

# Integrating Clipped Spherical Harmonics Expansions

LAURENT BELCOUR and GUOFU XIE, University of Montreal  
CHRISTOPHE HERY and MARK MEYER, Pixar Animation Studios  
WOJCIECH JAROSZ, Dartmouth College  
DEREK NOWROUZEZHAI, McGill University

Many applications in rendering rely on integrating functions over spherical polygons. We present a new numerical solution for computing the integral of *spherical harmonics expansions* clipped to polygonal domains. Our solution, based on zonal decompositions of spherical integrands and discrete contour integration, introduces an important numerical operation for spherical harmonic expansions in rendering applications. Our method is simple, efficient, and scales linearly in the bandlimited integrand's harmonic expansion. We apply our technique to problems in rendering, including surface and volume shading, hierarchical product importance sampling, and fast basis projection for interactive rendering. Moreover, we show how to handle general, non-polynomial integrands in a Monte Carlo setting using control variates. Our technique computes the integral of bandlimited spherical functions with performance competitive to (or faster than) more general numerical integration methods for a broad class of problems, both in offline and interactive rendering contexts. Our implementation is simple, relying only on self-contained spherical harmonic evaluation and discrete contour integration routines, and we release a full source CPU-only and shader-based implementations (< 750 lines of commented code).

CCS Concepts: • **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability;

Additional Key Words and Phrases: Zonal harmonics, numerical integration

## ACM Reference Format:

Laurent Belcour, Guofu Xie, Christophe Hery, Mark Meyer, Wojciech Jarosz, and Derek Nowrouzezahrai. 2018. Integrating Clipped Spherical Harmonics Expansions. *ACM Trans. Graph.* 37, 2, Article 19 (March 2018), 12 pages. <https://doi.org/10.1145/3015459>

## 1 INTRODUCTION

Spherical integrals of functions on polygonal domains are fundamental to physically-based rendering (PBR) operations. The most common of which include computing the integral of view-evaluated BRDFs with incident radiance distributions at surfaces, and the volumetric analogue of integrating phase functions and in-scattered radiance. Until recently, aside from specialized problems that admit analytic solutions with axial moments [Arvo 1995] (effectively,

---

Derek Nowrouzezahrai acknowledges funding from the Natural Science and Engineering Research Council of Canada.

Authors' addresses: L. Belcour, 51 avenue Jean Kuntzmann, 38330 Montbonnot, France; email: laurent.belcour@gmail.com; G. Xie, Dongsheng Plaza A, 5th Floor, 8 Zhongguancun East Rd, Haidian District, Beijing, China 100083; email: guofu85@gmail.com; C. Hery and M. Meyer, 1200 Park Avenue, Emeryville, California, U.S.; email: chery, mmeyer@pixar.com; W. Jarosz, Sudikoff 156, HB 6211, Dartmouth College, 9 Maynard St., Hanover, NH 03755; email: wojciech.k.jarosz@dartmouth.edu; D. Nowrouzezahrai, McConnell Eng. Building, 3480 University Street, Montreal, Quebec, Canada H3A 0E9; email: derek@cim.mcgill.ca.

---

© 2018 Association for Computing Machinery.

This is the author's version of the work. It is posted here for your personal use. Not for redistribution. The definitive Version of Record was published in *ACM Transactions on Graphics*, <https://doi.org/10.1145/3015459>.

integrals of circularly symmetric spherical *monomials*), the most robust and efficient methods for computing these integrals relied on numerical quadrature or Monte Carlo integration. Here, notable exceptions are the recent methods proposed by Heitz et al. [2016] and Lecocq et al. [2016], which we discuss in Section 2.

We propose a new numerical solution to integrals of bandlimited spherical *polynomials* (i.e. any finite spherical harmonic (SH) expansion) over arbitrary polygonal domains (i.e., spherical polygons with arbitrary angular variation), extending more limited analytic solutions for clipped spherical monomial integrals to numerical integrators for clipped finite/bandlimited SH expansions. We rely on a (lossless) zonal decomposition of the SH integrand, which we extend to leverage analytic *axial moment integrals* [Arvo 1995]. Our computation scales linearly with the harmonic expansion of the *unclipped integrand*, implicitly handling polygonal boundaries that introduce infinite frequencies in the spherical integrand, such as the shading cosine clamped to the upper hemisphere (an implicit polygonal boundary formed by the horizon).

Our approach handles bandlimited integrands more efficiently than previous exact numerical integration techniques. It is simple to implement (we provide full, self-contained source code). We discuss and apply our approach to several practical problems in rendering, focusing on applications to surface and volumetric shading, e.g., computing outgoing radiance at surface points with an bandlimited BRDFs, or out-scattered radiance at volumetric shading points with bandlimited phase functions. In both cases, we support not only diffuse polygonal light sources, but also polygonal sources with bandlimited angular emission distributions, and polygonal “portals” to environmental sources. Prior to integration, we must compute the SH expansion of the integrand. Only a subset of the aforementioned scenarios admit analytic SH expansions, with the remainder requiring numerical (pre)computation of the finite SH integrand expansions. We discuss the practical details of this constraint in Sections 5 and 6. Moreover, we show how our method can be combined with control variates to integrate arbitrary non-bandlimited functions; here we combine the benefits of SH, which have been used in graphics since the early 1990s, with the generality of Monte Carlo numerical integration.

We demonstrate our approach on both offline and interactive rendering applications, incorporating our integrator into PBRT [Pharr et al. 2016] as well as an open source shader-based rendering engine. Finally, we discuss and illustrate how our integration approach can be used to perform hierarchical sample warping [Clarberg and Akenine-Möller 2008b; Clarberg et al. 2005; Jarosz et al. 2009] for the product of BRDF times *local* area lights and applies to basis projection of lighting for relighting applications [Sloan et al. 2002].

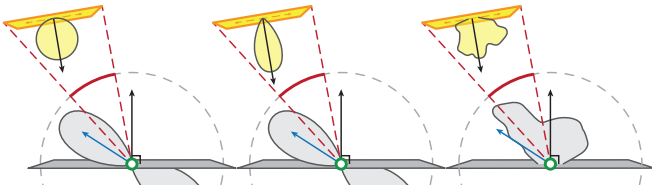


Fig. 1. Axial moments integrate cosine-power lobes & diffuse emitters [Arvo 1995], while we generalize to cosine-power lobes & cosine-power emitters, ZH expansions (not illustrated) and bandlimited emitters & BRDFs.

## 2 PREVIOUS WORK

We review relevant work on closed-form solutions to integration problems in rendering, as well as applications that apply tailored numerical solutions to the spherical polygonal integration problem.

*Closed-form integration in rendering.* Arvo’s 1995 seminal work on computing *axial moments*, integrals of circularly-symmetric cosine-power lobes over spherical polygons, was used to compute (unshadowed) reflected radiance on glossy surfaces from *diffuse* polygonal light sources (Figure 1 left). Chen and Arvo [2001, 2000] extend this formulation to linearly-varying luminaires. In a sense, Arvo’s axial moments are “incomplete” as they allow an arbitrary cosine-power lobe at a receiver, but only a diffuse lobe at the emitter. Our work generalizes the form of the integrand from cosine-power lobes to *bandlimited spherical polynomials*, subsuming these approaches. This allows us to not only support circularly symmetric lobes also at the emitter (Figure 1 middle), but in fact allows for *bandlimited spherical functions at both* the receiver and emitter (Figure 1 right). To do so, we decompose SH onto zonal harmonics (ZH; Section 3) to handle bandlimited integrands and applications.

Lecocq et al. [2016] reduce the algorithmic complexity of computing axial moments using an approximate closed-form expression, with only a modest impact on accuracy. Their approach still inherits the fundamental limitations of cosine-power integrands and diffuse polygonal luminaires; however, since our method builds directly atop axial moments, it would be possible to apply similar approximations to our work (see Section 7). Heitz et al. [2016] apply a linear transformation on a cosine distribution to approximate microfacet BRDFs of arbitrary roughness, yielding fast and analytical integration over clipped polygonal domains.

Previous work [Pegoraro and Parker 2009; Pegoraro et al. 2009; Sun et al. 2005] has investigated analytic or semi-analytic solutions to the air-light integral. This integration along a camera ray due to a point source can also be viewed as integrating a circularly symmetric spherical function (centered at the source) along the spherical arc formed by the camera ray’s spherical projection onto the source. We also treat spherical integration, but we consider a different and broader class of spherical integrands, and we support integrating along spherical polygons, not just spherical arcs.

Ramamoorthi and Hanrahan [2001] introduced the first analytic expression for (approximate) unshadowed diffuse shading from general spherical environment lights with SH, later extending the work to bandlimited glossy reflections [Ramamoorthi and Hanrahan 2002]. A similar extension to approximate scattering from hair

fibers also leverages SH [Mehta et al. 2012]. We instead consider the problem of integrating bandlimited spherical integrands over *polygonally-clipped subsets* of the full directional integration domain. This would, for example, allow us to extend Ramamoorthi and Hanrahan’s work to compute *exact* unshadowed diffuse shading from environment lights or shading from bandlimited anisotropic BRDFs and environment lighting masked by a polygonal occluder.

*Zonal decomposition.* Zonal decompositions are often used in interactive rendering to accelerate relighting applications. Perhaps the most common of such representations are spherical radial basis functions (SRBFs) and their variants [Green et al. 2006; Han et al. 2007; Tsai and Shih 2006; Wang et al. 2009]. These spherical Gaussian bases are powerful when computing very fast approximations to rendering problems, but they can require time-consuming and numerically challenging fitting to match ground truth profiles: specifically, optimal projection (in an  $L_2$ -sense) and reconstruction are challenging since representing arbitrary distributions with SRBFs relies on solving (costly) non-linear optimization problems. Delegating this fitting to precomputation, much like traditional precomputed radiance transfer (PRT) [Sloan et al. 2002], allows these approaches to very quickly compute plausible approximations, albeit not yet in the context of clipped spherical polygonal integration. The application of SRBFs to this problem is an interesting avenue of future work, given their representational potential. We instead are concerned with accurate numerical solutions to integrals of bandlimited spherical polynomials clipped to polygonal domains. To do so, we rely on *harmonic expansions* (with well-defined projection, reconstruction, and convergence properties) of the *unclipped* integrands. We extend these theoretical advantages of the SH representation to serve a practical, challenging integration problem: clipping the spherical integration domain to a polygon will introduce infinite frequencies, but our integration routine is robust to these. We need only to remain mindful of the bandlimit of the *unclipped* integrand, keeping in mind of course that the SH expansion of the integrand must be computed prior to integration. In some cases, this can be done analytically, but in general it requires a (well-defined and well-behaved) numerical precomputation. We discuss the practical implications of this requirement in Sections 5 and 6.

