Monte Carlo Techniques for Direct Lighting Calculations

To Appear in ACM Transactions on Graphics

Peter Shirley
Program of Computer Graphics
580 Engineering and Theory Center
Cornell University
Ithaca, NY 14853

Changyaw Wang
Alias Research Inc.
110 Richmond Street East
Toronto, Canada M5C 1P1

Kurt Zimmerman
Department of Computer Science
Lindley Hall
Indiana University
Bloomington, IN 47405

Abstract

In a distribution ray tracer, the crucial part of the direct lighting calculation is the sampling strategy for shadow ray testing. Monte Carlo integration with importance sampling is used to carry out this calculation. Importance sampling involves the design of integrand-specific probability density functions which are used to generate sample points for the numerical quadrature. Probability density functions are presented that aid in the direct lighting calculation from luminaires of various simple shapes. A method for defining a probability density function over a set of luminaires is presented that allows the direct lighting calculation to be carried out with one sample, regardless of the number of luminaires.

CR Categories and Subject Descriptors: G.1.4 [Mathematical Computing]: Quadrature and Numerical Differentiation; I.3.0 [Computer Graphics]: General; I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism.

Additional Key Words and Phrases: direct lighting, importance sampling, luminaires, Monte Carlo integration, ray tracing, realistic image synthesis.

1 Introduction

Monte Carlo Methods have been used extensively for image rendering since Cook et al.'s landmark paper [7] on distribution ray tracing. Monte Carlo techniques have been used in classic ray tracing (view-dependent) methods [22, 6, 16, 18, 21, 19, 30], zonal (view-independent) methods [23, 35], and hybrid techniques [1, 39, 3, 20, 29]. Most recent implementations have handled the direct lighting[1] using Monte Carlo integration in a view-dependent manner, and have stored the indirect lighting within some kind of spatial lookup table [47, 29]. This paper extends traditional Monte Carlo methods for carrying out the direct lighting calculation so that they can be applied to scenes with hundreds or thousands of luminaires (light emitting objects). Our method is designed to work with tens to thousands of samples for each numerical quadrature, and technically this is a "low sampling density", meaning that the results of traditional asymptotic analysis are not directly applicable. Instead, we follow the advice of Spanier and Maize [40] and focus on the design of probability

[1] Lighting is often separated into three components in image synthesis programs: *emitted* (or *self-emitted*) lighting that originates at an object (called a *luminaire*), *direct* lighting that consists of light emitted from a luminaire and reflected from exactly one surface before reaching the viewer, and *indirect* lighting that consists of light emitted from a luminaire and reflects from at least two surfaces before reaching the viewer.

density functions that are well-suited to the particular characteristics of the integrands that arise in direct lighting calculations.

As shown by Kajiya [16], the radiance [15, 27] at a point in a scene can be written down as an integral equation over the set of all points in the environment[2]. The direct lighting component of this radiance is a two-dimensional integral (as opposed to an integral equation) over the set of all points in the scene. As shown by Cook et al. [7], this direct lighting integral is a natural target for Monte Carlo integration [13, 17]. A difficult question in any application of Monte Carlo integration is what probability density function should be used to generate sample points on the domain of integration. This is especially difficult in the direct lighting calculation where the domain is typically all surfaces in a complex three-dimensional environment. The selection of a *good* density function (i.e. one that produces a low variance solution) is the key step in applying Monte Carlo integration. In this paper, we attempt to find good probability density functions to be used in the direct lighting calculation. The main contributions of this paper are a careful discussion of the design of probability density functions for single luminaires, and a method for the construction of a single probability density function over a set of luminaires, so that only one shadow ray need be shot per viewing ray, regardless of the number of luminaires in the scene. This makes ray tracing feasible for scenes with many luminaires. This method is most suitable for images where multiple samples will be taken in each pixel. If the number of samples per pixel is very low (e.g. four samples), then a culling method such as Ward's [45] would be preferable to the method described in this paper.

In Section 2 we state the direct lighting integral, and describe basic Monte Carlo integration. In Section 3 we design probability density functions for shapes commonly used to approximate luminaires. In Section 4, we describe how to design a probability density function over large sets of luminaires. In Section 5, we speculate on how the techniques presented in this paper will fit into future rendering systems.

2 Mathematical Formulation

In this section the basic Monte Carlo solution method for definite integrals is outlined, and Monte Carlo integration is applied to the direct lighting integral. Most of the basic Monte Carlo material of this section is also covered in several of the classic texts [13, 38, 12, 49]. This section differs by being geared toward classes of problems that crop up in realistic image synthesis.

2.1 Monte Carlo Integration

Suppose we have a real valued function $h : \mathcal{S} \to \mathbb{R}$, where $\mathcal{S}$ is a possibly multidimensional space, and we wish to estimate the expected value of $h(\psi)$, where ψ is a random variable with probability density function $\wp : \mathcal{S} \to \mathbb{R}^+$ (denoted $\psi \sim \wp$). That $\wp$ is a probability density function on $\mathcal{S}$ means that $(\mu(\wp) = 1)$[3] (where μ is a measure defined on $\mathcal{S}$). From the definition of expected value, we can approximate the expected value of $h(\psi)$, $E[h(\psi)]$, as a sum:

$$E[h(\psi)] = \int_{\mathcal{S}} h(\psi')\wp(\psi')\, d\mu(\psi') \approx \frac{1}{N}\sum_{i=1}^{N} h(\psi_i) \tag{1}$$

The samples ψ_i are a set of N instantiations of the random variable ψ. We need this formulation with arbitrary dimension and measure, rather than the one-dimensional discussion found in most Monte Carlo texts, because we will integrate over surface areas and solid angles which do not have the simple measure of length. The form of Equation 1 is a bit awkward; we would usually like to approximate an integral of a

[2] This transport equation only applies to steady-state scenes made up of surfaces (with vacuum between them) obeying geometrical optics. In practice, this is a reasonable approximation to the human-observable behavior of visible light for many scenes [10].

[3] The notation $\mu(\wp) = 1$ means that the measure of $\wp$ is one, and is equivalent to the notation $\int_{\mathcal{S}} \wp(\psi) d\mu(\psi) = 1$. We will occasionally use the symbol $d\mu(\psi)$ in isolation. Although this technically violates measure theoretic notation, we follow many authors by indulging in this violation for convenience.

