simulations typically result in very large linear systems, so explicitly computing and storing every element in the matrix A in Eq. 12 will not be feasible. The symmetric linear system is instead solved iteratively, and the matrix-vector multiplication step is accelerated using the adaptive integral method (AIM).

Approximating Matrix Elements. AIM was originally proposed by Bleszynski et. al. [1996], and is a good match for our simulations thanks to its low computational complexity for our problems and its relatively simple implementation. In AIM, the effects of each basis function are approximated by a set of point sources on a global 3D grid, and the effects of all the basis functions are propagated together, with the computations accelerated using FFT.

As detailed in our supplemental document, we can expand the dot and cross product terms as well as the $\mathcal{L}$ and $\mathcal{K}$ operators in Eq. 13–16 to rewrite every element in A_{EJ}, A_{EM}, A_{HJ}, and A_{HM} as the linear combination of some terms in the form of

$$\int_{\mathbf{f}_m} \int_{\mathbf{f}_n} \psi_m(\mathbf{r}) g(\mathbf{r}-\mathbf{r}') \xi_n(\mathbf{r}') d\mathbf{r}' d\mathbf{r} \tag{27}$$

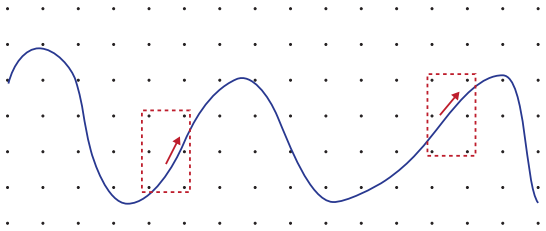

Fig. 6. A Cartesian grid surrounding the surface sample, illustrated in 2D. The red boxes indicate that each basis function can be approximated by a collection of nearby point sources. The Cartesian grid and point source approximation are established in 3D in our simulations.

where the scalar functions ψ_m, ξ_n come from the basis functions $\mathbf{f}_m, \mathbf{f}_n$—they could be the x, y, or z component of the vector basis functions, or could be related to the divergence of $\mathbf{f}_m, \mathbf{f}_n$. The function $g(\mathbf{r}-\mathbf{r}')$ is related to the 3D Green's functions. As implied from the way we represent it, g is shift-invariant, with its value only dependent on $\mathbf{r}-\mathbf{r}'$. See Eq. S.16–S.20 in our supplemental document for the exact expressions of each matrix element.

To approximate Eq. 27, we create a 3D Cartesian grid surrounding the surface, and replace the scalar functions ψ_m, ξ_n with linear combinations of Dirac Delta functions localized at nearby grid nodes:

$$\begin{aligned} \psi_m(\mathbf{r}) &\approx \hat{\psi}_m(\mathbf{r}) := \sum_{\mathbf{p}\in S_m} \Lambda_{m\mathbf{p}} \delta^3(\mathbf{r}-\mathbf{p}) \\ \xi_n(\mathbf{r}') &\approx \hat{\xi}_n(\mathbf{r}') := \sum_{\mathbf{q}\in S_n} \Lambda'_{n\mathbf{q}} \delta^3(\mathbf{r}'-\mathbf{q}) \end{aligned} \tag{28}$$

Here, S_m and S_n are the sets of grid points near the supports of ψ_m, ξ_n, as illustrated in Fig. 6. The coefficients $\Lambda_{m\mathbf{p}}, \Lambda'_{n\mathbf{q}}$ are computed by matching the field radiated by each basis function and that radiated by the collection of point sources, as described in [Yang and Yilmaz 2011] and discussed in our supplemental document. Substituting Eq. 28 into 27 reduces the integral to

$$\sum_{\mathbf{p}\in S_m} \sum_{\mathbf{q}\in S_n} \Lambda_{m\mathbf{p}} g(\mathbf{p}-\mathbf{q}) \Lambda'_{n\mathbf{q}} \tag{29}$$

Eq. 29 is a good approximation to Eq. 27 when the supports of ψ_m and ξ_n (equivalently, their respective quadrilaterals) are far apart.

Base and Correction Matrices. Based on Eq. 29, we define a set of *base approximation* matrices B_{EJ}, B_{EM}, B_{HJ}, and B_{HM} that approximate A_{EJ}, A_{EM}, A_{HJ} and A_{HM}. These base matrices are accurate for entries involving pairs of basis elements that are far apart. These matrices can each be written as the linear combination of some *convolution terms* in the form of $\Lambda_1 G \Lambda_2^T$, where Λ_1, Λ_2 are matrices that contain coefficients in the Dirac Delta function expansions, and G is a matrix that stores the g function values. Eq. S.26–S.27 in the supplemental document contains expressions for the B matrices.

The difference between the aforementioned A and B matrices can be considered sparse, because most entries in the B matrices, which involve far-apart basis elements, are very accurate. We exploit this sparseness by defining the *correction* matrices C_{EJ}, C_{EM}, C_{HJ}, and C_{HM} whose entries are only nonzero if their basis elements are sufficiently close to each other. The correction matrix elements are

defined as follows:

$$C_{\mathrm{X}}^{mn} = \begin{cases} A_{\mathrm{X}}^{mn} - B_{\mathrm{X}}^{mn} & d_{mn} \le d_{\mathrm{near}} \\ 0 & \text{otherwise} \end{cases} \quad \mathrm{X} \in \{\mathrm{EJ}, \mathrm{EM}, \mathrm{HJ}, \mathrm{HM}\} \tag{30}$$

where d_{mn} is the distance between the centers of basis functions m and n and d_{near} is a distance threshold chosen to control the approximation error. In summary, we can write

$$\begin{aligned} A_{\mathrm{EJ}} &\approx B_{\mathrm{EJ}} + C_{\mathrm{EJ}}; \quad & A_{\mathrm{EM}} &\approx B_{\mathrm{EM}} + C_{\mathrm{EM}}; \\ A_{\mathrm{HJ}} &\approx B_{\mathrm{HJ}} + C_{\mathrm{HJ}}; \quad & A_{\mathrm{HM}} &\approx B_{\mathrm{HM}} + C_{\mathrm{HM}} \end{aligned} \tag{31}$$

Fast Matrix-Vector Multiplication. Our approximation of the BEM matrices allows fast matrix-vector multiplication for an iterative solver. The sparse C matrices are computed before the iterative solving, and this step generally takes up a very small fraction of time in a simulation. In iterative solving, multiplying these sparse matrices with vectors is fast. Also, multiplying a vector by one of the B matrices reduces to multiplying that vector by a few convolution terms, which involves computing

$$y_1 = \Lambda_2^T x; \quad y_2 = G y_1; \quad y_3 = \Lambda_1 y_2 \tag{32}$$

for different combinations of x, Λ_1, G, and Λ_2. Here, Λ_1, Λ_2 are $N_f \times N_p$ coefficient matrices, where N_f is the total number of basis functions in the computation domain, and N_p is the total number of grid points in the Cartesian grid. In practice, we use 48 Dirac Delta functions to approximate each basis function, so there are only 48 nonzero elements in each row of Λ_1, Λ_2, making them very sparse. Thus, the steps $y_1 = \Lambda_2^T x$ and $y_3 = \Lambda_1 y_2$ are fast.

The vectors y_1, y_2 both represent discrete signals on the 3D grid, and the matrix G represents the shift-invariant function g. The computation of $y_2 = G y_1$ can therefore be accelerated using the discrete Fourier transform, since this multiplication is equivalent to convolving g with y_1:

$$y_2 = \mathcal{F}^{-1}\{\mathcal{F}(g)\mathcal{F}(y_1)\} \tag{33}$$

Therefore, multiplying the B matrices with vectors does not require explicitly computing or storing any matrix elements, and can be done efficiently using existing FFT libraries. On a surface whose lateral dimensions are large relative to the height variations, the number of grid points in the Cartesian grid is proportional to the number of basis elements on the surface. Thus, the computational complexity of a matrix-vector multiplication step is $O(N \log N)$ for a surface with N basis elements. We use a MINRES solver for complex symmetric matrices [Choi 2013] to solve the linear system.

4.2 GPU-Accelerated Iterative Solving

The AIM formulation moves the computational bottlenecks into the FFT and sparse matrix operations, enabling an efficient implementation on the GPU. We use the `cuFFT` library to compute the Fourier transforms involved in matrix-vector multiplications with the B matrices, and use the `cuSPARSE` sparse matrix library for multiplications involving the C matrices.

To discuss our implementation choices, we first note that computing the matrix-vector product

$$y = \begin{bmatrix} A_{\mathrm{EJ}} & A_{\mathrm{EM}} \\ A_{\mathrm{HJ}} & A_{\mathrm{HM}} \end{bmatrix} \begin{bmatrix} x_{\mathrm{J}} \\ x_{\mathrm{M}} \end{bmatrix} \tag{34}$$

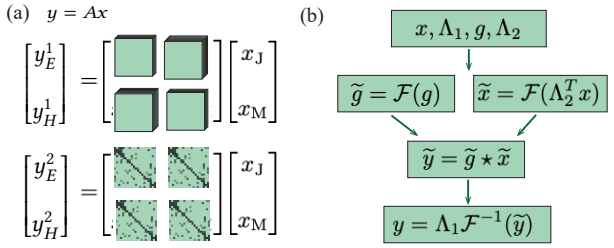

Fig. 7. The matrix-vector multiplication process in our framework. (a) The upper half represents the base approximation (B) matrices and the lower half represents the sparse correction (C) matrices. (b) A pipeline for multiplying a convolution term $\Lambda_1 G \Lambda_2^T$ to a vector.

can be divided into four subtasks (using Eq. 31 here):

$$\begin{aligned} y_{\mathrm{EJ}} &= B_{\mathrm{EJ}} x_{\mathrm{J}} + C_{\mathrm{EJ}} x_{\mathrm{J}}; \quad & y_{\mathrm{EM}} &= B_{\mathrm{EM}} x_{\mathrm{M}} + C_{\mathrm{EM}} x_{\mathrm{M}}; \\ y_{\mathrm{HJ}} &= B_{\mathrm{HJ}} x_{\mathrm{J}} + C_{\mathrm{HJ}} x_{\mathrm{J}}; \quad & y_{\mathrm{HM}} &= B_{\mathrm{HM}} x_{\mathrm{M}} + C_{\mathrm{HM}} x_{\mathrm{M}} \end{aligned} \tag{35}$$

Eq. S.26–S.27 in our supplemental document reveal that B_{EJ} and B_{HM} can each be written as the sum of 8 convolution terms, while B_{EM} and B_{HJ} can each be written as the sum of 12 convolution terms. Multiplying a convolution term to a vector can be done using the pipeline in Fig. 7 (b), which involves 3 calls to the FFT (or IFFT) routine. This pipeline needs to be executed 40 times in each MINRES iteration, and instead of invoking the FFT routines 120 times on one device, we use different matrix-vector multiplication modules for small and large problems, avoiding repeated computations as much as possible. Our small simulations require using 1 GPU, while large simulations are implemented to run on 4 GPUs.