Nowrouzezahrai et al. [2012] present a mapping of spherical polynomials expressed in the SH basis to weighted sums of rotated, circularly symmetric zonal harmonics, which they use to compute fast rotations of SH vectors. This representation is similar in spirit to the use of ZH for representing bandlimited functions on the sphere in the geophysics and computational physics community [James 1969; Kempinski 1995; Lessig et al. 2012], however tailored to the accuracy-performance trade-offs of image synthesis applications. We base our expansion on this representation (Section 3). Lessig et al. [2012] derive a similar result using reproducing kernel Hilbert spaces, further analyzing the conditioning of the change-of-basis. Soler et al. [2015] use a similar harmonic decomposition into isotropic kernels in order to efficiently filter spherical signals.

*Control variates.* When an integrand can be partitioned into components that admit different (and potentially advantageous) solutions, *control variates* can be used to design Monte Carlo estimators that combine closed-form and numerical solutions [Fan et al.

2006; Lafortune and Willems 1994]. Previous work has combined specialized analytic solutions with control variates [Clarberg and Akenine-Möller 2008a; Mehta et al. 2012; Novák et al. 2014]. We show how our approach can be combined with control variates to solve a broad class of integration problems in rendering, illustrating concrete results in the case of shadowed outgoing radiance on surfaces and in volumetric participating media.

*Quadratures on the sphere.* We propose a numerical integration solution for bandlimited integrands clipped to spherical polygons. Several spherical quadratures rules exist and some can handle spherical triangles. Contrary to spherical cubature rules (such as Gia and Mhaskar [2009]), specific cubature rules defined for spherical triangles remove the additional error caused by the integration domain. Most of the time, quadrature rules are designed on the planar projection of the spherical triangles. We refer readers to the surveys of Hesse et al. [2010] and Cools and Rabinowitz [1993] on such topic. These rules require to incorporate a non-bandlimited Jacobian which make them non-exact for bandlimited integrands. Our technique directly supports spherical polygonal domains, without the need for triangulation, change of variable and is exact for bandlimited integrands.

More recently, Beckmann et al. [2014] presented a method that relies on precomputing the integral of SH basis functions over spherical triangles (a problem our approach solves), and they noted that no closed form existed for this “numerically delicate problem”; their method instead relies on high-precision approximations [Beckmann et al. 2012]. Note that such a precomputation requires at least  $N^2$  evaluations of the  $N^2$  basis elements, for an order- $N$  integrand, which leads to an  $O(N^4)$  time performance. Also, this precomputation depends on the spherical triangle and cannot be factored before pixel shading like our method permits. This signal-independent precomputation is akin to the (one-time) zonal harmonic factorization necessitated by our general SH integration technique.

Our approach is better suited to problems in computer graphics, where efficiency and modularity are equally important design considerations: our integrator scales as  $O(N^2)$  due to *lobe sharing* and incremental product accumulation (Section 4), as well as adapting to arbitrary polygonal domains. We additionally show how control variates can be used to adaptively sample higher-frequency content without choosing a sampling rate (i.e., quadrature rule) a priori.

*Relighting.* PRT projects light transport operators and external lighting onto a basis, and conduct a fast “analytic” reconstruction of the double- (i.e., [Ng et al. 2003; Sloan et al. 2002]), triple- (i.e., [Ng et al. 2004a]), or multi-product integrals [Sun and Mukherjee 2006] entirely in the basis space (e.g., SH, Haar wavelets, SRBFs). While these operations can be interpreted as closed-form approximations, the key trade-offs in PRT techniques lie in the compactness of the chosen basis and its representational power. PRT produces plausible (but ultimately inaccurate) results. We are able to integrate bandlimited polynomials over polygonal domains, and no basis is capable of simultaneously treating this class of integrand as well as the polygonal boundary without, e.g., requiring a larger<sup>1</sup> basis

<sup>1</sup>Spherical monomials are an obvious alternative to SH, however SH basis functions span the same space with fewer basis functions (beyond an order-1 expansion).

expansion or introducing potentially unbounded error in the integral approximation. While more powerful spherical distributions have been applied in graphics, SH expansions remain an important tool for many interactive graphics applications. We use SH expansions to express our bandlimited *unclipped* integrands, which we then clip to a polygonal domain prior to integration (potentially introducing infinite frequencies into the integrand; our method is robust to these infinite frequencies), an interesting setting that has motivated other recent advancements in interactive graphics [Heitz et al. 2016]. We leave the problem of integrating fully general spherical distributions over clipped polygonal domains to future work, with an potentially interesting direction being the exploration of the expressive power of *linearly-transformed SH expansions*. We discuss how our approach can also be used in the context of PRT applications (Section 6), however we pay more attention to more general rendering problems.

*Product importance sampling.* Another interesting area where integrals over (spherical) polygonal domains arise is in hierarchical sample warping for product importance sampling in Monte Carlo integration [Clarberg and Akenine-Möller 2008b; Clarberg et al. 2005; Jarosz et al. 2009]. Here, uniform input samples are warped to match a target distribution—e.g. the *product* of the view-evaluated BRDF and lighting—and used for Monte Carlo integration. During hierarchical warping, integrals of the basis functions must be evaluated over polygonal domains. Prior techniques are limited to basis functions and integration subdomains that admit analytic solutions (i.e. Haar or SH over axis-aligned integration regions only), restricting them to distant lighting. We discuss and show how our integration technique lifts these restrictions and allows product sampling even for local area lights (Section 6.1).

A recent approach solves the problem of importance-sampling an environment map *viewed through a portal* that masks all but a portion of it. Bitterli et al. [2015] devise a portal-centric reparameterization of the environment to importance sampling within the visible region. While their method is efficient even for high-frequency environments, it does not support product sampling with the BRDF, and it requires a separate tabulated environment map for each portal. Our method can *directly* integrate bandlimited spherical integrands—such as a bandlimited product of the BRDF and environment map—over a polygonal portal (Section 5.1). We demonstrate this with surface shading and in volumetric participating media. We can also apply our technique to extend these works to *product sample* the BRDF and environment *through* the portal (Section 6.1).

### 3 INTEGRATING SPHERICAL POLYNOMIALS

We will derive a numerical solution to integrals of the form:

$$I = \int_{\mathcal{P}} f(\vec{\omega}) d\vec{\omega}, \quad (1)$$

where  $f$  is a bandlimited spherical integrand in the Cartesian coordinates  $\vec{\omega} = (x, y, z)$ , possibly composed of a sum and/or product of spherical polynomials, and  $\mathcal{P}$  is a polygon (or its spherical projection). To do so, we perform a specialized harmonic expansion of  $f$ , before reducing the problem to that of computing a set of simplified cosine-power integrals that admit closed-form solutions.

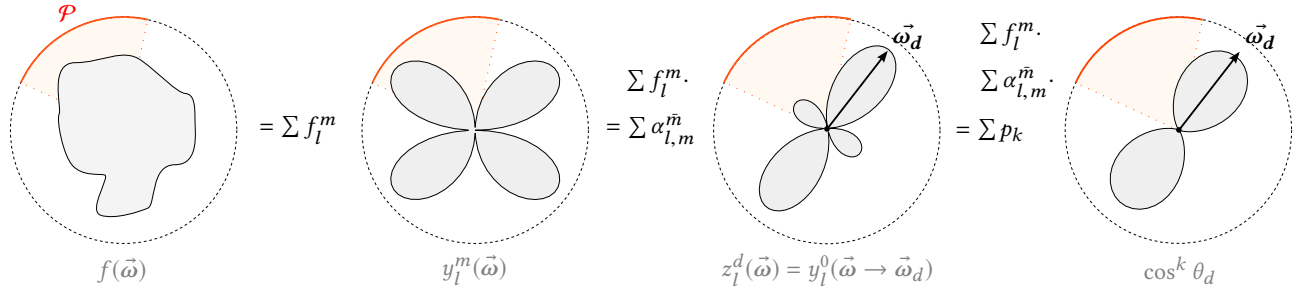


Fig. 2. We decompose bandlimited spherical integrands (left) into their SH expansion (left-middle), before reducing their representation using *zonal harmonic factorization* [Nowrouzezahrai et al. 2012] (right-middle) into a set of cosine-power integrals (right).

We review harmonic representations in Section 3.1, discuss their limitations when computing integrals of the form above, and derive our numerical solution in Section 3.2. We detail our efficient implementation in Section 4, leveraging recurrences in cosine-power integrals to reduce the cost of our method, which competes with (and often outperforms) more general numerical strategies.

### 3.1 Preliminaries

We adopt the following notation: italics for coefficients/scalars (e.g.  $f_l^m$ ), boldface barred italics for column vectors (e.g.  $\vec{f}$ ), boldface for points and directions (e.g.,  $\mathbf{x}$ ,  $\vec{\omega}$ ), and sans serif for matrices (e.g.  $M$ ).

*Spherical Harmonics.* Let  $f(\vec{\omega})$  be a spherical function, with  $\vec{\omega} = (x, y, z) = (\theta, \phi) \in S^2$  and  $(\theta, \phi)$  the spherical coordinates of the unit direction  $(x, y, z)$  on the surface of the 2-sphere,  $S^2$ .