$\mathbb{R}$	The set of real numbers: $(-\infty, +\infty)$.
$\mathbb{R}^+$	The set of non-negative real numbers: $[0, +\infty)$.
$\mathcal{S}$	An unspecified space used for examples.
μ	An unspecified measure defined on $\mathcal{S}$ used for examples.
$\wp(\psi)$	Some probability density function defined on $\mathcal{S}$ using measure μ.
ξ	A canonical random number: uniformly random in $[0, 1)$.
$f : \mathcal{A} \to \mathcal{B}$	A function that maps elements of set $\mathcal{A}$ to elements of set $\mathcal{B}$
$E(a)$	Expected value of random variable a.
$V(a)$	Variance of random variable a.
N	The number of samples taken in a Monte Carlo integration.
$\mathcal{X}$	The set of all points on surfaces.
$\mathbf{x}$	A point in $\mathcal{X}$ being illuminated.
$\mathbf{x}'$	A point in $\mathcal{X}$ s.t. $\mathbf{x}' = \mathbf{x} + t(-\hat{\omega}')$ with $t = \|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|$.
A	The Area measure.
$\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|$	The distance between points $\mathbf{x}$ and $\mathbf{x}'$.
$\hat{n}$	The unit surface normal direction at $\mathbf{x}$.
$\hat{n}'$	The unit surface normal direction at $\mathbf{x}'$.
Ω	The hemisphere of directions with pole $\hat{n}$ (i.e. for all $\hat{\omega} \in \Omega$, $\hat{\omega} \cdot \hat{n} > 0$).
$\mho$	The hemisphere of directions with pole $-\hat{n}$ (i.e. for all $\hat{\omega}' \in \mho$, $\hat{\omega}' \cdot \hat{n} < 0$).
σ	The solid angle measure.
$\hat{\omega}$	A direction (unit vector) in Ω from which $\mathbf{x}$ is viewed.
$\hat{\omega}'$	A direction (unit vector) in $\mho$ that is parallel to $\overrightarrow{\mathbf{x} - \mathbf{x}'}/\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|$.
$\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$	The spectral bidirectional reflectance distribution function (BRDF).
$g(\mathbf{x}, \mathbf{x}')$	Visibility function: one if no surface is between $\mathbf{x}$ and $\mathbf{x}'$ zero otherwise.
$L_s(\mathbf{x}, \hat{\omega})$	Surface spectral radiance at point $\mathbf{x}$ in direction $\hat{\omega}$.
$L_f(\mathbf{x}, \hat{\omega}')$	Field (incident) spectral radiance at point $\mathbf{x}$ coming from direction $\hat{\omega}'$.
$L_e(\mathbf{x}, \hat{\omega})$	Surface spectral radiance at point $\mathbf{x}$ in direction $\hat{\omega}$ emitted from $\mathbf{x}'$ (emitted lighting).
$L_d(\mathbf{x}, \hat{\omega})$	Surface spectral radiance at point $\mathbf{x}$ in direction $\hat{\omega}$ caused by light coming directly from a luminaire (direct lighting).
$p(\mathbf{x}')$	Some probability density function defined on $\mathcal{X}$ using measure A.
$q(\hat{\omega}')$	Some probability density function defined on Ω using measure σ.
N_L	The number of luminaires in the environment.
ℓ_i	The ith luminaire.
$\rho_{\max}$	The maximum BRDF of a point over all $\hat{\omega}$ and $\hat{\omega}'$.
$\mathcal{L}$	The set of all luminaires ℓ_i.
$\mathcal{L}_{\text{dim}}$	The set of "unimportant" luminaires.
$\mathcal{L}_{\text{bright}}$	The set of "important" luminaires.
$\hat{L}_d$	An estimate of the average L_i for $\ell_i \in \mathcal{L}_{\text{dim}}$.
N_d	The number of luminaires in $\mathcal{L}_{\text{dim}}$.
N_b	The number of luminaires in $\mathcal{L}_{\text{bright}}$.
L_i	The radiance conribution from ℓ_i at some $(\mathbf{x},\hat{\omega})$.
$\hat{L}_i$	An extimate of L_i.
LO	A type of function that has "low variation" so is roughly constant.
HI	A type of function that has "high variation" so is not roughly constant.
UN	A type of function that is unknown, but can be evaluated at points.

Table 1: Some important symbols and terms used in this paper.

single function f rather than a product $h\wp$. We can get around this by substituting $f = h\wp$ as the integrand:

$$\int_{\mathcal{S}} f(\psi')\, d\mu(\psi') \approx \frac{1}{N} \sum_{i=1}^{N} \frac{f(\psi_i)}{\wp(\psi_i)} \tag{2}$$

For this formula to be valid, $\wp$ must be positive where f is nonzero. The variance of the sum in Equation 2 is:

$$V\left[\frac{1}{N} \sum_{i=1}^{N} \frac{f(\psi_i)}{\wp(\psi_i)}\right] = \frac{1}{N} V\left[\frac{f}{\wp}\right] \tag{3}$$

Equation 3 implies that to get a good (low variance) estimate, we want as many samples as possible (i.e. N is large), and we want the density $f/\wp$ to have a low variance. Choosing $\wp$ intelligently is called importance sampling, because if $\wp$ is relatively large where f is relatively large, there will be more samples in important regions. Equation 3 also shows the fundamental problem with Monte Carlo integration: *diminishing return*. Because the variance of the estimate is proportional to $1/N$, the standard deviation is proportional to $1/\sqrt{N}$. Since the error in the estimate behaves similarly to the standard deviation, we will need to approximately quadruple n to halve the error.

Another way to reduce variance is to partition $\mathcal{S}$, the domain of the integral, into N several smaller domains $\mathcal{S}_i$, and evaluate the integral as a sum of integrals over the $\mathcal{S}_i$. This is called stratified sampling. Normally only one sample is taken in each $\mathcal{S}_i$ (with density $\wp_i$), and in this case the variance of the estimate is:

$$V\left[\sum_{i=1}^{N} \frac{f(\psi_i)}{\wp_i(\psi_i)}\right] = \sum_{i=1}^{N} V\left[\frac{f(\psi_i)}{\wp_i(\psi_i)}\right] \tag{4}$$

Typically, a density $\wp$ on $\mathcal{S}$ will be chosen and $\wp_i$ will be made proportional to $\wp$:

$$\wp_i(\psi) = \frac{\wp(\psi)}{\int_{\mathcal{S}_i} \wp(\psi') d\mu(\psi')} \tag{5}$$

There are some functions where stratification does no good. An example is a white noise function, where the variance is constant for all regions. Intuitively, stratification will pay off when f varies slowly relative to the distance between adjacent sample points.

Although distribution ray tracing is usually phrased as an application of Equation 2, many researchers replace the ψ_i with more evenly distributed (quasi-random) samples (e.g. [6, 24, 31]). This approach can be shown to be sound by analyzing decreasing error in terms of some discrepancy measure [50, 48, 24, 34] rather than in terms of variance. However, in practice it is often convenient to develop a sampling strategy using variance analysis on random samples, and then to turn around and use non-random, but equidistributed samples in an implementation. This approach is almost certainly correct, but its justification and implications have yet to be fully explained. For the rest of this paper we will assume unstratified random sample points for our derivations, but in our implementation we have used stratified random samples (which should never produce higher variance estimates than unstratified random samples would produce).

2.2 Generating Random Samples

There are many ways to generate random samples with specified density functions. Most of these methods assume we have the ability to generate a sequence of *canonical* random numbers $(\xi_1, \xi_2, \xi_3, \ldots)$, where each ξ_i is an identically distributed uniform random number between zero and one (more formally, $\varsigma : [0, 1) \to \mathbb{R}^+$ is the canonical probability density function, meaning for all $y \in [0, 1)$ that $\varsigma(y) = 1$, and ξ being a canonical random number means $\xi \sim \varsigma$). For example, to choose uniform random points in the ball with unit radius centered at the origin, we can pick a uniform point r in the cube with sidelength two centered at the origin, where $r = (2\xi_i - 1, 2\xi_{i+1} - 1, 2\xi_{i+2} - 1)$, and check whether r is inside the ball. If it is not in the ball, we repeat the process with $r = (2\xi_{i+2} - 1, 2\xi_{i+3} - 1, 2\xi_{i+4} - 1)$, and so on, until we find a point in the ball. This is a so called *rejection* method, and it is very useful for picking random points in spaces with complex boundaries.

Unfortunately, it is often not as obvious how to apply a rejection method for nonuniform densities, and a rejection method is not easy to combine with stratified sampling.