Small-Scale Simulations. Multiplying the B matrices to vectors requires computing and storing the Fourier transformed Green's function values (i.e. the G matrices), as well as performing many forward and backward FFTs, possibly storing intermediate results. The relevant data are stored in arrays with size proportional to the number of grid points in the 3D grid, and the `cuFFT` library allocates some extra memory as the workspace for FFT computations. For small simulations (e.g. $12\mu m \times 12\mu m$), the required data arrays are typically well under 1GB. With the available memory on a GPU (e.g. 24 GB), we can allocate one array for each shift-invariant function g_i and compute the discrete Fourier transforms of these function values, so the data $\mathcal{F}(g_i)$ are stored for use in each MINRES iteration.

A few more data arrays are used for storing intermediate results from the FFT steps. Since computations related to the four matrix blocks are all performed on the same GPU, very few repeated computations are needed. Moreover, in small-scale problems, the sparse C matrices typically occupy less than 5 GB of memory, and can easily fit into the single GPU used for simulation.

Large-Scale Simulations. In large-scale simulations (e.g. $24\mu m \times 24\mu m$), the FFT-related data arrays become large, and one such array may occupy at least 2 GB of memory. The number of nonzero elements in the C matrices, which is roughly proportional to the number of basis elements, also grows significantly with the problem size. For a $24\mu m \times 24\mu m$ simulation with 960×960 basis elements, the memory required to store all the nonzero elements in

$C_{\mathrm{EJ}}, C_{\mathrm{EM}}, C_{\mathrm{HJ}}, C_{\mathrm{HM}}$ as complex floating-point numbers (plus the row and column indices of nonzero elements) is around 20 GB.

Due to the limited memory on one GPU, we divide the computations across 4 GPUs, each used for computing the matrix-vector multiplications associated with one matrix block (see Eq. 35). Up to 6 larger FFT-related arrays are allocated on each GPU, and they can store the Fourier transformed Green's function values and intermediate results, or serve as the FFT workspace. On each GPU, around 5GB of memory is used to store the C matrices.

Compared to small-scale simulations, more repeated computations are needed. Still, our matrix-vector multiplication step that uses 4 GPUs is fast. In $24\mu m \times 24\mu m$ simulations, one MINRES iteration takes well under 1 second for most surfaces.

For both versions of our implementation, the MINRES solver is executed on the host CPU, while matrix-vector products $y = Ax$ are computed on GPU(s). Since the vectors x, y contain very minimal data (about 30 MB for our largest domain size), little time is spent on data transfer in each iteration.

4.3 FFT-Accelerated Scattered Field Evaluation

After solving the BEM linear system using our GPU-accelerated MINRES solver, we can use the surface current densities $\mathbf{J}, \mathbf{M}$ to compute the scattered field that propagates from the surface. For the specific purpose of computing a surface BRDF value $f_r(\omega_i, \omega_o)$, we need to evaluate the scattered field along ω_o in the far field region (at points very far away from the surface itself). In the far field region, the scattered field at $\mathbf{r} = r\hat{\mathbf{r}}$ (where $kr \gg 1$) satisfies

$$\mathbf{E}_s(\mathbf{r}) \approx \mathbf{E}(\hat{\mathbf{r}})\frac{e^{-jkr}}{r}; \qquad \mathbf{H}_s(\mathbf{r}) \approx \mathbf{H}(\hat{\mathbf{r}})\frac{e^{-jkr}}{r} \tag{36}$$

The exact forms of $\mathbf{E}(\hat{\mathbf{r}})$ and $\mathbf{H}(\hat{\mathbf{r}})$ can be derived from Eq. 6, and some approximations are applied in the derivation given $kr \gg 1$ (see Section 3 in the supplemental document). The Cartesian components of these far field quantities, denoted as $E_x(\hat{\mathbf{r}})$, $E_y(\hat{\mathbf{r}})$, $E_z(\hat{\mathbf{r}})$, $H_x(\hat{\mathbf{r}})$, $H_y(\hat{\mathbf{r}})$, and $H_z(\hat{\mathbf{r}})$, can each be written as a linear combination of the 6 following terms:

$$\begin{aligned} F_1(\hat{\mathbf{r}}) &= \int_V J_x(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}'; \quad F_2(\hat{\mathbf{r}}) = \int_V J_y(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}'; \\ F_3(\hat{\mathbf{r}}) &= \int_V J_z(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}'; \quad F_4(\hat{\mathbf{r}}) = \int_V M_x(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}'; \\ F_5(\hat{\mathbf{r}}) &= \int_V M_y(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}'; \quad F_6(\hat{\mathbf{r}}) = \int_V M_z(\mathbf{r}')e^{jk\mathbf{r}'\cdot\hat{\mathbf{r}}}d\mathbf{r}' \end{aligned} \tag{37}$$

where V is defined and explained after Eq. 4. Recall that the currents $\mathbf{J}, \mathbf{M}$ are represented using basis functions, and each Cartesian component of these basis functions is approximated on the 3D grid created for accelerating our linear solve (recall the Λ matrices). We reuse this approximation to reduce each expression in Eq. 37 into

$$F_i(\hat{\mathbf{r}}) = \sum_{\mathbf{p}\in S} h_i(\mathbf{p})e^{jk\mathbf{p}\cdot\hat{\mathbf{r}}} = \sum_{\mathbf{p}\in S} h_i(\mathbf{p})e^{j\mathbf{p}\cdot k\hat{\mathbf{r}}} \quad i = 1, 2, 3, 4, 5, 6 \tag{38}$$

where S is the set of all the grid points in the 3D grid and h_i stands for the relevant component of J or M.

Eq. 38 reveals that $F_i(\hat{\mathbf{r}})$ is the Fourier component of $h_i(\mathbf{p})$ for the spatial frequency $-k\hat{\mathbf{r}}$, so we can compute $F_i(\hat{\mathbf{r}})$ for a dense grid of directions by computing the FFT of the grid $h_i(\mathbf{p})$. The required spatial frequencies are not on the FFT grid but can be interpolated; we add zero padding prior to the FFT step, to ensure enough resolution in the frequency domain for the trilinear interpolation to be sufficiently accurate.

5 HIGH RESOLUTION BRDF GENERATION

The goal of conducting our full-wave simulations is to characterize the appearance of surfaces by computing BRDF values $f_r(\omega_i, \omega_o)$ for a number of incident and outgoing directions. The BRDF is a smooth function of both ω_i and ω_o, and the sampling rate required to resolve the high resolution details in the BRDF increases with the size of the illuminated area on the surface. This appears to make the cost of computing a full BRDF increase rapidly with the illuminated area, since more incident directions need to be computed as the simulated area grows, but in this section we propose an approach to reduce this cost by taking advantage of linear superposition to build the required results from a collection of smaller simulations.

In particular, we replace a single simulation using a large-area incident field with simulations using a grid of N^2 small-waist Gaussian beams to illuminate the same surface area. The large-area incident field can be approximated by a linear combination of the small-waist beams: even though the small-waist beams have a wider divergence, they interfere to produce a narrow-divergence, large-area field. Linearity implies that the far-field scattering pattern that would result from the large-area incident field can also be approximated by a linear combination of the results from the small-waist simulations. Each small simulation costs much less than a large-area simulation, and multiple small simulations can be done separately, even in parallel, leading to substantial savings.

This is useful by itself because it allows larger areas to be covered than what would fit in GPU memory, but the small beams can also be combined with complex-valued scale factors to produce a total field that is centered in any direction within the divergence of the small-waist beams. We call this process *beam steering* and present a method for computing BRDFs on dense grids of incident directions without doing large simulations for every required incident direction.

5.1 Basic and Derived Incident Directions

We first consider N^2 Gaussian beams propagating in the direction $\mathbf{u}$ and focused at a 2D grid of N^2 points on a receiving plane. Multiplying a complex-valued scale factor to the field produced by each beam and summing up the fields will result in some total field. As demonstrated in Fig. 8, different scale factors applied to the same collection of Gaussian beams can result in total fields with different net propagation directions.