Projecting  $f$  onto an SH basis function yields coefficients

$$f_l^m = \int_{S^2} f(\vec{\omega}) y_l^m(\vec{\omega}) d\vec{\omega}, \quad (2)$$

with  $f$ 's projection coefficient vector  $\vec{f} = [f_0^0, f_1^{-1}, \dots]$  and the SH basis functions defined as

$$y_l^m(\theta, \phi) = K_l^m \begin{cases} \cos(m\phi) P_l^m(\cos \theta), & m \geq 0 \\ \sin(|m|\phi) P_l^{|m|}(\cos \theta), & m < 0 \end{cases}, \quad (3)$$

where  $-l \leq m \leq l$  indexes the band- $l$  basis functions,  $P_l^m$  are associated Legendre polynomials, and  $K_l^m$  are normalization constants. Each of the  $2l + 1$  band- $l$  basis functions are degree- $l$  polynomials in the Cartesian coordinate  $(x, y, z)$  of  $(\theta, \phi)$  [Sloan 2008].

A bandlimited reconstruction  $\tilde{f}$  of  $f$  is obtained by weighting SH basis functions by the coefficient elements of  $\vec{f}$  as

$$f(\vec{\omega}) \approx \tilde{f}(\vec{\omega}) = \sum_{l=0}^{N-1} \sum_{m=-l}^l f_l^m y_l^m(\vec{\omega}) = \vec{f} \cdot \vec{y}(\vec{\omega}) \quad (4)$$

where  $\vec{y}(\vec{\omega}) = [y_0^0(\vec{\omega}), y_1^{-1}(\vec{\omega}), \dots]$  is a vector of the SH basis functions. An order- $N$  reconstruction requires  $N^2$  coefficients (for all bands  $l < N$ ) and is exact if  $f$  is order- $N$  bandlimited.

*Zonal Harmonic Factorization.* Zonal harmonics (ZH), the  $m = 0$  subset of the SH basis, are circularly-symmetric functions around  $\cos \theta = z$ . Each canonically-oriented band- $l$  ZH  $y_l^0(\vec{\omega})$  is a degree- $l$  polynomial in  $z$ . Sloan et al. [2005] apply the Funk-Hecke theorem to show that a weighted ZH function (oriented with  $z$  with coefficient  $g_l$ ) can be rotated to align with an arbitrary direction

$\vec{\omega}'$ , as  $f_l^m = n_l^* g_l y_l^m(\vec{\omega}')$ . Here, the resulting SH expansion coefficients  $f_l^m$  are simply scaled SH functions evaluated at  $\vec{\omega}'$ , and  $n_l^* = \sqrt{4\pi/(2l+1)}$  are the *convolution coefficients* [Sloan et al. 2005].

Several works in the computational physics and geophysics literature propose ZH as a representation for bandlimited spherical functions [James 1969; Kempfski 1995; Lessig et al. 2012], and Nowrouzezahrai et al. [2012] propose a similar *zonal harmonic factorization* (ZHF) of the SH basis to the graphics community. We choose to build atop this ZHF:

$$y_l^m(\vec{\omega}) = \sum_{\bar{m}=-l}^l \alpha_{l,m}^{\bar{m}} z_l^{(l+\bar{m})}(\vec{\omega}), \quad (5)$$

where  $z_l^d(\vec{\omega}) = y_l^0(\vec{\omega} \rightarrow \vec{\omega}_d)$  is the band- $l$  ZH rotated to align with the  $d^{\text{th}}$  pre-computed *lobe direction*  $\vec{\omega}_d$ . Note that Equation 5 can be thought of as the *dual* of the *SH Addition Theorem*, which states that the weighted sum of SH basis functions can be used to express a single rotated ZH basis function. Here, Equation 5 states that the weighted sum of rotated ZH basis functions can be used to express a single (canonically oriented) SH basis function. The ZHF weighting coefficients  $\alpha_{l,m}^{\bar{m}}$  and lobe directions are related, per band  $l$ , as follows:

$$N_l^* \underbrace{\begin{bmatrix} \alpha_{l,-l}^{-l} & \dots & \alpha_{l,l}^{-l} \\ \vdots & \ddots & \vdots \\ \alpha_{l,-l}^l & \dots & \alpha_{l,l}^l \end{bmatrix}}_{A_l} = \underbrace{\begin{bmatrix} y_l^{-l}(\vec{\omega}_{l^2}) & \dots & y_l^l(\vec{\omega}_{l^2}) \\ \vdots & \ddots & \vdots \\ y_l^{-l}(\vec{\omega}_{l^2+2l}) & \dots & y_l^l(\vec{\omega}_{l^2+2l}) \end{bmatrix}}_{(Y_l)^{-1}}. \quad (6)$$

Here,  $N_l^*$  is a diagonal matrix with each  $n_l^*$  repeated  $2l+1$  times, for  $0 \leq l < N$ , and any choice of lobe directions shared across bands  $l$  (i.e.,  $d \in [0, 2N-1]$ ; Section 4) that results in an invertible  $Y_l$  in Equation 6, yields a valid ZHF. We use a low-discrepancy pattern to generate  $\vec{\omega}_d$ . We discuss the choice of lobe directions, and their implications, in more detail in Section 6.2). We can now re-write Equation 5's generalization across all SH functions in matrix-vector form as  $\vec{y}(\vec{\omega}) = A \vec{z}(\vec{\omega})$ , where  $A$  is a band-diagonal matrix with sub-matrices  $A_l$ , and  $\vec{z}(\vec{\omega})$  is a vector of all  $z_l^d(\vec{\omega})$  functions.

### 3.2 Bandlimited Integrals Clipped to Spherical Polygons

Given the SH coefficients  $f_l^m$  of a function  $f(\vec{\omega})$ , it is trivial to compute the integral of  $f$  over the entire sphere  $\int_{S^2} f(\vec{\omega}) d\vec{\omega} = \sqrt{4\pi} f_0^0$ .



Computing the integral restricted to (the spherical projection of) a polygon (Equation 1) is, however, much harder. If the polygon is axis-aligned in  $(\theta, \phi)$ , then analytic expressions exist [Jarosz et al. 2009], but this kind of canonical orientation is too restrictive for general spherical integration applications.

One common approach to solving this problem is to rewrite Equation 1 as an integral over the entire sphere, modulating the integrand  $f$  by a *mask*  $v(\vec{\omega})$  that is 1 for  $\vec{\omega} \in \mathcal{P}$  and 0 elsewhere, and so

$$I = \int_{S^2} f(\vec{\omega}) v(\vec{\omega}) d\vec{\omega} \approx \vec{f} \cdot \vec{v}, \quad (7)$$

where the representation of  $v$  as an SH coefficient vector  $\vec{v}$  would require an **infinite** number of coefficients to yield the correct result, unless  $f$  is bandlimited: indeed, any spherical function with a sharp discontinuity (such as visibility, or polygonal emitters, etc.) will have infinite frequency content. Still, some integrands  $f$  in rendering may admit bandlimited expansions, however even then the formulation of Equation 7 remains problematic for the simple reason that computing the SH projection of  $v$  would require a costly numerical sampling-based integration. In fact, our work can also be used to more efficiently compute the SH projection of  $v$ , in addition to simply directly solving for  $I$  in Equations 1 and 7.

We instead present a numerical solution based on closed-form solutions of simpler *contour integrals*, over the boundary  $\partial\mathcal{P}$  of  $\mathcal{P}$ , that can **exactly** integrate masked bandlimited spherical integrands.

*Our Formulation.* Since integration is a linear operator, we can substitute the SH expansion of  $f$  into Equation 1 and exchange the order of the integral and sum to obtain:

$$I = \sum_{l,m} f_l^m \int_{\mathcal{P}} y_l^m(\vec{\omega}) d\vec{\omega}. \quad (8)$$

We will use ZHF and axial moments to reduce and solve the problem of computing Equation 8.

*Integrating SH over  $\mathcal{P}$ .* ZHF defines a linear mapping  $\vec{y}(\vec{\omega}) = A \vec{z}(\vec{\omega})$  (Section 3.1), so we substitute Equation 5 into Equation 8 and convert to matrix-vector form over *all* basis functions to obtain

$$I = \sum_{l,m} \sum_{\bar{m},d} f_l^m \alpha_{l,m}^{\bar{m}} \int_{\mathcal{P}} z_l^d(\vec{\omega}) d\omega = \vec{f} A \vec{z}_p, \quad (9)$$

and so we have now reduced the problem from that of computing Equation 1, to computing  $\vec{y}_p$ , to computing the elements of  $\vec{z}_p$ : namely, the integral of rotated zonal harmonics  $z_l^d(\vec{\omega}) = y_l^0(\vec{\omega} \rightarrow \vec{\omega}_d)$  over the spherical polygon  $\mathcal{P}$ . To do so, we will map the ZH basis into terms that admit closed-form axial moment integrals.

*Integrating ZH over  $\mathcal{P}$ .* Each rotated ZH  $z_l^d(\vec{\omega})$  is a Legendre polynomial (Equation 3) in  $(\vec{\omega} \cdot \vec{\omega}_d) = \cos \theta_d$ . Specifically, each band- $l$  ZH function  $z_l^d$  is a degree- $l$  polynomial in  $z$ , such that

$$z_l^d(\vec{\omega}) = y_l^0(\vec{\omega} \rightarrow \vec{\omega}_d) = \sum_{k=0}^l p_k \cos^k \theta_d, \quad (10)$$

and  $p_k$  are coefficients that weight powers of the  $z$  monomial, aligned with the lobe direction  $\vec{\omega}_d$  (i.e.,  $z = \cos \theta_d$ ), to form ZH basis functions. For example, for  $y_2^0(\theta) = \sqrt{5/(16\pi)}(3 \cos^2 \theta - 1)$ , we have  $p_0 = -\sqrt{5/(16\pi)}$ ,  $p_1 = 0$  and  $p_2 = 3 \times \sqrt{5/(16\pi)}$ . Substituting

---

**Algorithm 1:** Integrator routine: **Initialize** is called *once*, after which **Integrate** can be called many times with different inputs.