A method that does not suffer from the problems of the rejection method is to pass the canonical numbers through a function. To illustrate why this will work, notice that for some function $f : [0, 1) \to \mathbb{R}$, and a canonical random number ξ, $f(\xi)$ is a random variable with a possibly nonuniform density function. For example, the quantity $f(\xi) = \xi^2$ is more likely to have a value near zero than one, so it must have a nonuniform density function. We can reverse-engineer what $f(x)$ must be to get a sample with desired density function $\wp : [\psi_0, \psi_1] \to \mathbb{R}$. This will be $\mathcal{P}^{-1}(\psi)$ where $\mathcal{P} : [\psi_0, \psi_1] \to [0, 1]$ is given by

$$\mathcal{P}(\psi) = \int_{\psi_0}^{\psi} \wp(\psi')d\psi' \tag{6}$$

The function $\mathcal{P}$ is often called the *cumulative probability distribution function* associated with the probability density function $\wp$ because $\mathcal{P}(\psi)$ is the probability that a random variable with density $\wp$ has a value less than or equal to ψ. This basic method has been in use since at least the 1950s [13], and was first used in image synthesis by Ward et al. to generate reflection rays with a cosine density [47]. Note that $\mathcal{P}^{-1}$ will always exist, but may not be analytical, in which case numerical function inversion must be used.

One nice thing about generating a sequence of nonuniform random numbers using the inverse of the cumulative distribution function is that the ordering of the ξ_i and the $\mathcal{P}^{-1}(\xi_i)$ will be the same, and thus if we input stratified ξ_i instead of a completely random sequence, then $\mathcal{P}^{-1}(\xi_i)$ will also be stratified.

For a two dimensional probability density function $\wp : [\psi_0, \psi_1] \times [\tau_0, \tau_1] \to \mathbb{R}^+$, where $\wp$ is defined with respect to measure $d\mu(\psi, \tau) = m(\psi, \tau)d\psi\, d\tau$, we can do a similar transformation by using the marginal density function $\wp_\Psi : [\psi_0, \psi_1] \to \mathbb{R}^+$:

$$\wp_\Psi(\psi) = \int_{\tau_0}^{\tau_1} \wp(\psi, \tau')m(\psi, \tau')d\tau' \tag{7}$$

The function $\wp_\Psi$ is a valid density function that represents the density of ψ values without reference to τ. Once a ψ value has been chosen with $\psi_i = \mathcal{P}_\Psi^{-1}(\xi_i)$, the τ value can be chosen with density

$$\wp_{T|\Psi}(\tau|\psi_i) = \frac{\wp(\psi_i, \tau)}{\wp_\Psi(\psi)}, \tag{8}$$

where $\wp_{T|\Psi}(\tau|\psi_i)$ is the conditional probability of τ given ψ_i. In the special case where $\wp m$ is separable, meaning $\wp(\psi, \tau)m(\psi, \tau)$ can be expressed as $b_1(\psi)b_2(\tau)$, where both $b_1 : [\psi_0, \psi_1] \to \mathbb{R}^+$ and $b_2 : [\tau_0, \tau_1] \to \mathbb{R}^+$ are probability density functions, then we can chose $\psi_i = B_1^{-1}(\xi_i)$ and $\tau_i = B_2^{-1}(\xi_{i+1})$, where B_1 is the cumulative probability distribution function associated with b_1, and B_2 is the cumulative probability distribution function associated with b_2.

2.3 Design Strategies for Importance Sampling

The key step in an implementation of Monte Carlo integration is the choice of the probability density function of the sample points. This is something of a "black art" because the goal is not to have the lowest variance for a given number of samples, but is instead to have the lowest variance for a given execution time. This is an important distinction because samples that produce low variance may take many times longer to generate than higher variance samples. In this section we outline the general rules we have used when choosing our density functions for sampling. We will apply these rules when estimating the values of specific integrals in Section 3.

The integrals that arise in lighting calculations are often of the form

$$I = \int_{\mathcal{S}} f_1(\psi)f_2(\psi)\ldots f_\ell(\psi)d\mu(\psi) \tag{9}$$

where the f_i are strictly non-negative functions. The f_i may vary a great deal in both shape and execution cost. They may be known a priori or their character may only be attained through point sampling. In

this section we discuss several strategies for the design of density functions for Monte Carlo integration of integrals of the form of Equation 9

First we will categorize the f_i into one of the following general behaviors:

Known, low variation (LO) f_i is known a priori and has low variation over $\mathcal{S}$.

Known, high variation (HI) f_i is known a priori and has high variation over $\mathcal{S}$.

Unknown (UN) f_i is known only by sampling.

By grouping terms in Equation 9 we can rewrite it as a product of at most three functions of the types listed above (note that grouping two terms of type **HI** could result in one term of type **LO**, so this grouping must be done carefully). The integrals in this paper will always contain an integral of type **UN** (usually evaluated by tracing a ray), and some weighting function (usually representing local properties of the surface) that is a combination of functions of type **LO** and **HI**. We consider several cases below.

If the integrand is a product of a function f_{HI} of type **HI** and a function f_{UN} of type **UN**, then ideally we would set $\wp(\psi) \propto f_{HI}(\psi)f_{UN}(\psi)$. Since f_{UN} is unknown we might evaluate it at several locations and approximate it, but in image synthesis applications this is not usually practical because evaluations are so expensive, and f_{UN} may be quite complex. If we can assume nothing about f_{UN}, we might still want to make $\wp(\psi) \propto f_{HI}(\psi)$, which is often done in practice. This is in some limited sense optimal, as is shown in Appendix 5.

If the integrand is a product of a function f_{LO} of type **LO** and a function f_{UN} of type **UN**, we might follow the logic of the $f_{HI}(\psi)f_{UN}(\psi)$ case, but we might also just want to set $\wp(\psi)$ to be a constant. This is because the improvements in variance caused by a choice of $\wp(\psi) \propto f_{LO}(\psi)$ would probably be made inconsequential by the large variance that $f_{UN}(\psi)$ can cause. We certainly do not want to bother making $\wp(\psi) \propto f_{LO}(\psi)$ if this would add much computation time to generating the random points with density $\wp$.

In summary, when the integrand is a known weighting function times an unknown expensive function, we let $\wp$ be proportional to the components of the weighting function which are not roughly constant. If this is too expensive, we make $\wp(\psi)$ proportional to an approximation to the components of the weighting function which are not roughly constant. More formally, we rewrite $f(\psi)$ as $f_{LO}(\psi)f_{HI}(\psi)f_{UN}(\psi)$ and try to find a way to generate samples $\psi_i \sim \wp$ with $\wp(\psi) \propto f_{HI}(\psi)$. If this is difficult we use $\wp(\psi) \propto \hat{f}_{HI}(\psi)$ where $\hat{f}_{HI}$ is an approximation of $f_{HI}(\psi)$. This is the general strategy we will use in later sections to choose density functions.

2.4 Direct Lighting

Here we will review the rendering equation that governs our approximation to light transport, and we will separate out the direct lighting integral that we will later solve with Monte Carlo integration. The rendering equation is usually written in one of two basic ways, one as an integral of radiometric quantities[4] over solid angles, and one as an integral equation of radiometric quantities over all surfaces. It can be written down in terms of all directions $\hat{\omega}'$ visible to $\mathbf{x}$ (as done by Immel [14]):

$$L_s(\mathbf{x}, \hat{\omega}) = L_e(\mathbf{x}, \hat{\omega}) + \int_{\mho} \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')L_f(\mathbf{x}, \hat{\omega}')(-\hat{\omega}' \cdot \hat{n})d\sigma(\hat{\omega}') \tag{10}$$

where $L_s(\mathbf{x}, \hat{\omega})$ is the surface radiance[5] of $\mathbf{x}$ in direction $\hat{\omega}$, $L_e(\mathbf{x}, \hat{\omega})$ is the emitted surface radiance in direction $\hat{\omega}$ at $\mathbf{x}$, $\mho$ is the unit hemisphere of incoming directions oriented about $\hat{n}$, $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ is the BRDF

[4] In this paper we use the a slight variant of the radiometric terms and symbols standardized by the Illumination Engineering Society (IES) [15, 27]. To simplify notation, we drop the spectral term even though all of the radiometric quantities we use are spectral quantities. So to denote spectral radiance we use L rather than L_λ recommended by the IES.