Assuming the N^2 beams are focused at $\mathbf{p}_{st} = (x_s, y_t, 0)$ in space, for $1 \le s, t \le N$, where $\{x_s\}$ and $\{y_t\}$ are equally spaced sequences, we find that given the basic direction $\mathbf{u}$ and a desired incident direction ω_i (pointing away from the surface), the complex scale factor to multiply to the field from beam (s, t) should be given as

$$a_{st} = e^{jk\mathbf{p}_{st}\cdot\omega_i} \tag{39}$$

In other words, the factor a_{st} comes from the field value of a plane wave propagating in the direction of $-\omega_i$. In practice, the Gaussian beams can only be effectively summed into a total field that propagates in the desired direction if $-\omega_i$ is close to $\mathbf{u}$. When

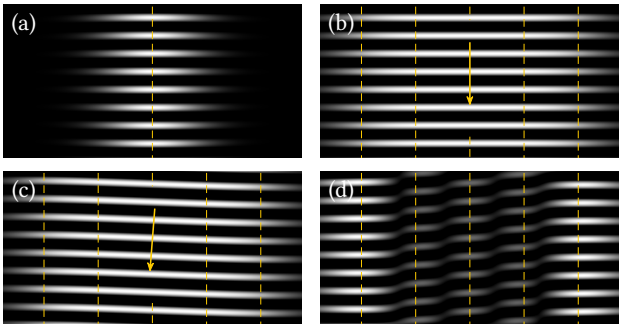

Fig. 8. Summing the fields from multiple Gaussian beams. (a) Wavefronts from a single Gaussian beam. (b) Summing Gaussian beams propagating in the same direction to obtain a wider field. (c) Summing multiple beams to obtain a field with a slightly different net propagation direction. (d) A failed attempt to construct a total field whose propagation direction forms a large angle with the propagation direction of the individual beams; destructive interference between neighboring Gaussian beams causes the net propagation direction to become undefined.

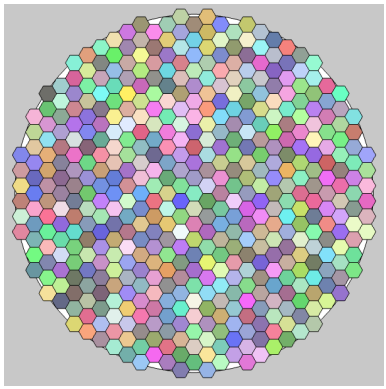

Fig. 9. An example of a collection of basic incident directions, visualized with hexagons that cover the unit circle which represents the projected hemisphere. The center of each hexagon locates each chosen basic incident direction. The region in the unit circle not covered by any hexagon represents incident directions very close to grazing that we fail to model.

the angle between $-\omega_i$ and $\mathbf{u}$ approaches the divergence of the small beams, aliasing artifacts begin to appear. An example is shown in Fig. 8 (d). We refer to the possible propagation directions $-\omega_i$ as the *derived* directions from the *basic* direction $\mathbf{u}$.

In our framework, we decide on a primary waist w and choose a collection of basic incident directions. In general, a smaller primary waist allows a larger range of incident directions to be derived from each basic direction, so fewer basic incident directions are required. On the other hand, a larger primary waist leads to low angular divergence in each Gaussian beam and even lower divergence in the synthesized total fields.

The hexagon grid covering the projected hemisphere in Fig. 9 is an example of dividing the hemisphere of all possible incident directions into territories belonging to a finite collection of basic directions. Interestingly, Eq. 25 and 26 implies that a basic incident direction with a larger incident angle θ_i corresponds to a larger range of derived directions, roughly by a factor of $1/\cos\theta_i$. This factor neatly cancels with the cosine factor involved in representing directions with the *projected* hemisphere, giving rise to the equal-sized hexagons used for illustration. In practice, we confirmed that basic incident directions closer to grazing "steer" to larger ranges of derived directions, so the hexagon grid is a faithful representation of the territories of each basic direction.

5.2 Individual Simulations and Synthesized Results

Given a chosen primary waist w, we can compute the sequences $\{x_s\}, \{y_t\}$ which locate the Gaussian beams used in the small simulations. Specifically, the spacing $\Delta_x = x_{s+1} - x_s, \Delta_y = y_{t+1} - y_t$ are usually chosen as $\Delta_x = \Delta_y = w$. We find this spacing ideal for synthesizing total incident fields with low angular divergence and near-constant intensity over a large area on the surface. In each small simulation, the local origin of coordinates is at the center of the single incident Gaussian beam. The Gaussian beams have field values of identical amplitudes and phases at the local origins in all the small simulations (i.e. identical fields are used). This way, when synthesizing the total scattered field from the large-area surface, the scale factors a_{st} (Eq. 39) can be directly applied to the scattered field quantities computed in each small simulation.

Since the collection of small simulations may be used to replace multiple large-area simulations featuring different incident directions, different sets of complex-valued scale factors a_{st} (Eq. 39) may be used. In addition to the scale factors a_{st}, another phase shift associated with the outgoing direction needs to be applied. This is because we need to convert from local coordinates to global coordinates, as simulation (s,t) is centered at $\mathbf{p}_{st} = (x_s, y_t, 0)$, rather than the origin, in global coordinates. According to Eq. 37, the scattered field values from simulation (s,t) for the outgoing direction ω_o need to be multiplied by a factor of

$$b_{st} = e^{jk\mathbf{p}_{st}\cdot\omega_o} \tag{40}$$

Combining with Eq. 39, the scattered fields in the far field region corresponding to the pair of directions (ω_i, ω_o) are given by:

$$\begin{aligned}\mathbf{E}(\omega_i,\omega_o) &= \sum_{s=1}^{n}\sum_{t=1}^{n} e^{jk\mathbf{p}_{st}\cdot(\omega_i+\omega_o)}\mathbf{E}_{st}(\omega_o)\\ \mathbf{H}(\omega_i,\omega_o) &= \sum_{s=1}^{n}\sum_{t=1}^{n} e^{jk\mathbf{p}_{st}\cdot(\omega_i+\omega_o)}\mathbf{H}_{st}(\omega_o)\end{aligned} \tag{41}$$

where $\mathbf{E}, \mathbf{H}$ refer to the far field quantities only associated with directions (without the e^{-jkr}/r term).

Lastly, we can compute the surface BRDF value as

$$f_r(\omega_i,\omega_o) = \frac{\frac{1}{2}|\mathbf{E}(\omega_i,\omega_o)\times\mathbf{H}(\omega_i,\omega_o)^*|}{\Phi_i\cos\theta_r} \tag{42}$$

where the incident power Φ_i is computed by integrating the incident irradiance over the surface:

$$\Phi_i = \frac{1}{2}\int_S \left|[\mathbf{E}_i(\mathbf{r}')\times\mathbf{H}_i(\mathbf{r}')^*]\cdot\mathbf{n}\right| d\mathbf{r}' \tag{43}$$

where $\mathbf{n}$ is the surface normal at the macro scale (+z). Note that Eq. 42 and Eq. 43 can also be applied in single simulations, where Φ_i is computed from a single Gaussian beam.

Given any collection of direction pairs (ω_i, ω_o), we can evaluate the corresponding BRDF values by determining the set of relevant basic incident directions that will derive all the queried directions ω_i and performing small simulations correspondingly. The BRDF values for each direction pair can be computed using Eq. 41–43.

6 RESULTS

In this section, we present our simulation results and demonstrate the validity and usefulness of our simulator in various ways. We also compare our simulations to previous methods, and show a comparison between our result and experimental measurement.

6.1 Demonstration of Consistency

To first demonstrate the validity of our simulation, we show that our results are consistent as we compute BRDFs for the same surface represented by height fields of increasing resolution. We use two surfaces for consistency testing—one features a 1D diffraction grating and the other contains a collection of corner reflectors, as shown in Fig. 10. Both surfaces are $16\mu m \times 16\mu m$ and are represented with 2D height fields of a few different resolutions. The surface material is defined by the wavelength-dependent refractive index of aluminum, giving rise to a near-constant overall reflectance around 0.9. For both surfaces, simulations were performed for normal incidence and repeated for 25 wavelengths between $0.4\mu m$ and $0.7\mu m$. The incident Gaussian beam waist was chosen as $w = 3.2\mu m$, and for each surface and wavelength, we perform a pair of simulations using Gaussian beams with two orthogonal linear polarizations and later average the BRDF values derived from the paired simulations.

Surface BRDFs are shown in Fig. 10, where each hemisphere plot visualizes a surface's BRDF values corresponding to a fixed incident direction—normal incidence in this case—and a collection of outgoing directions that cover the upper hemisphere. Both single-wavelength ($\lambda = 0.4\mu m$) and multiple-wavelength BRDF plots from different simulation resolutions are shown for comparison, and the color BRDFs are computed using the standard spectral data → XYZ → RGB conversion. The predicted BRDFs from the 80×80 simulations significantly deviate from higher resolution simulations. The 160×160 simulations also reveal some noticeable differences, but the results from the 320×320, 640×640 and 960×960 simulations are visually identical. For our shortest wavelength of $0.4\mu m$, our chosen resolutions correspond to 2, 4, 8, 16, and 24 samples per wavelength in the height field's domain. These results show that our simulation is consistent under refinement. Moreover, they suggest that using a resolution of around 8 samples per wavelength in the height field domain is sufficient to achieve accurate results for these examples, even though the effective sampling rate along the surface is lower in steeply sloped areas.

6.2 Comparison with Existing Wave Based Models

We now compare our results with those from other wave optics methods that can compute spatially varying BRDFs with high resolution details. A recent work by Yan et. al. [2018] provides these kinds of BRDF models, which can be directly applied to rough surfaces represented as 2D height fields. We use three of their models for comparing with our simulations—the reciprocal original Harvey-Shack (OHS) model, the reciprocal generalized Harvey-Shack (GHS) model, and the Kirchhoff-based model. The BRDFs based on OHS and GHS were derived by approximating the rough surface as a plane that reflects light with a spatially-varying phase shift. The spatially varying phase shifts depend on the surface heights and are approximated using the original Harvey-Shack diffraction theory and the generalized Harvey-Shack theory, which is more accurate for large incident and outgoing angles [Krywonos 2006]. The Kirchhoff-based BRDF is derived from the Kirchhoff diffraction integral, which removes the use of a planar proxy when approximating phase shifts and integrating over the surface.