---

```
Initialize(N) : // Precompute AP product & dirs. {ωd}
1: // Generate ZHF lobe directions for order-N
2: {ωd} = UniformSphericalDirections(2N - 1)
3: P = ComputeP({ωd}) // SphericalIntegration.hpp:58
4: Y = SHMatrix({ωd}) // SphericalIntegration.hpp:194
5: AP = inverse(Y) × P
```

```
Integrate(P, f̄, N) :
```

```
1: c̄p = AxialMoments(P, {ωd}, N) // AxialMoments.hpp:277
2: return transpose(f̄) × AP × c̄p
```

---

Equation 10 into 9 and re-writing matrix-vector notation yields our full decomposition, illustrated in Figure 2,

$$I = \sum_{l,m} \sum_{\bar{m},d} f_l^m \alpha_{l,m}^{\bar{m}} p_k \int_{\mathcal{P}} \cos^k \theta_d d\omega = \vec{f} A P \vec{c}_p, \quad (11)$$

we obtain a sum of cosine-power integrals,  $\int_{\mathcal{P}} \cos^k \theta_d d\omega$ , over  $\mathcal{P}$ . Matrices  $A$  and  $P$  are constant and we precompute their product, so evaluating Equation 11 reduces to evaluating the entries of  $\vec{c}_p$ , the cosine power integrals, which we can readily compute in closed-form using axial moments [Arvo 1995]. We discuss how sharing directions  $\vec{\omega}_d$  across bands allows us to eliminate redundant work and reuse intermediate axial moment computation (Section 4).

## 4 OPTIMIZED EVALUATION

Naïvely implementing Equation 11 has  $O(N^3)$  time complexity in the degree  $N$  of  $f$ : each band  $l$  requires  $2l + 1$  axial moment computations (to evaluate the band's  $2l + 1$  elements of  $\vec{c}_p$ ), each of which scales with  $O(l)$  [Arvo 1995]. This  $O(l^2)$  cost per band results in  $O(l_{\max}^2)$  cost for all  $l \in [0, l_{\max}]$ , where  $l_{\max} = N - 1$ .

The axial moment's  $O(l)$  cost stems from a recurrence that requires integrals for cosine powers  $0 \leq k < l$  to be computed before the integral for power  $l$  can be obtained. A naïve implementation wastes this incremental computation. Using *lobe sharing* [Nowrouzezahrai et al. 2012], where a subset of  $2l + 1$  of the  $2N - 1$  directions  $\vec{\omega}_d$  are used for each band- $l$  ZHF, we need to only evaluate an axial moment at the highest band  $l_{\max}$  and store the intermediate cosine power integrals for use in the remaining  $l < l_{\max}$  bands. Lobe sharing and cosine-power amortization are important optimizations, the latter of which is necessitated by the scalability of edge-integration w.r.t. the cosine-power [Arvo 1995]; note, however, that applying Lecocq et al.'s 2016 approximation here would improve the performance of our integration routine, at the cost of accuracy. Still, doing so and analyzing the degree of this error (in the context of integrating SH expansions, as opposed to cosine-power distributions) is an interesting avenue of future work. Our technique's pseudocode in Algorithm 1 refers to our documented open source implementation, and we also provide shader source code for our interactive demo.

## 5 APPLICATIONS

Integrals of the form in Equation 1 arise in many rendering problems, and we will directly apply our integrator to several of these, below. This section focuses on the most direct applications in surface and

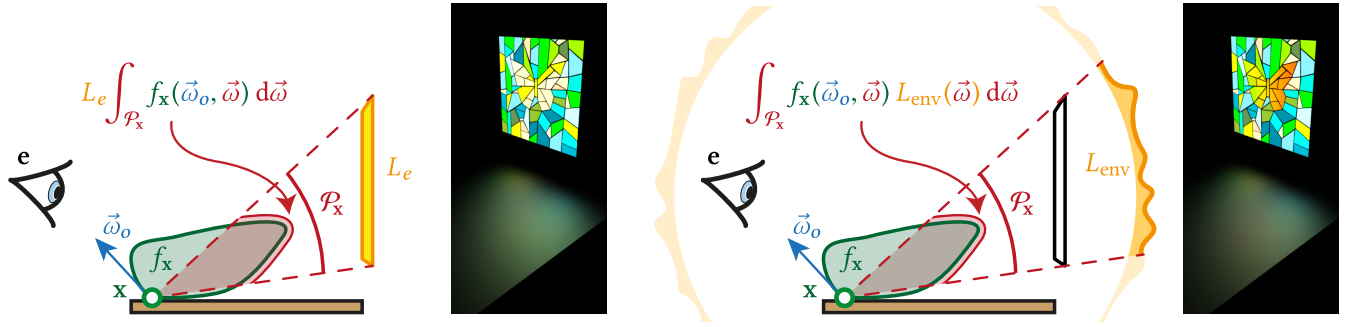


Fig. 3. Outgoing radiance due to a polygon light (left) and a portal-masked environment (right) where each shading point is lit by a different region of the environment. The renderings demonstrate these two types of sources in a scene inspired by Arvo [1995].

volumetric shading, but we also discuss applications to other use cases related to rendering in Section 6.1.

Specifically, we apply our integration routine to the computation of outgoing radiance on surfaces with bandlimited BRDFs, and in-scattered radiance in volumetric participating medium with a bandlimited phase function, both from polygonal light sources and *portal-masked* environment sources (Section 5.1). We first consider scenes without occluders before generalizing our integrator to more complex scenes, using *control variates* to handle any additional terms in the scattering equations (Section 5.2).

### 5.1 Surface and Volume Shading

For any application, we need to map the problem to an appropriate choice of  $f(\vec{\omega})$  (or, more specifically, its harmonic representation  $\tilde{f}$ ) and a definition of the polygonal domain  $\mathcal{P}$ . Any problem that can be mapped in this way can be solved with our method.

*Surface Shading with Polygonal Sources.* The outgoing radiance towards a viewer in direction  $\vec{\omega}_o$ , from a polygonal light with a *diffuse emission profile*, visible at a surface point  $\mathbf{x}$  with BRDF  $f_x$ , is defined by the reflection equation [Pharr et al. 2016]:

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e \int_{\mathcal{P}_x} f_x(\vec{\omega}_o, \vec{\omega}) d\vec{\omega}, \quad (12)$$

where  $\mathcal{P}_x$  is the spherical projection (centered at  $\mathbf{x}$ ) of the polygonal light (Figure 3, left). Note that different shading points will correspond to different polygonal domains, different viewing directions, and (potentially) different BRDFs.

By defining  $f(\vec{\omega}) = L_e f_x(\vec{\omega}_o, \vec{\omega})$  and  $\mathcal{P} = \mathcal{P}_x$ , we need only compute  $\tilde{f}$  before being able to apply our integration routine. In the case of certain analytic BRDFs, the coefficients of  $\tilde{f}$  may be computed analytically; however, for arbitrary (i.e., data-driven) BRDFs, these coefficients need to be precomputed once per BRDF.

When possible, we use analytic expressions for  $\tilde{f}$  (e.g., in the case of diffuse and Phong BRDFs) but in all other cases, including measured materials from the MERL [Matusik et al. 2003] database and anisotropic BRDFs, we precompute a doubly-projected BRDF matrix  $F_x$  of the BRDF. We evaluate the projection at  $\vec{\omega}_o$  as  $\tilde{f} = F_x \tilde{y}(\vec{\omega}_o)$ . Concretely, the elements of the BRDF matrix are:  $(F_x)_{i,j} = \int_{S^2} f_x(\vec{\omega}_o, \vec{\omega}) y_i(\vec{\omega}) y_j(\vec{\omega}_o) d\vec{\omega} d\vec{\omega}_o$ . We incorporate occlusion using control variates in an offline rendering context (Section 5.2), and we

implement real-time surface shading from polygonal sources in a shader-based rendering system (Figure 12).

*Surface Shading with Portal-masked Environments.* If, instead of a polygonal light our scene has a distant environment light (i.e., with arbitrary directional variation) which is only visible through a polygonal window, or *portal*, our method still applies. Here, the reflection equation now includes the environmental emission  $L_{env}(\vec{\omega})$  and  $\mathcal{P}_x$  corresponds to the portal (Figure 3, right):

$$L_o(\mathbf{x}, \vec{\omega}_o) = \int_{\mathcal{P}_x} f_x(\vec{\omega}_o, \vec{\omega}) L_{env}(\vec{\omega}) d\vec{\omega}. \quad (13)$$

In addition to different shading points having different view-directions and BRDFs, now they also “see” different portions of the environment through different (in relative coordinates) portals.

Here, we use  $f(\vec{\omega}) = L_{env}(\vec{\omega}) f_x(\vec{\omega}_o, \vec{\omega})$  and  $\mathcal{P} = \mathcal{P}_x$  (the portal geometry now, not a polygon light’s geometry) to apply our integration routine. The main difference from the previous application is that now the SH projection  $\tilde{f} = \tilde{f}_x \times_{SH} \tilde{L}$  is that of the *product* between the view-evaluated BRDF and the environment map. We obtain the BRDF SH coefficients  $\tilde{f}_x$  exactly as with the polygonal lights, and we can readily compute the projection coefficient of the lighting environment  $\tilde{L}$  (once, at initialization, with numerical integration), however computing  $\tilde{f}$  would require a costly SH product computation (i.e.,  $\times_{SH}$ ) between the BRDF and lighting environment SH coefficients, at every shade point (each integral evaluation).

We instead precompute and store an *SH product matrix*  $L$  for the lighting environment *once* at initialization. This matrix only changes upon environment rotation. Specifically, we precompute the individual SH product matrix elements of the lighting environment,  $(L)_{i,j} = \int_{S^2} L_{env}(\vec{\omega}) y_i(\vec{\omega}) y_j(\vec{\omega}_o) d\vec{\omega}$ , using Monte Carlo integration with  $N_s$  samples, as

$$(L)_{i,j} \approx \frac{1}{N_s} \sum_{s=1}^{N_s} \frac{L_{env}(\vec{\omega}_s) y_i(\vec{\omega}_s) y_j(\vec{\omega}_o)}{\text{pdf}(\omega_s)}, \quad (14)$$

with a uniform spherical sampling distribution with  $\text{pdf}(\omega) = 1/4\pi$ . This computation requires roughly 1 second for a  $1024 \times 512$  lighting environment. We show how to combine our technique with control variates, in order to handle occlusion, in Section 5.2.