[5] Radiance is a quantity that is defined for all points in space, directions $\omega \in \bigcirc$, and wavelengths $\lambda \in \Lambda$ in a homogeneous material and can be though of as a function $L : \mathbb{R}^3 \times \bigcirc \times \Lambda \rightarrow \mathbb{R}$. This function is not defined at boundaries between materials (i.e. at surfaces) because the light abruptly changes direction [32]. To solve this problem, authors commonly refer to "incoming" and "outgoing" radiance. Because the outgoing radiance is what we perceive when we look at a surface, the outgoing radiance is sometime called "surface radiance", and the incoming radiance from the "field" is called "field radiance" [2]. For transparent surfaces, both the surface and field radiances are defined on all directions in $\bigcirc$. For opaque surfaces, the surface radiance $L_s(\mathbf{x}, \hat{\omega})$ is defined on $\hat{\omega}$ in the outgoing hemisphere of directions Ω, and the field radiance $L_f(\mathbf{x}, \hat{\omega}')$ is defined on $\hat{\omega}'$ in the incoming hemisphere of directions $\mho$.

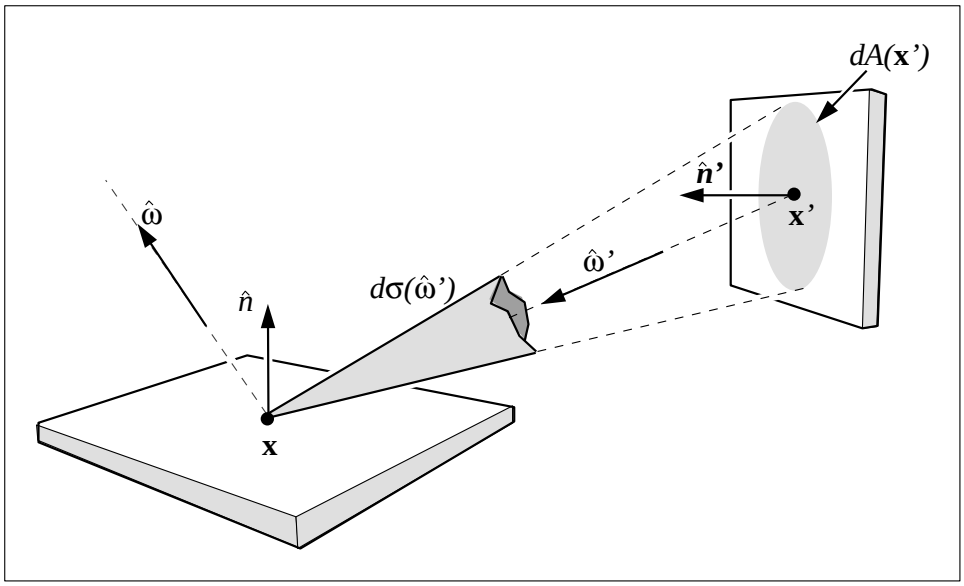

Figure 1: Geometry for the rendering equation

at $\mathbf{x}$, $L_f(\mathbf{x}, \hat{\omega}')$ is the field radiance from direction $\hat{\omega}'$ incident at $\mathbf{x}$, amd σ is the solid angle measure. The geometric quantities are illustrated in Figure 1.

Although Equation 10 has the form of an integral, the evaluation of the field radiance $L_f(\mathbf{x}, \hat{\omega}')$ will typically require the evaluation of $L_s(\mathbf{x}', \hat{\omega}')$ for some point $\mathbf{x}'$ "seen" by $\mathbf{x}$ in direction $-\hat{\omega}'$. This leads to other way the rendering equation is often written as integral equation over all surfaces (as done by Kajiya [16][6]):

$$L_s(\mathbf{x}, \hat{\omega}) = L_e(\mathbf{x}, \hat{\omega}) + \int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}') \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_s(\mathbf{x}', \hat{\omega}') (-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \, dA(\mathbf{x}') \tag{11}$$

where $\mathcal{X}$ is the set of all points on surfaces and $g(\mathbf{x}, \mathbf{x}')$ is the *geometry term*, which is zero if there is an obstruction between $\mathbf{x}$ and $\mathbf{x}'$ and one otherwise, $\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|$ is the distance between $\mathbf{x}$ and $\mathbf{x}'$, and A is the area measure.

The direct lighting, $L_d(\mathbf{x}, \hat{\omega})$, can be written down as an integral over all directions by dropping the reflected light at $\mathbf{x}'$ in Equation 10:

$$L_d(\mathbf{x}, \hat{\omega}) = \int_{\mho} \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}') (-\hat{\omega}' \cdot \hat{n}) d\sigma(\hat{\omega}') \tag{12}$$

where $\mathbf{x}'$ is the point seen by $\mathbf{x}$ in direction $-\hat{\omega}'$. In a program a ray $(\mathbf{x} + t(-\hat{\omega}'))$ will be traced from $\mathbf{x}$ in direction $-\hat{\omega}'$ to find $\mathbf{x}'$. The direct lighting can also be written down as an integral over all points by dropping the reflected light at $\mathbf{x}'$ from Equation 11:

$$L_d(\mathbf{x}, \hat{\omega}) = \int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}') \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}') (-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \, dA(x') \tag{13}$$

Here the geometry term $g(\mathbf{x}, \mathbf{x}')$ is calculated by sending the "shadow" ray $(\mathbf{x} + t(\overrightarrow{\mathbf{x}' - \mathbf{x}}))$ into the environment. The geometry term is one if $t \approx 1$ at the first surface the ray hits.

When Monte Carlo integration is used to approximate Equation 12, a probability density function $q(\hat{\omega}')$ (where q is defined using the solid angle measure σ) is built over $\mho$. Applying Equation 2 with $N = 1$ (one

[6] Technically, Kajiya's equation is not written in terms of radiance, but it can be mapped to Equation 11.

sample) gives:

$$L_d(\mathbf{x}, \hat{\omega}) \approx \frac{\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}')(-\hat{\omega}' \cdot \hat{n})}{q(\hat{\omega}')} \tag{14}$$

where $\hat{\omega}' \sim q$. Using Monte Carlo integration to approximate Equation 13 using a probability density function $p(\mathbf{x}')$ where p is defined on $\mathcal{X}$ using the area measure A yields:

$$L_d(\mathbf{x}, \hat{\omega}) \approx g(\mathbf{x}, \mathbf{x}') \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}')(-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{p(\mathbf{x}') \|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \tag{15}$$

where $\mathbf{x}' \sim p$.

All that must be done to actually implement the direct lighting calculation is to choose which integral we will evaluate (the directional integral or the integral over all points), create a density function, and somehow generate random samples according to that density. The details of these steps are described for direct lighting from a single luminaire in Section 3 and for multiple luminaires in Section 4.

3 Direct Lighting from One Luminaire

To calculate $L_d(\mathbf{x}, \hat{\omega})$ for a single luminaire using Monte Carlo integration, we can use Equation 14, with a density that is positive for the directions subtended by the luminaire, or we can use use Equation 15, with a probability density function that is positive at all visible points on the luminaire (in practice it will be positive for all points on the luminaire because the visibility is hard to establish a priori). For large luminaires (such as the sky), sampling from solid angles might make sense, but for luminaires that are small in angle space as seen from the points where we are calculating direct lighting (e.g. the Sun and most man-made luminaires), most of the integrand will be zero, and it will thus be hard to design good density functions. Therefore, we will use Equation 15 because it is usually easier to design densities on the surfaces themselves. For reflection rays from specular or near-specular surfaces it is usually more natural to use Equation 14, but we will not address reflection rays in this paper. The issues involved in choosing between Equations 14 and 15 are discussed in more detail in [37]. Several of the examples used in this section have been used by other authors (e.g. [18, 21, 25]) who use the same transformation strategies.