Another first-order reflection model, based on vector rather than scalar fields, is introduced in [Xia et al. 2023]. Like our method, this model introduces fictitious surface currents and computes scattered fields using the source-field relationships. However, it assumes that at each point on the surface, the incident field reflects from the plane tangent to the surface at this point, so that the reflected field values can be computed from closed-form formulas that describe flat surface reflection [Bohren and Huffman 2008]. We refer to this method as the *tangent plane* method in the rest of the text.

To compare our simulations with other methods, we performed simulations on 6 different $24\mu m \times 24\mu m$ surface samples, each discretized into 960×960 basis elements. Simulations were done using five incident directions—the normal direction, and four other directions given by $(\theta_i, \phi_i) = (18°, 0°), (36°, 90°), (54°, 180°), (72°, 270°)$. The Gaussian beam waists were computed using Eq. 26 with a primary waist of $w = 5.5\mu m$. As with the consistency tests, the surface material is aluminum, and for each incident direction, simulations were performed with 25 wavelengths and 2 linear polarizations. Thus, we perform $5 \times 25 \times 2 = 250$ simulations for each surface.

The tangent plane method was also applied a total of 250 times for each surface, while the methods from [Yan et al. 2018] were each applied 125 times for each surface, since these methods do not distinguish between polarizations. The Gaussian beam incident fields map naturally to the Gaussian coherence kernel used in these methods, allowing us to match the illumination conditions across all methods. In this paper, we feature a subset of surface samples and present BRDF plots for some selected incident directions. Our supplemental document contains a larger collection of results.

Unstructured Surfaces. We performed simulations on isotropic surfaces generated from random processes and brushed surfaces synthesized through resampling the height data provided in [Yan et al. 2018]. The first row of Fig. 1 characterizes a smooth surface with isotropic bumps, and the incident direction is given by $\theta_i = 36°, \phi_i = 90°$. Note that in Fig. 1, 11 and 12, the BRDF values computed using the five different methods match in magnitude. For this very smooth surface, the scattered energy is constrained in a relatively small cone of directions, and the methods show excellent agreement, with only the OHS-based method slightly deviating from the others.

The top rows of Fig. 11 show a brushed metal surface with moderate roughness, and BRDF plots are provided for normal incidence and an incident direction with $\theta_i = 18°, \phi_i = 0°$. The BRDF plots are very similar across the methods, except that the OHS model overestimates BRDF values for outgoing directions close to grazing.

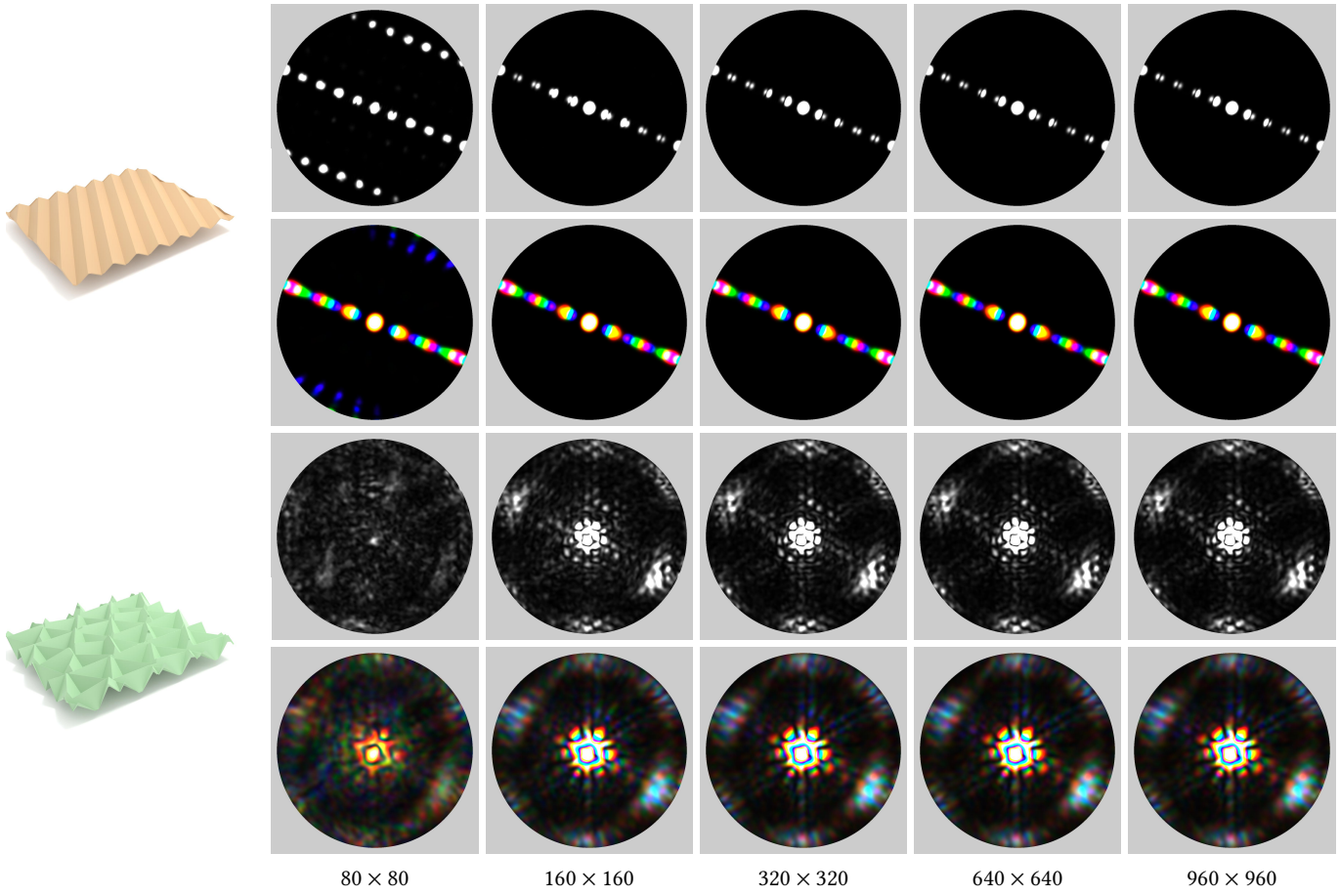

Fig. 10. Single-wavelength ($\lambda = 0.4\mu m$) and multiple-wavelength BRDF plots for two surfaces samples, obtained from simulations of different resolutions. Normal incidence is used in all simulations. The predicted BRDFs from the 80 × 80 simulations significantly deviate from those from higher-resolution simulations. The 160 × 160 simulations produce reasonable results, while results from the 320 × 320, 640 × 640 and 960 × 960 simulations are visually identical.

The bottom rows of Fig. 11 show a very rough, isotropic surface, with BRDF plots provided for normal incidence and an oblique direction with $\theta_i = 36°, \phi_i = 90°$. The surface scatters energy into the entire upper hemisphere. With normal incidence, the BRDF plots appear similar for the last three methods, while the larger incident angle gives rise to a slightly different plot for almost every method. The Kirchhoff and tangent plane method mostly match our simulation well, but compared with our simulation, they both overestimate the BRDF values for θ_o close to 90°. We believe this is because methods that only model first-order reflection fail to account for shadowing and masking.

Structured Surfaces. We also include some structured surfaces that demonstrate more differences between our simulations and other methods. The first example—a scaled version of the corner cube sample used for consistency testing—is covered with small corner reflectors, each consisting of three mutually perpendicular and intersecting flat surfaces. As incident light hits the sample, it generally gets reflected three times by interacting with some corner reflector, and propagates back along the incident direction. This retro-reflecting behavior can only be modeled by our simulation.

The BRDF plots are in the first two rows of Fig. 12, featuring normal incidence and another incident direction given by $\theta_i = 54°, \phi_i = 180°$. For normal incidence, our simulation predicts energy reflected back along the normal direction, while the other four methods predict "empty" BRDF plots. In fact, only modeling the first-order reflection from this surface results in scattered light that propagates in directions that belong in the *lower* hemisphere.

To prevent the surface from functioning as a diffraction grating, we gave each corner cube a different size and orientation, and some corner cubes partly overlapped with others. As a result, especially with non-normal incidence, the incoming light sometimes interacts with only one or two faces of a corner cube. For the 54° incident angle, the common component in the BRDF plots from the last three methods suggests that some corner cubes only induce first-order reflection of the incident light, which is predicted by our simulation and some approximate methods. Features in the BRDF plots unique to the approximate methods result from their incorrect prediction