SH product matrices are related to the *SH tripling 3-tensor*  $\Gamma$ , where the elements  $(\Gamma)_{i,j,k} = \int_{S^2} y_i(\vec{\omega}) y_j(\vec{\omega}) y_k(\vec{\omega}_o) d\vec{\omega}$  of the tensor are

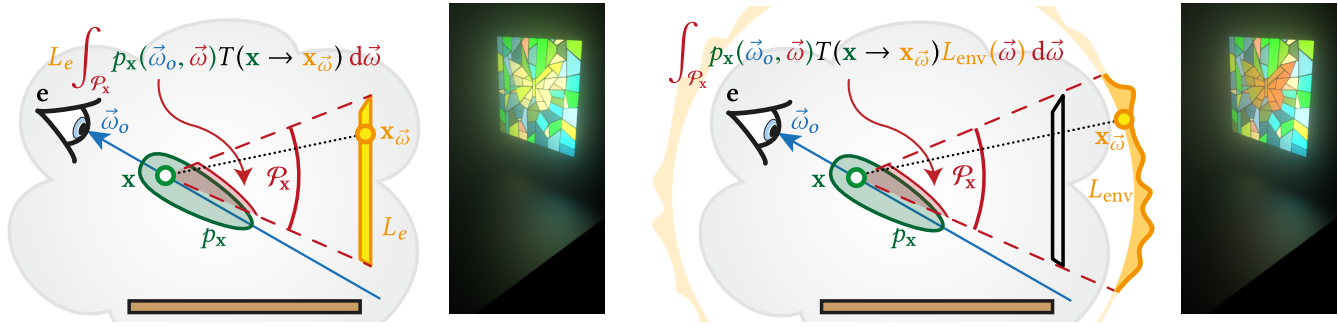


Fig. 4. In-scattered radiance in a participating medium due to a polygon light (left) and a portal-masked environment (right) where each shading point is lit by a different region of the environment. The renderings demonstrate these two types of sources in a scene inspired by Arvo [1995].

often referred to as the Clebsh-Gordan coefficients, as follows: any SH product matrix can be computed as a tensor-vector product of  $\Gamma$  and the SH coefficient vector of the associated product term [Ng et al. 2004b].

In the case of portal-masked environment shading, the elements of  $L$  could have alternatively been computed as  $(L)_{i,j} = \sum_k (\Gamma)_{i,j,k} (\bar{L})_k$  however, as we need only compute the product matrix infrequently (i.e., only when the lighting changes), we chose to directly compute its elements (using Equation 14) to avoid precomputing and storing the  $\Gamma$  tensor. If we wished, however, to support more local lighting variation, one could compute SH coefficients for some local lighting emission model before employing the triple product tensor-vector contraction equation to compute the necessary SH product matrix.

Environmental portals are used for lighting design in films. Bitterli et al. [2015] note that a portal is equivalent to a polygonal light with the environment map acting as a directionally varying emission profile, so our method also applies to this dual setup. Unlike Bitterli et al. however, who look to warp samples according to the portal-masked environment, we *directly compute* the outgoing radiance in this scenario (additionally taking a bandlimited BRDF into account). We also support arbitrary polygons, while Bitterli et al.’s approach only handles quads. In Section 6.1, we detail additional applications of our method, including hierarchical sample warping, which could extend Bitterli et al.’s technique to importance sampling according to the **product** of the BRDF and portal-masked environment, as Bitterli et al. proposes for future work.

*Volume Shading with Polygonal Sources.* The two applications above can be extended to compute in-scattered radiance towards a viewer in direction  $\vec{\omega}_o$  from a point  $\mathbf{x}$  in a participating medium with phase function  $p_x$ . For diffuse polygonal lights, the equation for in-scattered radiance is (see Figure 4, left)

$$L_o(\mathbf{x}, \vec{\omega}_o) = L_e \int_{\mathcal{P}_x} p_x(\vec{\omega}_o, \vec{\omega}) T(\mathbf{x} \rightarrow \mathbf{x}_{\vec{\omega}}) d\vec{\omega}, \quad (15)$$

where  $T(\mathbf{x} \rightarrow \mathbf{x}_{\vec{\omega}}) = e^{-\sigma_t |\mathbf{x} - \mathbf{x}_{\vec{\omega}}|}$  is the transmittance between  $\mathbf{x}$  and the line-of-sight point on the light  $\mathbf{x}_{\vec{\omega}}$  from a ray in direction  $\vec{\omega}$ , and  $\mathcal{P}_x$  is once again the spherical projection of the light.

We set  $f(\vec{\omega}) = p_x(\vec{\omega}_o, \vec{\omega})$  and  $\mathcal{P} = \mathcal{P}_x$  for this application, and use closed-form expressions for the phase function  $p$ ’s SH coefficients

$\bar{p}_x$ . We treat isotropic, Henyey-Greenstein, and Schlick scattering profiles, all of which admit analytic projection; however, SH projections can also be precomputed numerically for phase functions that do not admit closed-form projection. To compute the final pixel color, we raymarch along eye rays and apply our integrator to Equation 15 at the marching points by evaluating the transmittance at the center of the polygon and using our control variates to compensate for the true variation of transmittance and occluded geometry (Section 5.2). This strategy performs well, especially for distant sources and/or media with slowly varying scattering properties.

*Volume Shading with Portal-masked Environments.* Finally, if we generalize surface shading with portal-masked environments to the volumetric setting, the in-scattered radiance is

$$L_o(\mathbf{x}, \vec{\omega}_o) = \int_{\mathcal{P}_x} p_x(\vec{\omega}_o, \vec{\omega}) T(\mathbf{x} \rightarrow \mathbf{x}_{\vec{\omega}}) L_{env}(\vec{\omega}) d\vec{\omega}, \quad (16)$$

where each point along an eye ray “sees” a different portion of the environment (Figure 4, right). We set  $f(\vec{\omega}) = p_x(\vec{\omega}_o, \vec{\omega}) L_{env}(\vec{\omega})$  and  $\mathcal{P} = \mathcal{P}_x$ . We now need to use a product matrix in order to obtain the product’s SH coefficients:  $\bar{f} = \bar{p}_x \times_{SH} \bar{L} = L \bar{p}_x$ .

As before, we can efficiently precompute the product matrix for the environment  $L$  once. As before, we march along eye rays to compute accumulated outgoing radiance and use control variates to compute the residual of our point-wise evaluation of transmittance and  $T$ . Finally, we only treat transmission from the raymarch shading point up to the portal geometry and no further (since the environment is infinitely distant, and so not technically visible if the scene is engulfed in, i.e., a homogeneous media).

## 5.2 Generalization with Control Variates

The representational and functional properties of SH have promoted its adoption in many problems in computer graphics, and our integration technique adds to the benefits of SH. In many integration settings, however, integrands may not be bandlimited and SH might not immediately appear as an ideal representation; here, we show that by applying even a naïve *control variate* strategy, we can gracefully handle arbitrary integrands in a generic (and efficient) way, all while leveraging many advantages of the SH representation. For example, the applications above do not account for visibility (due to



occluders, with the exception of the portal’s surroundings), nor do they fully account for spatial variation in transmittance.

If product<sup>2</sup> terms in  $f(\vec{\omega})$  cannot be easily represented with spherical polynomials, we can express a new integrand as  $f = f_1 \times f_2$  and solve this problem with a control variate Monte Carlo estimate:

$$\begin{aligned} I &= \int_{\mathcal{P}} f(\vec{\omega}) d\vec{\omega} = I_{f_1} - \int_{\mathcal{P}} (f_1(\vec{\omega}) - f(\vec{\omega})) d\vec{\omega} \\ &= I_{f_1} - \frac{1}{N} \sum_{i=1}^N f_1(\vec{\omega}_i) (1 - f_2(\vec{\omega}_i)) / \text{pdf}(\vec{\omega}_i), \end{aligned} \quad (17)$$

where we still apply our method for  $I_{f_1}$ , and the second term in Equation 17 is a Monte Carlo (MC) estimate of the “difference” between  $I$  and  $I_{f_1}$ :  $N$  sampling directions  $\vec{\omega}_i$  are chosen according to a sampling probability distribution function  $\text{pdf}(\vec{\omega}_i)$ . The only important requirement is that we are able to sample the *integrand*  $f_1(\vec{\omega})$  of  $I_{f_1}$  as well as the additional term  $f_2(\vec{\omega})$ . Concretely, we need to be able to represent  $f_1$  exactly using an SH expansion.

Concretely, for the four applications in Section 5.1, modeling the effect of occluders necessitates an additional visibility term  $f_2(\vec{\omega}) = v_x(\vec{\omega})$  in the integrands of Equations 12–16. From a practical standpoint, compared to an MC estimate of Equation 1, our method provides the exact unoccluded result (for bandlimited integrands), whereas an MC estimate would require time to converge to a noise-free rendering; for scenes with occluders or non-bandlimited integrands, both techniques have noise that dissipates as  $N$  increases, with the noise being concentrated more in the shadow regions for our method.

In general,  $f(\vec{\omega})$  can be arbitrarily complex and, as long as  $f_1(\vec{\omega})$  is a good approximation of it *in certain instances*, applying our numerical solution with a control variate is a reasonable solution: our solution accounts for the bandlimited portion of an integrand, deferring the residual (higher-frequency) effects to MC integration. Note that we use only the most basic form of control variates, without applying any additional variance reduction techniques, such as importance sampling and/or stratification schemes (all PBRT renderings, however, use multiple importance sampling of the light and/or polygon and BRDF). We leave applications of other variance reduction strategies atop our control variate solution to future work.

## 6 RESULTS AND DISCUSSION

We validate the accuracy of our method, both in unoccluded settings and with control variates, against ground truth computed with importance-sampled Monte Carlo. We compare shading errors in a variety of scenes and lighting setups, testing the applications outlined in Section 5.1. In all cases, our method performs at least as well as *optimized* importance-sampling code on the surface and volume shading scenarios.

DRAGON (Figure 6) and SAN MIGUEL (Figure 10) scenes use Lambertian and Phong BRDFs, whereas SPHERES (Figure 5) uses measured MERL BRDFs. DRAGON and SPHERES are rendered with polygonal and portal lights. FOG (Figure 9) illustrates volume scattering with both polygonal and portal lights and anisotropic phase functions,

<sup>2</sup>A **sum** of additional integrand terms is trivial to handle, as  $I = I_{f_1} + I_{\text{other}}$  can be split into several integrals and treated independently.