3.1 Designing Probability Density Functions

In designing a density function $p(\mathbf{x}')$ to use in Equation 15, we can immediately set $p(\mathbf{x}') = 0$ wherever $L_e(\mathbf{x}', \hat{\omega}') = 0$ (all points not on the luminaire). A perfect (zero variance) estimate would result from:

$$p(\mathbf{x}') = C g(\mathbf{x}, \mathbf{x}') \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}')(-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \tag{16}$$

where C is a normalization constant. As is usually pointed out in the Monte Carlo literature, the integral must be evaluated before we know C, so this is not very practical.

Instead, as was suggested in Section 2.3, we will attempt to partition the integrand into f_{LO}, f_{HI}, and f_{UN}. If we assume the luminaire is diffuse or has a radiance that does not vary much across the surface as seen by most directions, and we assume that $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ is roughly constant over the directions subtended by the luminaire, then our integrand can be classified as

$$\underbrace{\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}') L_e(\mathbf{x}', \hat{\omega}')}_{\text{LO}} \underbrace{(-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2}}_{\text{HI}} \underbrace{g(\mathbf{x}, \mathbf{x}')}_{\text{UN}} \tag{17}$$

So following the strategy of using the **HI** component of the function guide our design of p we get

$$p(\mathbf{x}') \propto (-\hat{\omega}' \cdot \hat{n}) \frac{(\hat{\omega}' \cdot \hat{n}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \tag{18}$$

Note that for some luminaires some of the components of Equation 18 might be roughly constant. For example, if the luminaire is small as seen from $\mathbf{x}$, then $(-\hat{\omega}'\cdot\hat{n})$ will be roughly constant. If the luminaire is also planar, then $(\hat{\omega}'\cdot\hat{n}')$ is roughly constant. If the luminaire's extent is small compared to its distance from $\mathbf{x}$, then $\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ is roughly constant. If we can move any of these terms into the **LO** component of the integrand, then it will make p easier to design. For the remainder of this section, we discuss how to choose p for several simple luminaire shapes. In particular, we discuss probability density functions for luminaires that are approximated by spheres, polygons, disks, and cylinders. We will use the case of spheres to illustrate the mechanics of choosing nonunifomly random points, and the case of triangles as an example how to deal with non-separable densities, and will then summarize results for the other shapes. More details on the derivations of the material in this section can be found in [44].

3.2 Sampling Spherical Luminaires

In this section we will discuss three different density functions that could be used to choose samples on a spherical luminaire with radius r and center $\mathbf{c} = (c_x, c_y, c_z)$. To classify the components of $(-\hat{\omega}'\cdot\hat{n})(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$, we first notice that even for small distant luminaires, $(\hat{\omega}'\cdot\hat{n}')$ varies from zero to one, so it is a **HI** component. The other two terms, $(-\hat{\omega}'\cdot\hat{n})$ and $\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ can be **LO** or **HI** depending on the parameters of the sphere and the location of $\mathbf{x}$. For example, if the sphere almost touches $\mathbf{x}$ then both terms are **HI**, but for a distant source such as the Sun, both terms are **LO**. So in different circumstances, different density functions are useful.

3.2.1 Uniform density

For the purposes of debugging, a constant density function is useful. Because the density function must have unit volume on the surface of the sphere, this density is just the inverse of the sphere area:

$$p_1(\mathbf{x}') = \frac{1}{4\pi r^2}$$

To generate a random point on the sphere, we define the density in a spherical coordinate system with θ being the polar angle (the positive $\hat{z}$ axis is $\theta = 0$), and ϕ being the azimuthal angle (the positive $\hat{x}$ axis is $\phi = 0$). This yields a density with a related differential measure $dA(\mathbf{x}') = r^2\sin\theta d\theta d\phi$. Using the techniques of Section 2.2, we separate the density $1/(4\pi r^2)$ and $r^2\sin\theta d\theta d\phi$ into two density functions: $p_a(\phi) = 1/(2\pi)$ and $p_b(\theta) = \sin\theta/2$. This gives us $(\theta,\phi) = (\arccos(1-2\xi_1), 2\pi\xi_2)$. Converting to cartesian coordinates gives: $\mathbf{x}' = [c_x + r\cos\phi\sin\theta, c_y + r\sin\phi\sin\theta, c_z + r\cos\theta]^T$ which can be simplified to

$$\mathbf{x}' = \begin{bmatrix} c_x + 2r\cos(2\pi\xi_2)\sqrt{\xi_1(1-\xi_1)} \\ c_y + 2r\sin(2\pi\xi_2)\sqrt{\xi_1(1-\xi_1)} \\ c_z + r(1-2\xi_1)) \end{bmatrix} \tag{19}$$

An immediate optimization one could do is to take samples only on the portion of the sphere visible to $\mathbf{x}$. However, because the uniform area case is ideal for debugging, we will instead leave it as is and move on to nonuniform densities.

3.2.2 Sampling uniformly in directional space

The first nonuniform density we might try is $p(\mathbf{x}') \propto (\hat{\omega}'\cdot\hat{n}')$. This turns out to be just as complicated as sampling with $p(\mathbf{x}') \propto (\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$, so we instead discuss that here. We observe that sampling on the luminaire using $(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ is the same as using a density constant function $q_2(\hat{\omega}')$ defined in the space of directions subtended by the luminaire as seen from $\mathbf{x}$. We now use a coordinate system defined with $\mathbf{x}$ at the origin, and a right-handed orthonormal basis with $\hat{w} = \overrightarrow{\mathbf{c}-\mathbf{x}}/\|\overrightarrow{\mathbf{c}-\mathbf{x}}\|$, and $\hat{v} = (\hat{w}\times\hat{n})/\|(\hat{w}\times\hat{n})\|$ (see

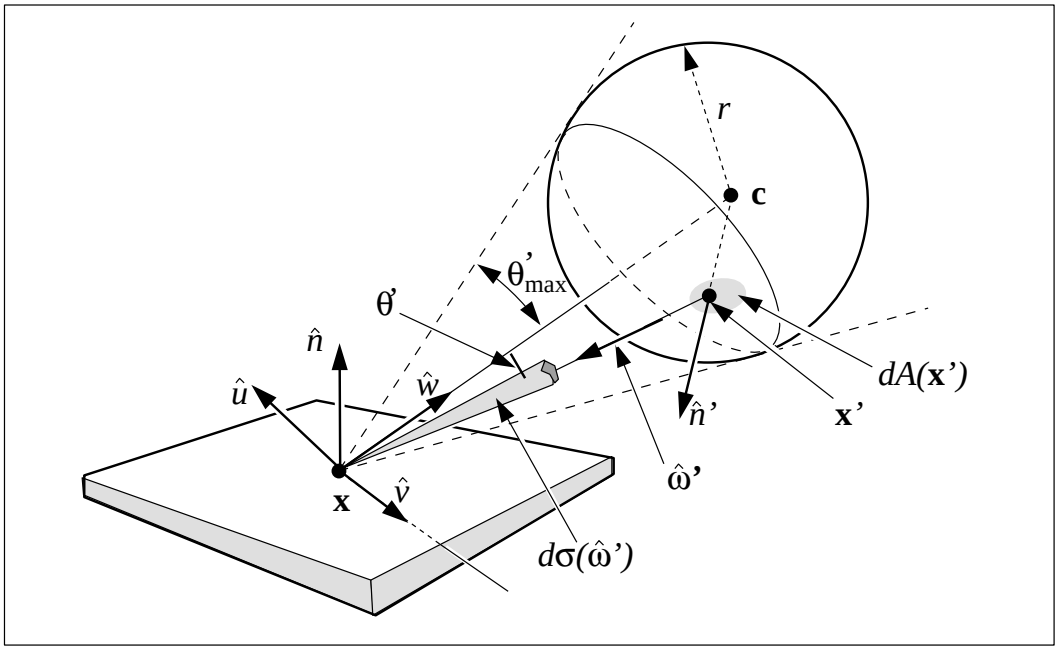

Figure 2: Geometry for spherical luminaire.