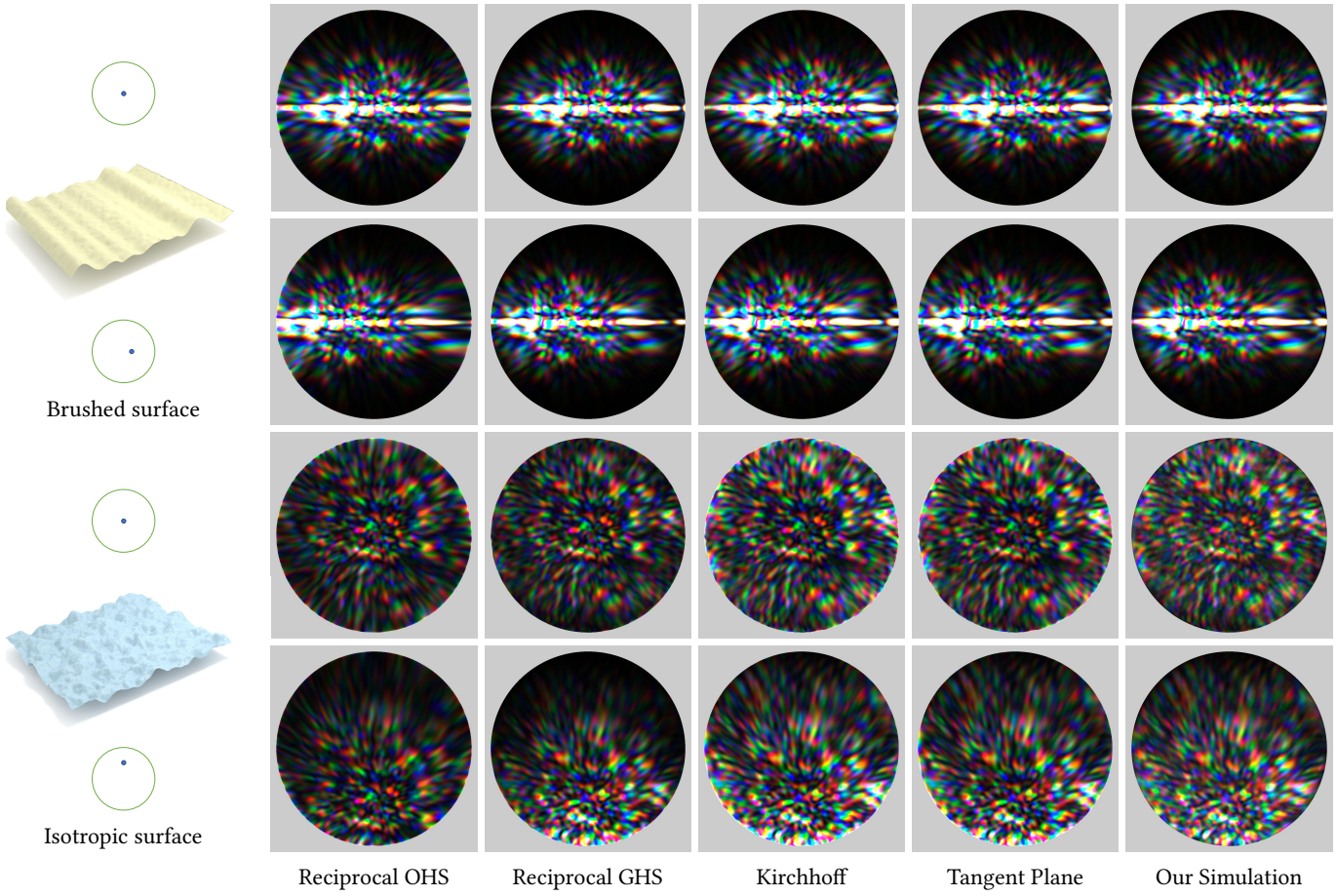

Fig. 11. Wavelength-dependent BRDFs for a moderately rough brushed metal surface and a very rough isotropic, bumpy surface. The circle and dot icons in the leftmost column represent the incident directions corresponding to each row of results. The first and third rows demonstrate normal incidence on the surfaces, while the second and fourth rows feature 18° and 36° incident angles, respectively.

of first-order reflection from corner cubes that in fact induce higher-order reflection. On the other hand, features unique in the BRDF plot computed from our simulation represent higher-order reflection, and the peak in BRDF values around the incident direction indicates retro-reflection from the surface.

The lower half of Fig. 12 describes a surface covered with spherical pits, which, due to the high slopes at their edges, also induce multiple reflection. BRDF plots are shown for normal incidence and a direction with a 72° incident angle. The BRDF patterns are overall similar for the methods, with similarly arranged color bands predicted from the last four methods. Compared to other methods, our simulation predicts brighter color bands, which we believe result from the interference between reflection of different orders.

Table 2 provides a summary of our simulations on the surfaces and incident directions featured in Fig. 11 and Fig. 12. In the table, the third dimension in the "size" column refers to the range of total height variation in the surface sample. The number of iterations in MINRES solving (using a certain tolerance in the solver code) and the total simulation time are similar across wavelengths and polarizations, but vary significantly among the surfaces and incident directions. Thus, we provide estimates of these data by averaging across wavelengths and polarizations.

Table 2. A summary of simulations on selected surfaces, reporting average number of MINRES iterations and average simulation time for each surface and presented incident directions.

surface	size (μm)	(θ_i, ϕ_i)	iter	time
brushed	$24 \times 24 \times 2.6$	normal	540	4.9 min
		(18°, 0°)	548	4.9 min
isotropic	$24 \times 24 \times 2.7$	normal	607	5.3 min
		(36°, 90°)	638	5.5 min
corner	$24 \times 24 \times 3.9$	normal	1468	14.4 min
		(54°, 180°)	1782	17.3 min
spherical	$24 \times 24 \times 1.7$	normal	644	5.0 min
		(72°, 270°)	816	6.0 min

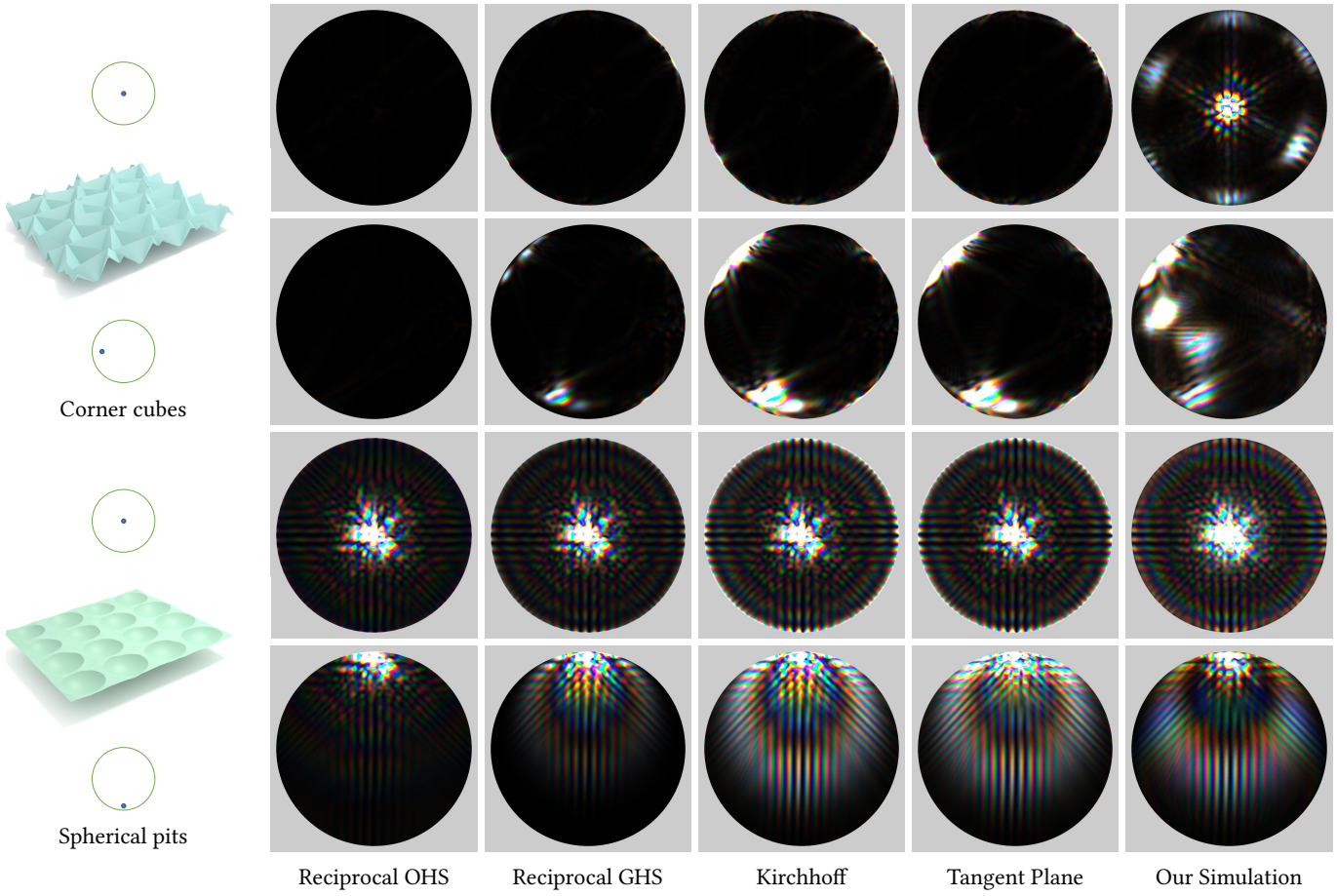

Fig. 12. Wavelength-dependent BRDFs for two surfaces, covered with corner reflectors and spherical pits. The circle and dot icons represent the incident directions for each row of results. The first and third rows demonstrate normal incidence, while the second and fourth rows feature 54° and 72° incident angles.

6.3 Coherence Area and BRDFs

We further show some examples where the same surfaces are illuminated by incident fields with different coherence areas. We are interested in comparing the BRDFs resulting from illumination of different levels of coherence, such as from light sources of different sizes. Two $30\mu m \times 30\mu m$ surface samples are used—one contains isotropic bumps while the other simulates a brushed metal surface. Instead of illuminating each surface with one Gaussian beam, we use an idea related to the beam steering approach of Section 5 and consider $8 \times 8 = 64$ small simulation domains which partly overlap, each with a size of $12.5\mu m \times 12.5\mu m$ (the "shift" between adjacent domains is $2.5\mu m$). In these examples we only demonstrate normal incidence, and in each small domain, we use a Gaussian beam with waist $w = 2.5\mu m$. The 64 simulations are performed separately, and to simulate illumination with different coherence areas, the scattered field values are combined in four ways:

- 1×1 groups: The Gaussian beams used in the small simulations are considered mutually incoherent, and the surface BRDF is the averaged BRDF of the 64 simulation areas.
- 2×2 groups: The Gaussian beams are divided into 16 coherence groups, each containing 2×2 Gaussian beams. BRDFs are computed for each group by summing the scattered fields with the proper phase shifts (Eq. 40). The surface BRDF is the average of the 16 subregion BRDFs.
- 4×4 groups: The Gaussian beams are divided into 4 groups, each containing a 4×4 grid of Gaussian beams.
- 8×8 group: All the 64 Gaussian beams are considered mutually coherent.