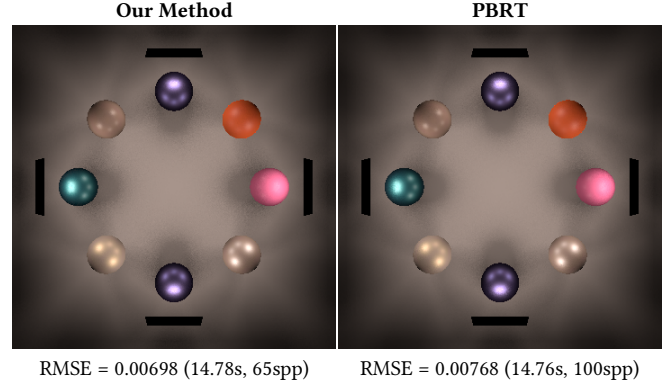


Fig. 5. The SPHERES scene uses measured MERL BRDFs using an order 15 SH expansion.

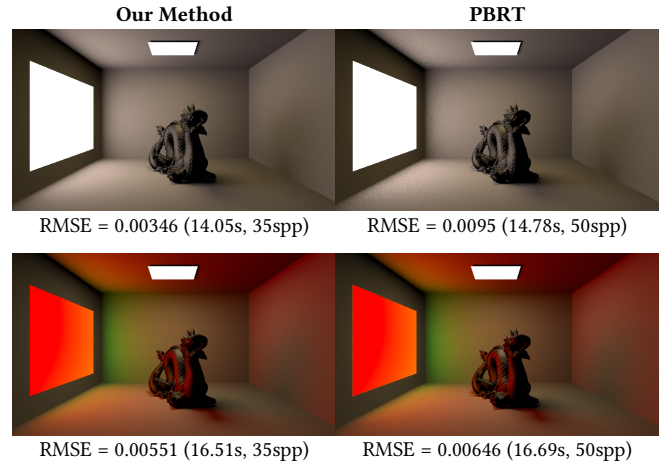


Fig. 6. DRAGON: a shiny dragon on a matte ground with a color gradient environment (green to red) portal light and an area light. We use an order 9 SH expansion for this scene.

and SAN MIGUEL illustrates our approach on a scene with complex coarse- and fine-scale geometry. We demonstrate interactive shading results with an unoptimized shader port of our CPU code, in the CRYTEK SPONZA with uniform and portal light sources, using diffuse, Phong, and data-driven BRDFs, all approximated using order-5 SH (Figure 12). Here, timings include the entire shading algorithm, including polygonal horizon clipping and SH product computation (when necessary). In this example, we use a multi-lobe BRDF (combining a diffuse clamped-cosine lobe with an order-5 SH expansion of a data-driven BRDF from the MERL dataset) and a multi-lobe order-5 portal-masked environment mapped emitter.

All images were captured on an Intel i5 2.6GHz with 8 GB of RAM and an Intel Iris 6100 mobile GPU with 1.5 GB of VRAM.

### 6.1 Additional Applications

Section 5 discussed some immediate applications of our method; however, several other problems lend themselves to our technique.

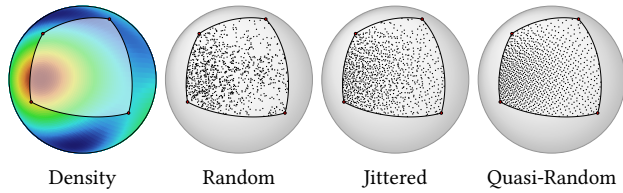


Fig. 7. We importance sample spherical polynomials over polygons (left) with hierarchical sample warping. This procedure can be applied to random points, but also preserves stratification of higher-quality sequences.

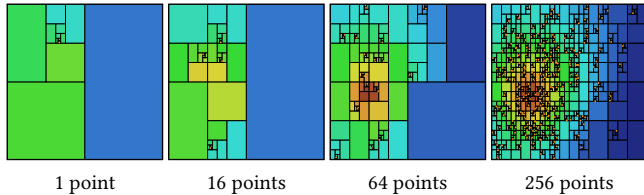


Fig. 8. Hierarchical sample warping distributes points by sparsely integrating over spherical subdomains (shown in Figure 7 projected onto the sphere). The number of evaluations for one point is exactly  $(2D + 1)$  where  $D$  is the recursive depth. Subsequent points only need to integrate unexplored regions which, in the worst case, scales linearly with sample count.

We will present two such applications below, mapping them to our integrator (as with the four applications in Section 5.1).

**Hierarchical Sample Warping.** Prior work on importance sampling the product of the BRDF and lighting [Clarberg et al. 2005; Jarosz et al. 2009] relies on a hierarchical strategy to warp uniformly distributed random points to these target distributions. To do so, the warping routine needs to compute integrals of the sampling distribution over restricted sub-domains on the sphere. Previous works either rely on approximations that require prohibitive amounts of precomputation and storage, only support axis-aligned sub-domains, or do not treat both the BRDF and lighting when warping. We can use our method to warp samples, for the first time, according to: the product of the BRDF, lighting and portal, **or** the product of the BRDF and a local polygon light with a general directional emission distribution (Figure 7).

We directly extend prior hierarchical warping algorithms [Clarberg et al. 2005; Jarosz et al. 2009] by exchanging the integration routine used to compute the integrals of the target density function over integration sub-domains. To simplify the hierarchical decomposition, we only consider quadrilateral polygons for this application. In the general case, we would require a 2D parameterization of the polygon in order to quickly decompose it (see below). Prior approaches start at the root (the entire integration domain) and, at each level, compute integrals over four *axis-aligned* rectangular sub-domains in *spherical coordinates*. We instead start with the light’s (or portal’s) spherical projection as our root and hierarchically subdivide this spherical quad into four child quads arranged in a  $2 \times 2$  grid. At each stage we compute the integral of the target density over the four spherical quads to warp input points into their respective

quadrants. This forms a quadtree-like subdivision of the integration domain (see Figure 8) based on which quadrants the points fall within as they are warped. After reaching a maximum recursion depth  $D$ , we obtain points distributed according to  $f$  within the spherical quad bounds (Figure 7) as well as corresponding pdf values for each point. For BRDF  $\times$  quad-light product-sampling we set  $\tilde{f} = \tilde{f}_x$ , and for BRDF  $\times$  lighting  $\times$  portal triple-product-sampling we set  $\tilde{f} = \tilde{f}_x \times_{\text{SH}} \tilde{L}$ , both as described in Section 5.1.

**Basis Projection.** In PRT, light transport is precomputed, projected onto a basis, and stored in a scene. At run-time, the projection of dynamic lighting is used to relight the scene with a basis-space multi-product integration [Ng et al. 2003; Sloan et al. 2002; Sun and Mukherjee 2006]. Projecting the lighting can sometimes be optimized with additional precomputation, but some dynamic computation is needed whenever lighting changes (e.g., rotation of the pre-projected light in SH, or re-projection for Wavelet bases).

We can use our method to compute lighting projection coefficients in many scenarios, some of which were previously infeasible as they relied on costly sampling-based numerical integration.

For distant polygonal sources  $\mathcal{P}$ , we can (pre)compute their projection coefficient  $L_i$  onto an arbitrary basis function  $b(\vec{\omega})$  with  $\tilde{f}_i = \int_{S^2} b(\vec{\omega}) y_i(\vec{\omega}) d\vec{\omega}$ , as this integral corresponds to the coupling coefficient of  $b$  onto SH. Of course, if we seek an SH projection of the lighting, then  $\tilde{f}_i = \tilde{f}_i^m = \delta_{i,m}$ , where  $\delta$  is a Kronecker delta vector that only “activates” a single SH basis. Interestingly, even the coupling coefficient itself can be computed with our method, for certain basis functions  $b(\vec{\omega})$ . For example, Haar Wavelet basis functions have piecewise constant form over the sphere, and so they are an immediate candidate for an application of our technique in this exact manner.

For local polygon lights, with spherical projections  $\mathcal{P}_x$  at each shade point  $x$ , we apply the same procedure at every shade point,

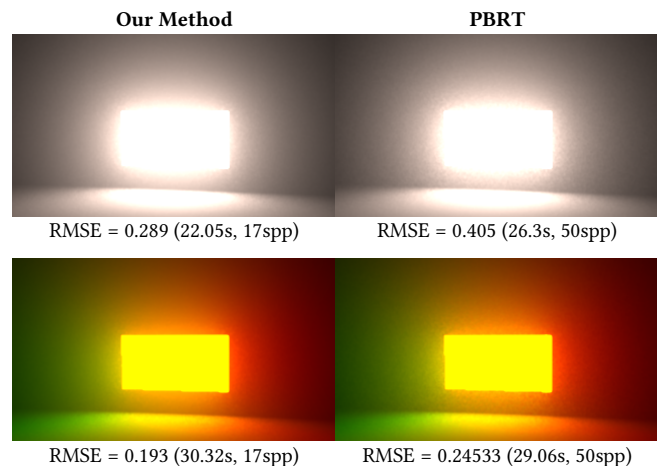


Fig. 9. The FOG scene with participating media: a homogeneous fog with a Henyey-Greenstein phase function (anisotropy  $g = 0.5$ ), which scatters light towards the observer. Our method integrates both a uniform area light (top) and portal light with directional variations (bottom). We used an order 3 SH expansion for this scene.