Figure 2). We also define (θ', ϕ') to be the azimuthal and polar angles with respect to the $\hat{u}\hat{v}\hat{w}$ coordinate system.

The maximum θ' that includes the spherical luminaire is given by:

$$\theta'_{\max} = \arcsin\left(\frac{r}{\|\overrightarrow{\mathbf{x}-\mathbf{c}}\|}\right) = \arccos\sqrt{1-\left(\frac{r}{\|\overrightarrow{\mathbf{x}-\mathbf{c}}\|}\right)^2}$$

Thus a uniform density (with respect to solid angle) within the cone of directions subtended by the sphere is just the reciprocal of the solid angle $2\pi(1-\cos\theta'_{\max})$ subtended by the sphere:

$$q_2(\hat{\omega}') = \frac{1}{2\pi\left(1-\sqrt{1-\left(\frac{r}{\|\mathbf{x}-\mathbf{c}\|}\right)^2}\right)}$$

And we get

$$\begin{bmatrix} \theta' \\ \phi' \end{bmatrix} = \begin{bmatrix} \arccos(1-\xi_1+\xi_1\sqrt{1-\left(\frac{r}{\|\mathbf{x}-\mathbf{c}\|}\right)^2}) \\ 2\pi\xi_2 \end{bmatrix}$$

This gives us the direction to $\mathbf{x}'$. To find the actual point, we need to find the first point on the sphere in that direction. The ray in that direction is just $(\mathbf{x}+t\hat{a})$, where $\hat{a}$ is given by:

$$\hat{a} = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \begin{bmatrix} \cos\phi'\sin\theta' \\ \sin\phi'\sin\theta' \\ \cos\theta' \end{bmatrix}$$

We must also calculate $p_2(\mathbf{x}')$, the probability density function with respect to the area measure (recall that the density function q_2 is defined in solid angle space). Since we know that q_2 is a valid probability density function using the σ measure, and we know that $d\sigma(\hat{\omega}') = dA(\mathbf{x}')(\hat{\omega}'\cdot\hat{n}')/\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^2$, we can relate

any probability density function $q(\hat{\omega}')$ with its associated probability density function $p(\mathbf{x}')$:

$$q(\hat{\omega}') = \frac{(\hat{\omega}' \cdot \hat{n}')p(\mathbf{x}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \tag{20}$$

So we can solve for $p_2(\mathbf{x}')$:

$$p_2(\mathbf{x}') = \frac{(\hat{\omega}' \cdot \hat{n}')}{2\pi\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2 \left(1 - \sqrt{1 - \left(\frac{r}{\|\mathbf{x}-\mathbf{c}\|}\right)^2}\right)}$$

For further details on sampling the sphere with p_2 see [43].

3.2.3 Cosine-Weighted Solid Angle:

To bring the $(-\hat{\omega}' \cdot \hat{n})$ term into the density function we again use the space of directions: $q_3(\hat{\omega}') \propto (-\hat{\omega}' \cdot \hat{n})$. Integrating this density function gives the constant of proportionality for the case that the entire sphere is above the horizon as seen from $\mathbf{x}$ (if this is not the case we must use a different density function because q_3 as stated above would have regions with negative values), and thus the density function is:

$$q_1(\hat{\omega}') = \frac{(-\hat{\omega}' \cdot \hat{n})}{\pi(\hat{w} \cdot \hat{n}) \sin^2 \theta_{\max}}$$

The direction $\hat{\omega}'$ can be found by in the following two formulas:

$$\xi_1 = \frac{\sin^2 \theta'}{\sin^2 \theta_{\max}}$$

and

$$\xi_2 = \frac{(\hat{u} \cdot \hat{n}) \sin\theta' \sin\phi' + (\hat{w} \cdot \hat{n})\phi' \cos\theta'}{2\pi(\hat{w} \cdot \hat{n}) \cos\theta'} \tag{21}$$

So we get $\theta' = \arcsin(\sqrt{\xi_1} \sin\theta_{\max})$ but, we must numerically invert the second formula to solve for ϕ'. For our implementation, we used a simple binary search to perform this inversion, which requires tens of evaluations of Equation 21 so it is quite costly (so in practice we do not usually use p_3 to sample spherical luminaires). Getting the actual point $\mathbf{x}'$ on the luminaire is handled in the same way as in Section 3.2.2, and the value of $p_3(\mathbf{x}')$ is found using Equation 20.

A spherical luminaire sampled with one sample per pixel for each of the three cases discussed in this section, along with a uniform sampling of the visible region of the sphere, is shown in Figure 3. Although the sphere sampled with p_3 generates the smoothest image, in practice we use p_2 because it is much less expensive than p_3, and because we use multiple samples for edge and shadow antialiasing, we can tolerate the variance caused by p_2.

3.3 Sampling Planar Luminaires

For any planar luminaire, the quantity $(-\hat{\omega}' \cdot \hat{n})(\hat{\omega}' \cdot \hat{n}')\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^{-2}$ stays roughly constant provided the luminaire subtends a relatively small solid angle at $\mathbf{x}$. So for planar luminaires, we use a constant density function to generate most of our images. In our implementation we have disks, triangles, rectangles, and general polygons.

3.3.1 Sampling Polygonal Luminaires

For simple polygons (polygons that have no holes) we have resorted to picking uniformly from the bounding rectangle of the polygon and using a rejection technique based on whether the sample is in the polygon. This test can be accomplished using a ray test in the plane of the polygon[11], which works for all polygons, even those with holes. This probability density function is just the inverse of the polygon area, which can be found using Stoke's Theorem [28].

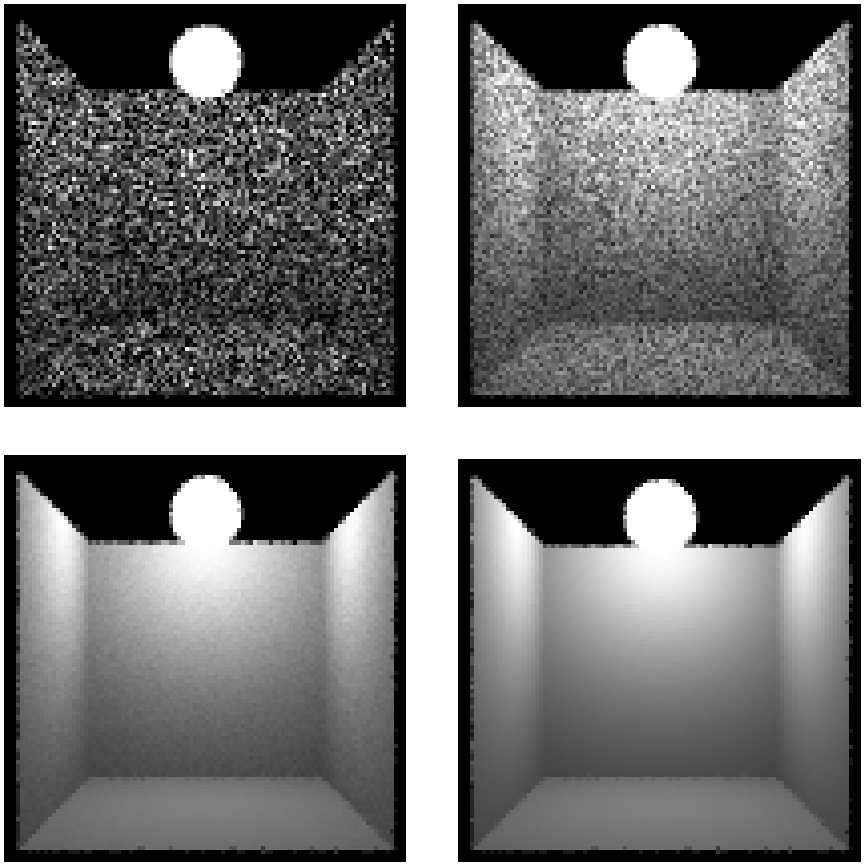

Figure 3: A spherical luminaire sampled using p_1 (top left), p_1' (uniform on visible portion; top right), p_2 (bottom left) and p_3 (bottom right).