Fig. 13 shows that for both surfaces, the BRDFs contain more high resolution details when the illumination is more coherent, and the BRDFs corresponding to less coherent illumination appear blurry. Two factors contributing to this difference are the band limit on the far-field scattering pattern that is imposed by the small coherence area and the averaging of intensity patterns over several subregions. For more complete comparison, we also include examples of subregion BRDFs computed from a single $2.5\mu m$ beam. The supplemental document shows all 64 single-beam BRDFs for each surface.

The diameter of coherence area (D_c) is estimated for each illumination scheme and reported in Fig. 13. We consider an infinitely large surface normally illuminated by infinitely many Gaussian beams with the same waists and spacing as in our simulations. We then consider a sequence of distances $\{D_i\}$, and for each distance $D_{i'}$, we randomly sample many pairs of points $\mathbf{x}_1, \mathbf{x}_2$ on the surface, such that $|\mathbf{x}_1 - \mathbf{x}_2| = D_{i'}$. For each pair of points, we evaluate the coherence function $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ [Roelandt et al. 2013]

$$\gamma(\mathbf{x}_1, \mathbf{x}_2) = \frac{\langle E(\mathbf{x}_1) E^*(\mathbf{x}_2) \rangle}{\sqrt{I_1 I_2}} \tag{44}$$

where I_1, I_2 denote the field intensities at $\mathbf{x}_1, \mathbf{x}_2$ and $\langle \cdot \rangle$ refers to averaging over time. Averaging the value of $\gamma(\mathbf{x}_1, \mathbf{x}_2)$ for collections of samples that satisfy $|\mathbf{x}_1 - \mathbf{x}_2| = D_{i'}$ gives us an estimate of the coherence at distance $D_{i'}$. Repeating the same process for different distances gives rise to the relationship between spatial coherence and distance for an illumination scheme. The popular definition of coherence diameter—the distance over which the value of $|\gamma|$ has dropped to $1/e^2$—is then used to estimate D_c [Roelandt et al. 2013]. Note that since the illumination areas of neighboring Gaussian beams partly overlap, D_c does not scale linearly with the size of the coherence groups (i.e. 1, 2, 4, 8).

6.4 Some Beam Steering Examples

Here we demonstrate our beam steering technique for fast BRDF computation and its application both to our simulations and to the first-order tangent plane method. We consider a $32.5\mu m \times 32.5\mu m$ surface and $9{\times}9 = 81$ small simulation domains which partly overlap, each with a size of $12.5\mu m \times 12.5\mu m$. As shown in Fig. 14, the surface reflects specularly along one direction and is retro-reflecting along the other, thanks to the right angles between some neighboring planar components of the surface. We computed the surface BRDF values for a collection of smoothly varying incident angles, and as illustrated in Fig. 14, the plane of incidence is perpendicular to the direction along which the surface is smooth.

Since the surface reflects specularly along one direction, the BRDF plot for each incident direction reduces to a thin slice, so in Fig. 15, we visualize the BRDFs for incident angles between -30° and 30° in a compact way. The BRDFs on the left were computed by synthesizing small simulations performed with 13 basic incident directions (while the range of considered incident angles is 60°), and the BRDFs on the right were computed by using the tangent plane method in place of our simulation in the same framework.

The comparison in Fig. 15 reveals that the tangent plane method fails to model 2nd-order reflection from the surface. Still, as some of our results imply, the tangent method can be an accurate alternative to our simulation for computing the BRDFs of smoother surfaces. Since the tangent plane method is less expensive, we used it with our BRDF generation framework to compute BRDFs for two other surfaces, which we present in our supplemental video. Specifically, in the video we demonstrate smoothly varying BRDFs as the incident direction is steered smoothly in the upper hemisphere. Notably, the tangent plane method was invoked on the small domains for less than 500 basic incident directions, while the BRDF plots were synthesized for 1800 incident directions.

6.5 Comparison with Measurements

We tested the simulator's ability to predict measurements of a real surface exhibiting strong multiple reflection effects. For an example of a surface where both the BRDF and surface geometry are known, we look to the work of Sturniolo et al. [2022], who fabricated grooved surfaces as shown in 16 (a), where the cross section of each individual half-cylinder is shown in 16 (b). We compare our simulation results with the informal BRDF measurement for this sample included in their paper. Since this surface has constant cross section along one direction, we modeled it with a 2D version of our simulator. Light reflection from the interface between R_1 and R_2 in 16 (b) was computed by our simulator, while the transmission of incident and outgoing light through the flat interface between R_0 and R_1 was computed based on Fresnel's laws. We also used the tangent plane method to model reflection from the same structure.

The sample exhibits structural colors under normal incidence, as indicated from the blue, red, and green color bands in Sturniolo et. al.'s measurement, shown in Fig. 17. The tangent plane method did not predict much reflection at all, since the colors cannot be explained by first-order reflections. Our simulation predicts a reflection pattern that shows green around the normal direction and transitions into purple, red, and eventually a dim glimpse of green. In the measurement, the color around the normal direction is covered by the reflection of the light source, and the rest of the color progression is overall similar in terms of hue.

There are significant differences in the exact colors, and the mismatch could be caused by a combination of discrepancies between our idealized geometric model and the real structure of the fabricated sample, differences in color processing between our system and the unknown imaging pipeline of the consumer RGB camera used in the measurement, and unmodeled aspects of the illumination beam in the experiment. There are also some differences between the outgoing angles at which the colors transition, which could again be related to discrepancies in the exact surface geometry. Despite these differences, this comparison shows that the overall appearance of this sample can be predicted far more reasonably by BEM than by previous physical optics based scattering models.

7 DISCUSSION

In this section, we discuss the ranges of validity for different reflection models based on our results from the previous section. We also discuss the advantages and limitations of our simulation and BRDF generation framework.

7.1 Evaluation of Reflection Models

The results in the previous section demonstrate different levels of agreement among the five compared methods. For the very smooth surfaces, as expected, there is excellent agreement between our simulation and all four of the first-order methods. For rougher isotropic and brushed surfaces, the BRDFs computed from the methods are still largely similar, especially for normal incidence. The weakness of the OHS model for modeling large-angle incidence and scattering is revealed in many of our results, while other first-order methods, especially the Kirchhoff based model and the tangent plane model, seem to remain accurate even on fairly rough surfaces.

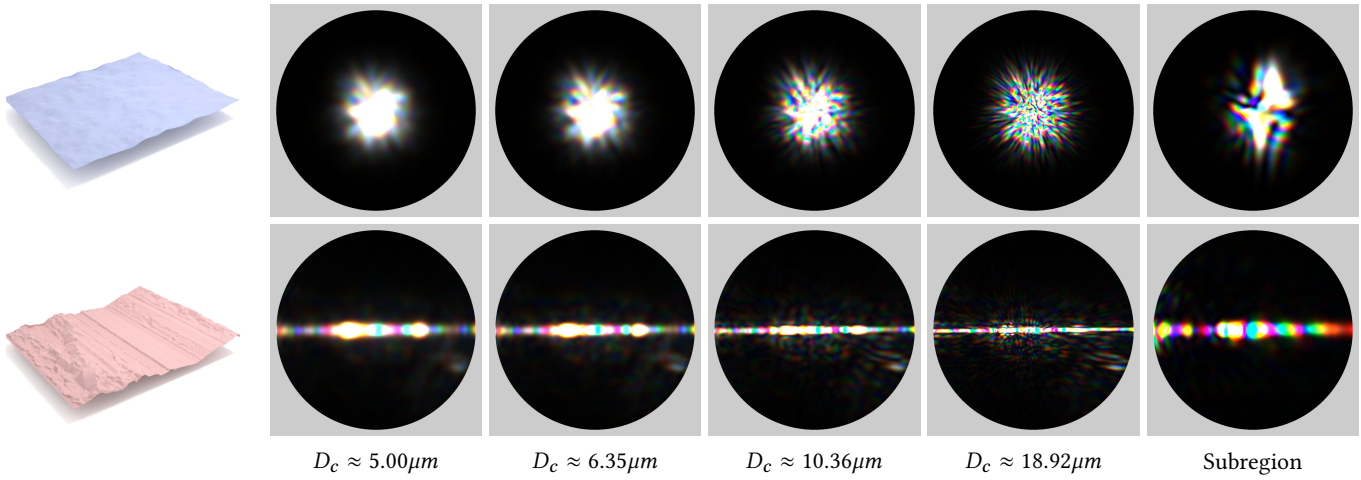

Fig. 13. Multi-wavelength BRDFs of an isotropic surface and a brushed surface under illumination with different coherence diameters D_c (e.g. due to light sources of different sizes or solid angles). These BRDFs are computed through performing multiple simulations on different subregions of the surfaces and summing the results coherently or incoherently. For each surface, we also included an example of a single subregion BRDF—see Section 6.3 for explanations.

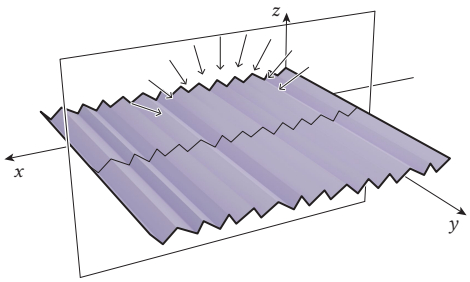

Fig. 14. A surface that reflects specularly along one direction and is retro-reflecting along the other. BRDF values are evaluated and shown for this surface, and the featured incident angles all lie in the plane of incidence that is perpendicular to the direction along which the surface is smooth.