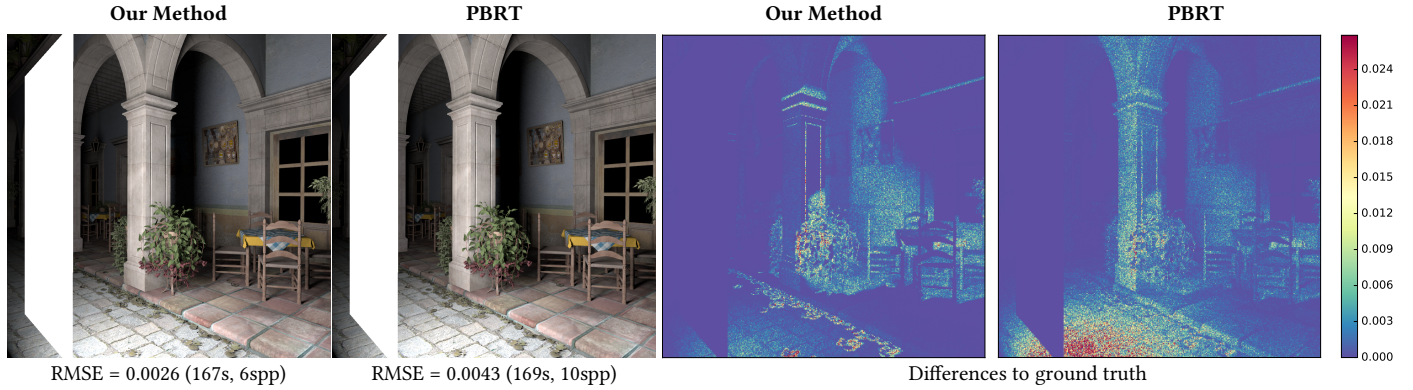


Fig. 10. The SANMIGUEL scene illustrates our method on complex geometry, such as the foliage and chairs. We used an order 3 SH expansion for this scene.

instead of once per frame. Supporting arbitrary local polygonal sources in PRT previously required costly precomputation over spatial locations in the scene [Kristensen et al. 2005] or approximate gradient-based interpolation [Annen et al. 2004]. The key contribution here is that, in addition to integrating SH *expansions* over polygons, one can simplify our method to integrate *individual SH basis functions* over the polygonal domain.

Both the distant and local polygonal lighting projection scenarios above can be extended to computing the projection coefficients for a portal-masked environment map, too. The polygonal domains remain the same, and only  $\hat{f}$  changes to  $\hat{f}_i = \int_{S^2} L_{\text{env}}(\vec{\omega}) b(\vec{\omega}) y_i(\vec{\omega}) d\vec{\omega}$ , which can be precomputed once for each basis function  $b(\vec{\omega})$  to allow for dynamically changing portal shapes. To project the (potentially dynamic) portal onto SH,  $f_i^m = L \vec{\sigma}_{l,m}$ .

## 6.2 Discussion

*Scalability.* As shown in Figure 11, our method’s rendering time scales linearly with number of SH coefficients (square of the polynomial degree). A naïve implementation scales quadratically.

A naïve shader port of our method (Figure 12) is a proof-of-concept of its potential in interactive shading. Heitz et al. [2016] linearly warp the polygonal integration domain to induce an effective integrand warp to form anisotropic single-lobe distributions. Their method warps a cosine distribution, whereas we represent integrands in SH. One avenue of future work could combine these ideas to explore the expressive power of *linearly-warped SH*.

Both our and Jarosz et al.’s 2009 hierarchical sample warping approaches evaluate SH integrals over many spherical sub-domains. In both cases, the number of such integrals scales linearly with the average recursion depth of the warp, and at most linearly with the number of samples (since points only need to integrate basis functions in previously unexplored regions). If the user specifies a maximum depth, then Jarosz et al.’s method can precompute all necessary SH integrals, while our method cannot leverage such an optimization since the integration domains differ at each point.

*Numerical Stability.* Albeit closed form, axial moments can suffer from numerical instability and precision loss at high cosine powers. Since we rely on this method, we unfortunately inherit this limitation. With double precision, Arvo’s (and our) method remains stable

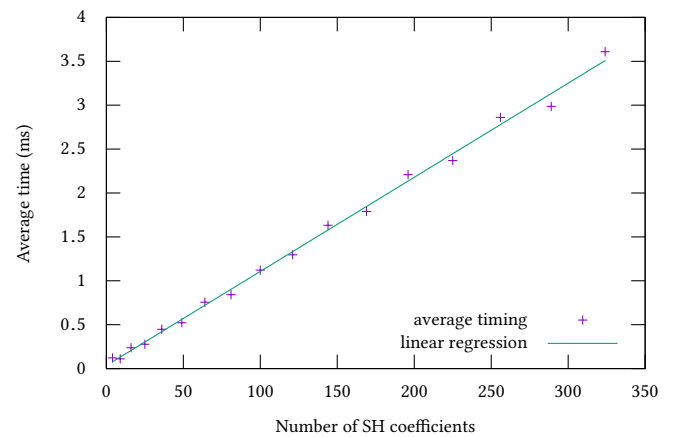


Fig. 11. Our method scales linearly with the number of SH coefficients (square of the SH order). We plot computation time with respect to number of SH coefficients for our SH integration method (function INTEGRATE of Algorithm 1). We ran the integral computation of a single triangle 1000 times and report the average timing.

up to about order 20. Applying Lecocq et al. [2016] approximation to our method could reduce numerical issues, at the price of bias. Beyond order 20 SH expansions, numerical imprecisions in the axial moment computation can affect the precision of our SH integration as the reliance of the ZHF mapping on accurate cosine-power evaluation grows.

*Zonal Harmonic Factorization Lobe Directions.* The choice of the ZHF lobe directions  $\omega_d$  can impact the numerical stability of the ZHF [Nowrouzezahrai et al. 2012]. Nowrouzezahrai et al. use a stochastic optimization in order to promote sparsity in the ZHF, an optimization better suited to their core application (efficient SH rotation); however, our application (SH expansion integration) does not require any such sparsity. Still, we experimented with various strategies for choosing the  $\omega_d$ : uniform spherical sampling, low-discrepancy sampling, random sampling, and the original ZHF sparsity mapping. We found that a low-discrepancy sample set yielded

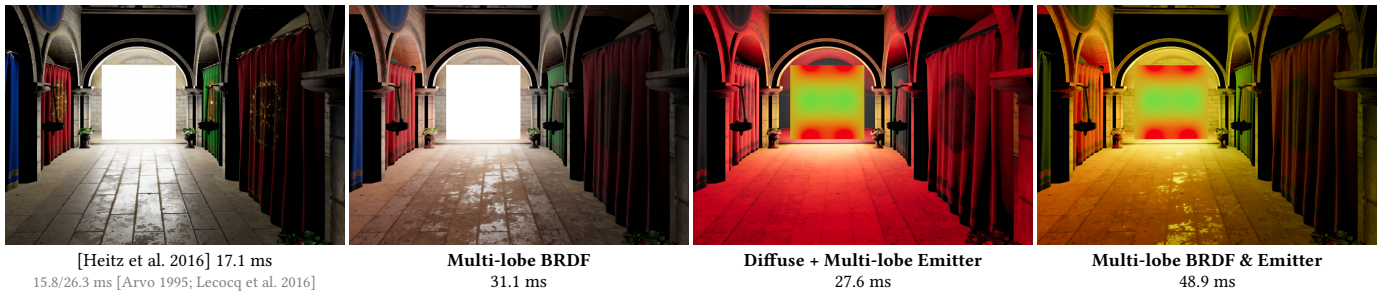


Fig. 12. CRYTEK SPONZA in an interactive renderer. Left to right: existing methods [Arvo 1995; Heitz et al. 2016; Lecocq et al. 2016] are fast and support single-lobe profiles. A direct shader port of our method supports SH expansions for the BRDF and for the emitter (i.e., portal lights), albeit at a performance cost: while interactive, it is not yet suitable for game engine integration. Timings are captured at 720p and 100% pixel coverage and we include performance statistics for three separate single-lobe methods: for [Arvo 1995; Lecocq et al. 2016], we use a Phong exponent that matches an average GGX roughness of 0.5.

better mappings, in the context of the ZHF matrix conditioning. As the SH order increases, ZHF matrix conditioning becomes more of a numerical issue, but for our applications the effect of different lobe direction settings was marginal. Since we use lobe sharing, we need only choose  $d = 2l_{\max} + 1 = 2N - 1$  directions.

*High-frequency Integrands & Generality.* As with any technique that uses a Fourier basis like SH, our approach requires many SH coefficients for high-frequency signals. While our approach is general and broadly applicable to many problems, specialized techniques may outperform our approach for high-frequency integrands. For instance, we support a product of BRDF  $\times$  lighting  $\times$  portal while Bitterli et al. [2015] only support lighting  $\times$  portal, but their approach coupled with MIS may be a more practical solution if the BRDF contributes little to the variance and the environment map is extremely high frequency.

*Variance in Control Variates.* Control variates allow us to apply our integrator to more general Monte Carlo integration problems. As our results have shown, this can lead to variance reduction if our integral (e.g. product of BRDF and lighting) accounts for much of the variation of the full shading. There are, however, a few important caveats to consider: since control variates probabilistically estimate the *difference* between two integrals, a control variate's sample can take on negative values (unlike radiance). It is therefore important to not clamp negative values until all pixel samples have been averaged. Moreover, while both our method and standard MC converge with more samples, the distribution of the error across the image will be different. For instance, standard MC produces a noise-free result in fully occluded regions and—in general—a noisy result in partially occluded and fully illuminated regions. Our control variates solution, on the other hand, produces a noise-free (and unbiased for bandlimited integrands) result in fully illuminated regions and a noisy result in partially- and fully-occluded regions. While we show improvement in the total RMSE across an image for several scenes, the relative error for dark, mostly occluded regions may be perceptually more noticeable.

## 7 CONCLUSION

We presented a new numerical technique for integrating spherical harmonic expansions over polygonal domains, broadening the class of applications that benefited from related closed-form solutions for

polygonal integration. We directly apply our integration routine to problems in surface and volume shading, as well as discussing its extension to other families of rendering techniques (i.e., product importance sampling, lighting projection in PRT).

Using control variates, we extend the applicability of our technique to integrands with non-analytic terms, all while remaining competitive with specialized importance-sampling MC estimators. Our implementation is simple and efficient, and we provide full source code that integrates readily into existing offline and shader-based interactive renderers (only < 750 lines of commented code for the self-contained integration routine).