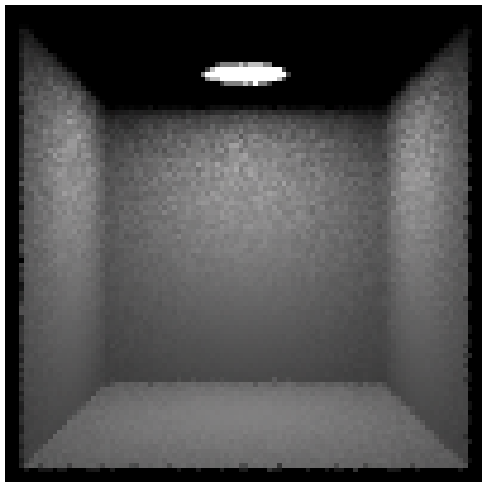

Figure 4: A disk luminaire sampled using p_4.

3.3.2 Sampling Disk Luminaires

To choose a random sample from a disk, we suppose its center is at the $\mathbf{c}$, the radius is r, and $\hat{n}' = \hat{w}$ in its $\hat{u}\hat{v}\hat{w}$ coordinate system. So $p_4(\mathbf{x}') = 1/(\pi r^2)$, and

$$\mathbf{x}' = \mathbf{c}' + \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \begin{bmatrix} r\sqrt{\xi_1}\cos(2\pi\xi_2) \\ r\sqrt{\xi_1}\sin(2\pi\xi_2) \\ 0 \end{bmatrix}$$

An image with a disk luminaire sampled using p_4 with one sample per pixel is shown in Figure 4.

3.3.3 Sampling Rectangular Luminaires

If a rectangle is defined with a point $\mathbf{x}_0$ and side vectors $\overrightarrow{v}_1$ and $\overrightarrow{v}_2$, then a uniform random point is given by

$$\mathbf{x}' = \mathbf{x}_0 + \xi_1 \overrightarrow{v}_1 + \xi_2 \overrightarrow{v}_2$$

and has density $p_5(\mathbf{x}') = 2/\|\overrightarrow{v}_1 \times \overrightarrow{v}_2\|$. If the luminaire is entirely above the horizon, then it is also possible to choose a random point with density $p_6(\mathbf{x}') \propto (\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ and $p_7(\mathbf{x}') \propto (-\hat{\omega}'\cdot\hat{n})(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$, but the evaluation of p_6 and p_7 are costly, and their implementation is difficult (see [44] for details). One sample per pixel images using p_5, p_6, and p_7 are shown in Figure 5. Note that the complex density functions performs better when the luminaire and $\mathbf{x}$ are close to each other, as should be expected because all the terms in $(-\hat{\omega}'\cdot\hat{n})(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ vary most in those regions.

3.3.4 Sampling Triangular Luminaires

To generate uniform random points on a triangle, we use the natural barycentric coordinates of the triangle, and note that the determinant of the Jacobian that takes a triangle defined with barycentric coordinates to a triangle with an unnormalized coordinate system is a constant. This means that if we pick random barycentric coordinates for one triangle, we can use them for any other triangle and the random point will still be uniform. We now go into some detail on the mechanics of choosing a uniform $\mathbf{x}'$, because when the uniform density function is written in terms of the two coordinates on the triangle, it is not separable. Using barycentric coordinates, every point, on the plane containing a triangle with vertices $\mathbf{p}_0$, $\mathbf{p}_1$, and $\mathbf{p}_2$ can be

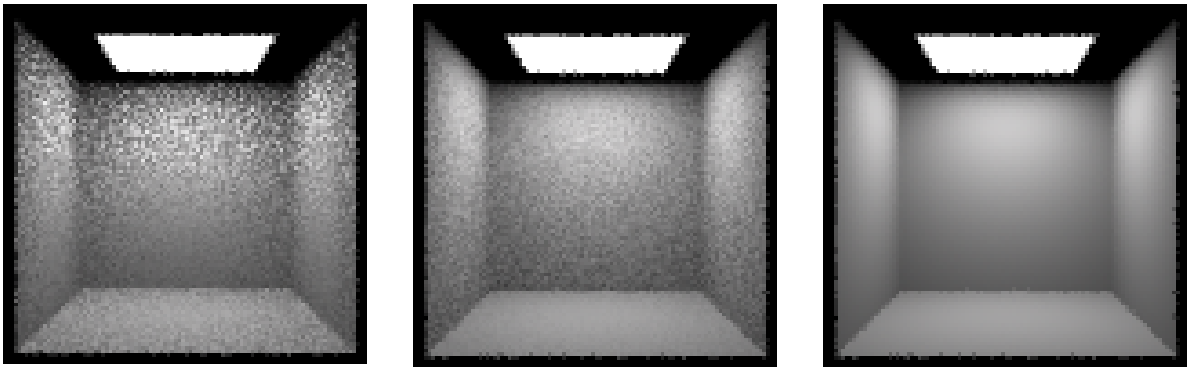

Figure 5: A rectangular luminaire sampled with p_5 (left) and p_6 (middle) and p_7 (right).

described by $\mathbf{x}' = \mathbf{p}_0 + \beta(\overrightarrow{\mathbf{p}_1 - \mathbf{p}_0}) + \gamma(\overrightarrow{\mathbf{p}_2 - \mathbf{p}_0})$[7] and the $\mathbf{x}'$is in the triangle if an only if $(\beta > 0)$ and $(\gamma > 0)$ and $(\beta + \gamma < 1)$. Integrating the constant one across the triangle gives

$$\int_{\gamma=0}^{1} \int_{\beta=0}^{1-\gamma} d\beta \, d\gamma = \frac{1}{2}$$

This means the value of our density function is 2 in barycentric coordinates (and $1/A$ in cartesian space, where A is the area of the triangle, implying the determinant of the Jacobian that transforms from barycentric to cartesian space has the value $2A$).