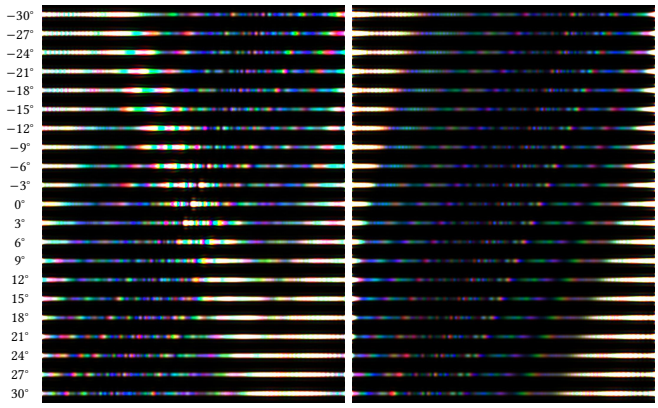

Fig. 15. BRDF slices from the surface shown in Fig. 14. The plane of incidence is perpendicular to the direction along which the surface is smooth, and the incident angle ranges from −30° to 30°. The differences between the left and right figures represent the 2nd-order reflection that the tangent plane method fails to capture. In the left figure, the movement of the 2nd-order reflection pattern with the incident angle indicates retro-reflection.

These suggest that cheaper first-order wave-based reflection models are usually sufficient for characterizing common types of rough surfaces characterized by isotropic or brushed patterns. These wave models, instead of expensive simulations, can potentially be combined with our BRDF generation framework to compute high-resolution BRDFs for large surface samples. Moreover, the observation that first-order reflection models tend to be most inaccurate at large outgoing angles suggests that these methods may be improved if they are combined with some ad hoc modeling of shadowing and masking (e.g. [Dong et al. 2016]).

Our last few examples show that the rendering of many particular types of surfaces requires more sophisticated wave models. Our simulations of multiple reflection from these example surfaces demonstrate how the simulation can be used to study surfaces with unusual optical properties, including artificial "metasurfaces" and complex natural surfaces. Our simulator can be used to design or study such surfaces, and even though our simulator cannot be directly applied to rendering due to its high cost, it can contribute to rendering through motivating and guiding the design of new efficient appearance models.

7.2 Limitations

There are a few aspects of our simulations that we hope to improve, and some considerations may lead to interesting future work.

Upper Bound on Surface Slopes. Our basis elements in each simulation are based upon the regular 2D grid associated with the

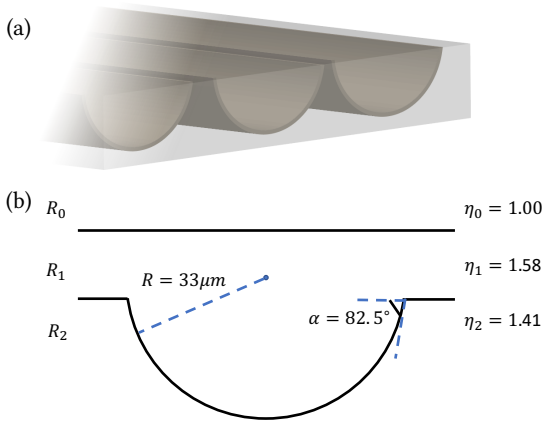

Fig. 16. Illustration of samples fabricated by Sturniolo et. al. (a) The grooved surface contains many copies of the same structure; this figure is adopted from [Sturniolo et al. 2022]. (b) The cross section of each individual cylindrical structure; the structure consists of two types of dielectric materials.

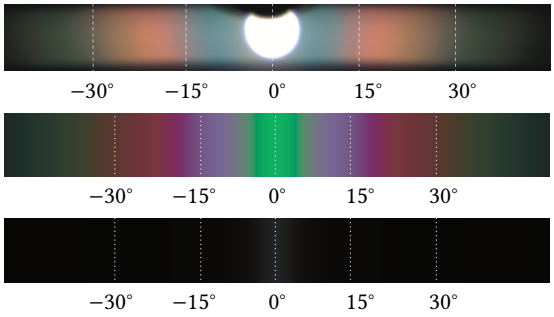

Fig. 17. Reflected colors from the cylinder-shaped sample under normal incidence, as measured in experiments (top), predicted by our simulation (middle), and predicted by the tangent plane method (bottom).

surface height field. Thus, on regions of the surface that contain high slopes, the effective size of each basis element can be a lot larger than its xy footprint size. Thus, a given xy footprint size d for the basis elements puts an upper bound on the surface slopes. For instance, with a choice of $d \approx \lambda/16$ for the minimum wavelength ($0.4\mu m$), our simulated surfaces, in theory, should not contain regions with slopes higher than 60°. Still, the consistency tests in Fig. 10 suggest that simulation resolutions lower than what we use (e.g. $d \approx \lambda/8$) can lead to results indistinguishable from those produced in high resolution simulations, even for surfaces with high slopes. This means that in practice, many surfaces containing slopes much higher than our theoretical upper bound may be well characterized by simulations using our usual resolution. Conversely, surfaces containing low slopes might be well simulated using lower resolutions, as long as all the sharp features in the surface samples are resolved.

Single-Layered Modeling. Despite its ability to model surfaces represented by largely arbitrary height fields, our simulator is currently limited to structures that only contain single-layered interfaces. The BEM surface simulations can be easily generalized to multi-layer interface structures, although introducing basis elements on multiple surfaces and constructing larger linear systems make the simulations increasingly expensive and thus limited to smaller domain size. Some multi-layered structures may be reasonably characterized by modeling the behavior of each interface individually, yet we believe that more sophisticated and efficient methods are needed for modeling multi-layered structures.

Grazing Incident Angles. BRDFs are most difficult to measure and model for grazing incident directions. According to Eq. 26, the incident Gaussian beams for large incident angles in our simulations have small waists, and eventually are not very good models of collimated light. This puts an upper bound on the incident angles we can model, depending on the primary beam waist w. For instance, the primary waist we used in beam steering is $2.5\mu m$, which gives rise to a waist less than $1\mu m$ for incident angles larger than 70° and results in large divergence, according to Eq. 25. This limits the maximum incident angle in our *basic* incident directions to around 72°; beam steering further allows us to simulate incident angles up to around 80°, though these very large incident angles can occasionally lead to visible artifacts in results.

Applicability of Beam Steering. The steering-based BRDF generation framework we presented works well on most types of rough surfaces, but there are cases where the individual domains cannot be made much smaller. For instance, if multiple reflections involve far-apart regions on the surface, then these two regions on the surface must both be included in a single simulation domain.

8 CONCLUSION AND FUTURE WORK

We have presented a 3D full-wave simulator for explicitly modeled rough surfaces, together with a framework for efficiently computing BRDF values for many incident and outgoing directions. The simulation is very useful for exploring the scattering behavior of unusual surface geometries and for evaluating the accuracy of approximate models that can be used in rendering.

Our results confirm the applicability of many existing first-order reflection models for low roughness, unstructured surfaces, but they indicate a strong need for wave optics based models that accurately compute high-order reflection, in order to extend our ability to render more structured surfaces. We hope to develop computationally inexpensive but accurate BRDF models, test their validity against our reference simulations on small problems, and generalize these appearance models to large-area surfaces involved in rendering or designing complicated materials. We believe that this work is an important step to accurate appearance modeling using wave optics, and look forward to its future applications.

ACKNOWLEDGMENTS

We would like to thank Dr. Anil Damle for his helpful suggestions on iterative solvers. We also want to thank Nathaniel Sturniolo and Dr. Lauren Zarzar for providing us with their experimental data on

grooved surfaces and for many fruitful discussions on structural color formation and modeling. Lastly, we would like to thank Xi Deng for her help with illustrating some of the figures in this paper. This work is supported by the National Science Foundation under grant IIS-1909467.

REFERENCES

Thomas Auzinger, Wolfgang Heidrich, and Bernd Bickel. 2018. Computational Design of Nanostructural Color for Additive Manufacturing. *ACM Trans. Graph.* 37, 4 (jul 2018). https://doi.org/10.1145/3197517.3201376

Petr Beckmann and Andre Spizzichino. 1987. *The scattering of electromagnetic waves from rough surfaces.* Artech House.

Laurent Belcour and Pascal Barla. 2017. A Practical Extension to Microfacet Theory for the Modeling of Varying Iridescence. *ACM Trans. Graph.* 36, 4 (jul 2017). https://doi.org/10.1145/3072959.3073620

Elizabeth Bleszynski, Maria Bleszynski, and Thomas Jaroszewicz. 1996. AIM: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems. *Radio Science* 31, 5 (1996), 1225–1251.

Craig F Bohren and Donald R Huffman. 2008. *Absorption and scattering of light by small particles.* John Wiley & Sons.

Brent Burley. 2012. *Physically-Based Shading at Disney.* Technical Report.

Sou-Cheng T Choi. 2013. Minimal residual methods for complex symmetric, skew symmetric, and skew Hermitian systems. (2013). arXiv:1304.6782

R. L. Cook and K. E. Torrance. 1982. A Reflectance Model for Computer Graphics. *ACM Trans. Graph.* 1, 1 (jan 1982). https://doi.org/10.1145/357290.357293

Tom Cuypers, Tom Haber, Philippe Bekaert, Se Baek Oh, and Ramesh Raskar. 2012. Reflectance Model for Diffraction. *ACM Trans. Graph.* 31, 5 (sep 2012). https://doi.org/10.1145/2231816.2231820

Daljit Singh Dhillon, Jeremie Teyssier, Michael Single, Iaroslav Gaponenko, Michel Milinkovitch, and Matthias Zwicker. 2014. Interactive Diffraction from Biological Nanostructures. In *Eurographics 2014 - Posters*, Mathias Paulin and Carsten Dachsbacher (Eds.). The Eurographics Association. https://doi.org/10.2312/egp.20141064

Miroslav Djordjevic and Branislav M Notaros. 2004. Double higher order method of moments for surface integral equation modeling of metallic and dielectric antennas and scatterers. *IEEE Transactions on antennas and propagation* 52, 8 (2004), 2118–2129.