In future work, seeking higher-dimensional solutions to the nested line-area integrals that arise in participating media and subsurface scattering could be very interesting. Moreover, many interesting approximation problems can be solved (i.e. similarly to Lecocq et al. [2016] and Heitz et al. [2016]) to improve the performance and flexibility of our approach in the context of interactive rendering. Our current implementation is fast enough to admit interactive rendering on GPUs, with our video demonstrating a proof-of-concept shader implementation, but we are more interested in (much) faster approximations that remain suitable for our applications.

## ACKNOWLEDGMENTS

We thank Cyril Soler for his help addressing numerical stability issues in our harmonics evaluation, and Michael Morandi for initial prototyping.

## REFERENCES

- Thomas Annen, Jan Kautz, Frédo Durand, and Hans-Peter Seidel. 2004. Spherical harmonic Gradients for Mid-Range Illumination.. In *Rendering Techniques (Proc. EGSR)*. Eurographics Association.
- James R. Arvo. 1995. Applications of Irradiance Tensors to the Simulation of Non-Lambertian Phenomena. In *Proc. SIGGRAPH Annual Conference*.
- J. Beckmann, H. N. Mhaskar, and J. Prestin. 2012. Quadrature formulas for integration of multivariate trigonometric polynomials on spherical triangles. *Int. Journal on Geomathematics* (2012).
- J. Beckmann, H. N. Mhaskar, and J. Prestin. 2014. Local numerical integration on the sphere. *Int. Journal on Geomathematics* (2014).
- Benedikt Bitterli, Jan Novák, and Wojciech Jarosz. 2015. Portal-Masked Environment Map Sampling. *Comp. Graph. Forum (Proc. EGSR)* (2015). <https://doi.org/10.1111/cgf.12674>
- Min Chen and James Arvo. 2001. Simulating Non-Lambertian Phenomena Involving Linearly-Varying Luminaires. In *Rendering Techniques (Proc. EG Workshop on Rendering)*. 25–38.
- Min Chen and James R. Arvo. 2000. A Closed-Form Solution for the Irradiance due to Linearly-Varying Luminaires. In *Rendering Techniques (Proc. EG Workshop on*

- Rendering). 137–148.
- Petrik Clarberg and Tomas Akenine-Möller. 2008a. Exploiting Visibility Correlation in Direct Illumination. *Comp. Graph. Forum (Proc. EGSR)* (2008). <https://doi.org/10.1111/j.1467-8659.2008.01250.x>
- Petrik Clarberg and Tomas Akenine-Möller. 2008b. Practical Product Importance Sampling for Direct Illumination. *Comp. Graph. Forum (Proc. Eurographics)* 27, 2 (2008), 681–690.
- Petrik Clarberg, Wojciech Jarosz, Tomas Akenine-Möller, and Henrik Wann Jensen. 2005. Wavelet Importance Sampling: Efficiently Evaluating Products of Complex Functions. *ACM Trans. Graph. (Proc. SIGGRAPH)* 24 (2005). <https://doi.org/10.1145/1073204.1073328>
- Ronald Cools and Philip Rabinowitz. 1993. Monomial cubature rules since “Stroud”: a compilation. *Journal of Computational & Applied Mathematics* (1993).
- Shaohua Fan, Stephen Chenney, Bo Hu, Kam-Wah Tsui, and Yu-Chi Lai. 2006. Optimizing Control Variate Estimators for Rendering. *Computer Graphics Forum (Proceedings of Eurographics 2006)* (2006).
- Q. T. Le Gia and H. N. Mhaskar. 2009. Localized Linear Polynomial Operators and Quadrature Formulas on the Sphere. *SIAM J. Numer. Anal.* 47, 1 (2009).
- Paul Green, Jan Kautz, Wojciech Matusik, and Frédo Durand. 2006. View-dependent precomputed light transport using nonlinear Gaussian function approximations. In *Symp. on Interactive 3D Graphics & Games*. ACM.
- Charles Han, Bo Sun, Ravi Ramamoorthi, and Eitan Grinspun. 2007. Frequency Domain Normal Map Filtering. *ACM Trans. Graph. (Proc. SIGGRAPH)* 26, 3 (2007), 28:1–28:12.
- Eric Heitz, Jonathan Dupuy, Stephen Hill, and David Neubelt. 2016. Real-Time Polygonal-Light Shading with Linearly Transformed Cosines. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2016).
- Kerstin Hesse, Ian H. Sloan, and Robert S. Womersley. 2010. *Handbook of Geomatics*. Chapter Numerical Integration on the Sphere.
- R. W. James. 1969. Transformation of Spherical Harmonics Under Change of Reference Frame. *Geophysical J. of the Royal Astronomical Society* (1969).
- Wojciech Jarosz, Nathan A. Carr, and Henrik Wann Jensen. 2009. Importance Sampling Spherical Harmonics. *Comp. Graph. Forum (Proc. Eurographics)* (2009). <https://doi.org/10.1111/j.1467-8659.2009.01398.x>
- Boz Kempfski. 1995. *Extension of the Whittaker-Shannon Sampling Series Aided by Symbolic Computation*. Ph.D. Dissertation. Anglia Polytechnic.
- Anders Wang Kristensen, Tomas Akenine-Möller, and Henrik Wann Jensen. 2005. Precomputed Local Radiance Transfer for Real-time Lighting Design. In *ACM Trans. Graph. (Proc. SIGGRAPH)*.
- Eric P. LaFortune and Yves D. Willems. 1994. The ambient term as a variance reducing technique for Monte Carlo ray tracing. In *Photorealistic Rendering Techniques (Proc. EG Workshop on Rendering)*. Darmstadt, Germany, 168–176.
- Pascal Lecocq, Arthur Dufay, Gaël Sourimant, and Jean-Eudes Marvie. 2016. Accurate Analytic Approximations for Real-time Specular Area Lighting. In *Symp. on Interactive 3D Graphics & Games*.
- Christian Lessig, Tyler de Witt, and Eugene Fiume. 2012. Efficient and accurate rotation of finite spherical harmonics expansions. *J. Comput. Phys.* 231, 2 (2012), 243–250. <https://doi.org/10.1016/j.jcp.2011.09.014>
- Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. 2003. A Data-Driven Reflectance Model. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2003).
- Soham Mehta, Ravi Ramamoorthi, Mark Meyer, and Christophe Hery. 2012. Analytic Tangent Irradiance Environment Maps for Anisotropic Surfaces. *Comp. Graph. Forum (Proc. EGSR)* (2012).
- Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. 2003. All-frequency shadows using non-linear wavelet lighting approximation. In *ACM Trans. Graph. (Proc. SIGGRAPH)*, Vol. 22. ACM.
- Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. 2004a. Triple product wavelet integrals for all-frequency relighting. In *ACM Trans. Graph. (Proc. SIGGRAPH)*, Vol. 23. ACM, 477–487.
- Ren Ng, Ravi Ramamoorthi, and Pat Hanrahan. 2004b. Triple product wavelet integrals for all-frequency relighting. *ACM Transactions on Graphics (TOG)* 23, 3 (2004), 477–487.
- Jan Novák, Andrew Selle, and Wojciech Jarosz. 2014. Residual Ratio Tracking for Estimating Attenuation in Participating Media. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 33, 6 (Nov. 2014). <https://doi.org/10.1145/2661229.2661292>
- Derek Nowrouzezahrai, Patricio Simari, and Eugene Fiume. 2012. Sparse Zonal Harmonic Factorization for Efficient SH Rotation and Shading. *ACM Transactions on Graphics* (2012). <https://doi.org/10.1145/2167076.2167081>
- Vincent Pegoraro and Steven G. Parker. 2009. An Analytical Solution to Single Scattering in Homogeneous Participating Media. *Comp. Graph. Forum (Proc. Eurographics)* 28, 2 (2009).
- Vincent Pegoraro, Mathias Schott, and Steven G. Parker. 2009. An Analytical Approach to Single Scattering for Anisotropic Media and Light Distributions. In *Proc. of Graphics Interface*.
- Matt Pharr, Wenzel Jakob, and Greg Humphreys. 2016. *Physically Based Rendering: From Theory To Implementation* (3rd ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Ravi Ramamoorthi and Pat Hanrahan. 2001. An Efficient Representation for Irradiance Environment Maps. In *Proc. SIGGRAPH Annual Conference*. ACM, 497–500. <https://doi.org/10.1145/383259.383317>
- Ravi Ramamoorthi and Pat Hanrahan. 2002. Frequency Space Environment Map Rendering. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2002). <https://doi.org/10.1145/566654.566611>
- Peter-Pike Sloan. 2008. Stupid spherical harmonics (SH) tricks. In *Game Developers Conference*.
- Peter-Pike Sloan, Jan Kautz, and John Snyder. 2002. Precomputed Radiance Transfer for Real-time Rendering in Dynamic, Low-frequency Lighting Environments. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2002). <https://doi.org/10.1145/566654.566612>
- Peter-Pike Sloan, Ben Luna, and John Snyder. 2005. Local, deformable precomputed radiance transfer. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2005).
- Cyril Soler, Mahdi M. Bagher, and Derek Nowrouzezahrai. 2015. Efficient and Accurate Spherical Kernel Integrals using Isotropic Decomposition. *ACM Transactions on Graphics* 34, 6 (Dec. 2015).
- Bo Sun, Ravi Ramamoorthi, Srinivasa G. Narasimhan, and Shree K. Nayar. 2005. A practical analytic single scattering model for real time rendering. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2005).
- Weifeng Sun and Amar Mukherjee. 2006. Generalized wavelet product integral for rendering dynamic glossy objects. *ACM Trans. Graph. (Proc. SIGGRAPH)* 25, 3 (2006), 955–966. <https://doi.org/10.1145/1141911.1141980>
- Yu-Ting Tsai and Zen-Chung Shih. 2006. All-frequency precomputed radiance transfer using spherical radial basis functions and clustered tensor approximation. *ACM Trans. Graph. (Proc. SIGGRAPH)* (2006).
- Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. 2009. All-frequency Rendering of Dynamic, Spatially-varying Reflectance. *ACM Trans. Graph. (Proc. SIGGRAPH Asia)* 28, 5 (2009). <https://doi.org/10.1145/1618452.1618479>

Received May 2016; revised November 2017; accepted December 2017