Zhao Dong, Bruce Walter, Steve Marschner, and Donald P. Greenberg. 2016. Predicting Appearance from Measured Microgeometry of Metal Surfaces. *ACM Trans. Graph.* 35, 1 (2016). https://doi.org/10.1145/2815618

V. Falster, A. Jarabo, and J. R. Frisvad. 2020. Computing the Bidirectional Scattering of a Microstructure Using Scalar Diffraction Theory and Path Tracing. *Computer Graphics Forum* 39, 7 (2020). https://doi.org/10.1111/cgf.14140

Jeppe Revall Frisvad, Niels Jørgen Christensen, and Henrik Wann Jensen. 2007. Computing the Scattering Properties of Participating Media Using Lorenz-Mie Theory. *ACM Trans. Graph.* 26, 3 (jul 2007). https://doi.org/10.1145/1276377.1276452

Walton C Gibson. 2021. *The method of moments in electromagnetics.* CRC press.

Ibón Guillén, Julio Marco, Diego Gutierrez, Wenzel Jakob, and Adrian Jarabo. 2020. A General Framework for Pearlescent Materials. *ACM Trans. Graph.* 39, 6 (nov 2020). https://doi.org/10.1145/3414685.3417782

Yu Guo, Adrian Jarabo, and Shuang Zhao. 2021. Beyond mie theory: systematic computation of bulk scattering parameters based on microphysical wave optics. *ACM Transactions on Graphics (TOG)* 40, 6 (2021), 1–12.

James E Harvey. 1979. Fourier treatment of near-field scalar diffraction theory. *American Journal of Physics* 47, 11 (1979), 974–980.

Xiao D He, Kenneth E Torrance, Francois X Sillion, and Donald P Greenberg. 1991. A comprehensive physical model for light reflection. *ACM SIGGRAPH computer graphics* 25, 4 (1991), 175–186.

Nicolas Holzschuch and Romain Pacanowski. 2017. A Two-Scale Microfacet Reflectance Model Combining Reflection and Diffraction. *ACM Trans. Graph.* 36, 4 (jul 2017). https://doi.org/10.1145/3072959.3073621

Aleksei P Kiselev and Alexandr B Plachenov. 2019. Astigmatic Gaussian beams: exact solutions of the Helmholtz equation in free space. *Journal of Physics Communications* 3, 11 (2019), 115004.

Andrey Krywonos. 2006. *Predicting Surface Scatter using a Linear Systems Formulation of Non-Paraxial Scalar Diffraction.* Ph. D. Dissertation. University of Central Florida.

Andrey Krywonos, James E Harvey, and Narak Choi. 2011. Linear systems formulation of scattering theory for rough surfaces with arbitrary incident and scattering angles. *JOSA A* 28, 6 (2011), 1121–1138.

Ann M. Lanari, Samuel D. Butler, Michael Marciniak, and Mark F. Spencer. 2017. Wave optics simulation of statistically rough surface scatter. In *Earth Observing Systems XXII*, Vol. 10402. International Society for Optics and Photonics, SPIE. https://doi.org/10.1117/12.2274234

Anat Levin, Daniel Glasner, Ying Xiong, Frédo Durand, William Freeman, Wojciech Matusik, and Todd Zickler. 2013. Fabricating BRDFs at High Spatial Resolution Using Wave Optics. *ACM Trans. Graph.* 32, 4 (2013). https://doi.org/10.1145/2461912.2461981

Tien-Hao Liao, Leung Tsang, Shaowu Huang, Noppasin Niamsuwan, Sermsak Jaruwatanadilok, Seung-bum Kim, Hsuan Ren, and Kuan-Liang Chen. 2016. Copolarized and Cross-Polarized Backscattering From Random Rough Soil Surfaces From L-Band to Ku-Band Using Numerical Solutions of Maxwell's Equations With Near-Field Precondition. *IEEE Transactions on Geoscience and Remote Sensing* 54, 2 (2016), 651–662. https://doi.org/10.1109/TGRS.2015.2451671

YJ Liu and N Nishimura. 2006. The fast multipole boundary element method for potential problems: a tutorial. *Engineering Analysis with Boundary Elements* 30, 5 (2006), 371–381.

Michael I Mishchenko, Larry D Travis, and Andrew A Lacis. 2002. *Scattering, absorption, and emission of light by small particles.* Cambridge university press.

A. Musbach, G. W. Meyer, F. Reitich, and S. H. Oh. 2013. Full Wave Modelling of Light Propagation and Reflection. *Comput. Graph. Forum* 32, 6 (sep 2013). https://doi.org/10.1111/cgf.12012

Michael Oren and Shree K. Nayar. 1994. Generalization of Lambert's Reflectance Model *(SIGGRAPH '94)*. https://doi.org/10.1145/192161.192213

Ardavan F Oskooi, David Roundy, Mihai Ibanescu, Peter Bermel, John D Joannopoulos, and Steven G Johnson. 2010. MEEP: A flexible free-software package for electromagnetic simulations by the FDTD method. *Computer Physics Communications* 181, 3 (2010), 687–702.

Kyung Pak, Leung Tsang, and Joel Johnson. 1997. Numerical simulations and backscattering enhancement of electromagnetic waves from two-dimensional dielectric random rough surfaces with the sparse-matrix canonical grid method. *JOSA A* 14, 7 (1997), 1515–1529.

Rüdiger Paschotta. 2008. *Field guide to lasers.* Vol. 12. SPIE press Bellingham, WA.

Stijn Roelandt, Jani Tervo, Youri Meuret, Guy Verschaffelt, and Hugo Thienpont. 2013. Propagation of partially coherent light through a light pipe. *Optics express* 21, 14 (2013), 17007–17019.

Iman Sadeghi, Adolfo Munoz, Philip Laven, Wojciech Jarosz, Francisco Seron, Diego Gutierrez, and Henrik Wann Jensen. 2012. Physically-based simulation of rainbows. *ACM Transactions on Graphics (TOG)* 31, 1 (2012), 1–12.

Ryo Shimada and Yoichiro Kawaguchi. 2005. Spectral BRDF creation for structural colors. In *ACM SIGGRAPH 2005 Posters*. 48–es.

Jos Stam. 1999. Diffraction shaders. *Siggraph* 99 (1999), 101–110.

Shlomi Steinberg, Pradeep Sen, and Ling-Qi Yan. 2022. Towards Practical Physical-Optics Rendering. *ACM Trans. Graph.* 41, 4 (jul 2022). https://doi.org/10.1145/3528223.3530119

Shlomi Steinberg and Ling-Qi Yan. 2022. Rendering of Subjective Speckle Formed by Rough Statistical Surfaces. *ACM Trans. Graph.* 41, 1 (feb 2022). https://doi.org/10.1145/3472293

Nathaniel E Sturniolo, Krista Hirsch, Caleb H Meredith, Beau C Beshires, Shawn Khanna, Malak S Rayes, Michael A Gallegos, Shannon McGee, Bryan Kaehr, and Lauren D Zarzar. 2022. Iridescence from total internal reflection at 3D microscale interfaces: mechanistic insights and spectral analysis. *Advanced Materials* (2022), 2210665.

Antoine Toisoul and Abhijeet Ghosh. 2017. Practical Acquisition and Rendering of Diffraction Effects in Surface Reflectance. *ACM Trans. Graph.* 36, 5 (2017). https://doi.org/10.1145/3012001

Bruce Walter, Stephen R Marschner, Hongsong Li, and Kenneth E Torrance. 2007. Microfacet models for refraction through rough surfaces. In *Proceedings of the 18th Eurographics conference on Rendering Techniques*. 195–206.

Sebastian Werner, Zdravko Velinov, Wenzel Jakob, and Matthias B. Hullin. 2017. Scratch Iridescence: Wave-Optical Rendering of Diffractive Surface Structure. *ACM Trans. Graph.* 36, 6 (2017). https://doi.org/10.1145/3130800.3130840

Christopher A White and Martin Head-Gordon. 1994. Derivation and efficient implementation of the fast multipole method. *The Journal of Chemical Physics* 101, 8 (1994), 6593–6605.

Mengqi Xia, Bruce Walter, Christophe Hery, Olivier Maury, Eric Michielssen, and Steve Marschner. 2023. A Practical Wave Optics Reflection Model for Hair and Fur. *ACM Transactions on Graphics (TOG)* 42, 4 (2023).

Mengqi Xia, Bruce Walter, Eric Michielssen, David Bindel, and Steve Marschner. 2020. A wave optics based fiber scattering model. *ACM Transactions on Graphics (TOG)* 39, 6 (2020), 1–16.

Ling-Qi Yan, Miloš Hašan, Steve Marschner, and Ravi Ramamoorthi. 2016. Position-normal distributions for efficient rendering of specular microstructure. *ACM Transactions on Graphics (TOG)* 35, 4 (2016), 1–9.

Ling-Qi Yan, Miloš Hašan, Bruce Walter, Steve Marschner, and Ravi Ramamoorthi. 2018. Rendering Specular Microgeometry with Wave Optics. *ACM Trans. Graph.* 37, 4 (2018). https://doi.org/10.1145/3197517.3201351

Kai Yang and Ali E Yilmaz. 2011. Comparison of precorrected FFT/adaptive integral method matching schemes. *Microwave and Optical Technology Letters* 53, 6 (2011), 1368–1372.