To perform an inversion we must first find densities which correspond to β and γ separately. If β and γ were independent we could simply use the marginal density for each. In this case, β is dependent on γ. Therefore, we will use the marginal density for γ, $f_G(\gamma)$, and the conditional density of β given γ, $f_{B|G}(\beta|\gamma)$. By definition

$$f_G(\gamma) = \int_0^{1-\gamma} f(\beta, \gamma) \, d\beta = 2(1-\gamma)$$

and

$$f_{B|G}(\beta|\gamma) = \frac{f(\beta,\gamma)}{f_G(\gamma)} = \frac{1}{1-\gamma}$$

Now we can perform the inversion by letting $\xi_1 = F_G(\gamma')$ and $\xi_2 = F_{B|G}(\beta'|\gamma')$ and solving for β' and γ'. Here F_G and $F_{B|G}$ are the cumulative densities corresponding to the random variables γ' and $\beta'|\gamma'$ respectively. Again by definition

$$\xi_1 = F_G(\gamma') = \int_0^{\gamma'} f_G(\gamma) \, d\gamma = 2\gamma' - {\gamma'}^2$$

Solving for γ' we get $\gamma' = 1 - \sqrt{1-\xi_1}$. For β'

$$\xi_2 = F_{B|G}(\beta'|\gamma') = \int_0^{\beta'} f_{B|G}(\beta'|\gamma') \, d\beta = \frac{\beta'}{1-\gamma'}$$

thus $\beta' = \xi_2(1-\gamma') = \xi_2\sqrt{1-\xi_1}$. Therefore

$$\mathbf{x}' = \mathbf{p}_0 + \xi_2\sqrt{1-\xi_1}\left(\overrightarrow{\mathbf{p}_1 - \mathbf{p}_0}\right) + 1 - \sqrt{1-\xi_1}\left(\overrightarrow{\mathbf{p}_2 - \mathbf{p}_0}\right) \tag{22}$$

A triangular luminaire sampled using this constant p_8, as well as using $p_9 \propto (-\hat{\omega}'\cdot\hat{n})(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^{-2}$ (as with spheres and rectangles, this only works if the entire luminaire is above the horizon) is shown in Figure 6. Details on using p_9 can be found in [44].

[7] Three barycentric coordinates are often used with $\mathbf{x}' = [\mathbf{o} + \alpha(\overrightarrow{\mathbf{p}_0 - \mathbf{o}}) + \beta(\overrightarrow{\mathbf{p}_1 - \mathbf{o}}) + \gamma(\overrightarrow{\mathbf{p}_2 - \mathbf{o}})]$, where $\mathbf{o}$ is the coordinate origin and $\alpha + \beta + \gamma = 1$. We just use the substitution $\alpha = (1 - \beta - \gamma)$.

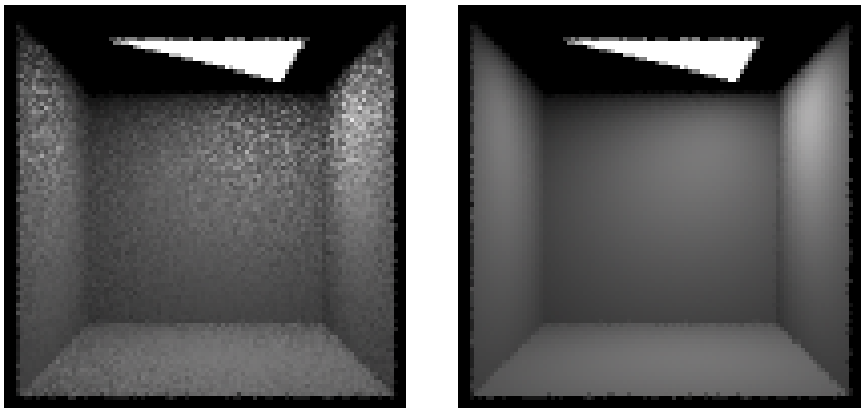

Figure 6: Triangular luminaire sampled using p_8 (left) and p_9 (right).

3.4 Sampling Cylindrical Luminaires

If we have a cylinder of radius r_c with its axis along the $\hat{w}$ axis, and boundaries $w = 0$ and $w = w_{\max}$, then we can choose a point in the $\hat{u}\hat{v}\hat{w}$ coordinate system that is uniform on the cylinder: $[u, v, w]^T = [r_c \cos(2\pi\xi_1), r_c \sin(2\pi\xi_1), \xi_2 w_{\max}]^T$. If the base origin of the $\hat{u}\hat{v}\hat{w}$ coordinate system is $\mathbf{o}$, then a random point in cartesian coordinates is:

$$\mathbf{x}' = \mathbf{o} + \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \begin{bmatrix} r_c \cos(2\pi\xi_1) \\ r_c \sin(2\pi\xi_1) \\ \xi_2 w_{\max} \end{bmatrix}$$

A more practical (lower variance) cylinder sampling strategy that approximates a density function proportional to $(\hat{\omega}' \cdot \hat{n}')$ is presented in [51].

3.5 Discussion

Since the density functions developed in this section assume that $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ and $L_e(\mathbf{x}', \hat{\omega}')$ are roughly constant in the directions subtended by the luminaire, a variance increase can be introduced by specular reflection or directional luminaires. If $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ or $L_e(\mathbf{x}', \hat{\omega}')$ vary more than other terms in $L_s(\mathbf{x}, \hat{\omega})$, then all the methods discussed in this section may have a high variance. For example, if we use a rectangle with a texture controlling emission to represent a TV screen, then the variance will be roughly proportional to the variance of the pixels in the texture. When $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ is the most important term, we want $p \propto \rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ as in [45, 21] and when $L_e(\mathbf{x}', \hat{\omega}')$ is the key factor, we want to use $p \propto L_e(\mathbf{x}', \hat{\omega}')$.

There are many luminaire shapes we have not discussed that might be of interest. For example, neon lights could be considered swept disks, and common light bulbs can be modeled as surfaces of revolution. However, in practice we can often use the simple shapes discussed in this section in preference to the complex shapes of the real luminaires. For example, Figure 7 shows a room lit by flourescent fixtures that are approximated by rectangles for the lighting calculations at other surfaces. This idea is used extensively in Ward's *Radiance* program [46], where the far-field photometry of the entire fixture is known (see [10] for details on getting CIE or IES standard far-field photometry for real luminaires) and the luminaire is displayed with a complex geometry. This can be thought of as a special case of geometric simplification as used by Rushmeier et al. [29].

4 Direct Lighting from Many Luminaires

Traditionally, when N_L luminaires are in a scene, the direct lighting integral is broken into N_L separate integrals [7]. This implies at least N_L samples must be taken to approximate the direct lighting, or some

Figure 7: Rectangular Luminaires are used as imposters

bias must be introduced (as done by Ward where small value samples are not calculated [45]). Instead we leave the direct lighting integral intact and design a probability density function over all N_L luminaires.

As an example, suppose we have two luminaires, ℓ_1 and ℓ_2, and we devise two probability functions $p_1(\mathbf{x}')$ and $p_2(\mathbf{x}')$, where $p_i(\mathbf{x}') = 0$ for $\mathbf{x}'$ not on ℓ_i and $p_i(\mathbf{x}')$ is found by a method such as one of those described Section 3 for generating $\mathbf{x}'$ on ℓ_i. These functions can be combined into a single *mixture density* [41] over both lights by applying a weighted average:

$$p(\mathbf{x}') = \alpha p_1(\mathbf{x}') + (1-\alpha)p_2(\mathbf{x}')$$

where $\alpha \in (0,1)$. We can see that p is a probability density function because its integral over the two luminaires is one, and it is strictly positive at all points on the luminaires. The coefficients α and $(1-\alpha)$ are called the *mixing weights*.

To estimate $L = (L_1 + L_2)$, where L is the direct lighting and L_i is the lighting from luminaire ℓ_i, we first choose a random canonical pair (ξ_1, ξ_2), and use it to decide which luminaire will be sampled. If $0 \leq \xi_1 < \alpha$, we estimate L_1 with $\hat{L}_1$ using the methods described in Section 3 to choose $\mathbf{x}'$ and to evaluate $p_1(\mathbf{x}')$, and we estimate L with $\hat{L}_1/\alpha$. If $\xi_1 \geq \alpha$ then we estimate L with $\hat{L}_2/(1-\alpha)$. In either case, once we decide which source to sample, we cannot use (ξ_1, ξ_2) directly because we have used some knowledge of ξ_1. So if we choose ℓ_1 (so $\xi_1 < \alpha$), then we choose a point on ℓ_1 using the random pair $(\xi_1/\alpha, \xi_2)$. If we sample ℓ_2 (so $\xi_1 \geq \alpha$), then we use the pair $((\xi_1 - \alpha)/(1-\alpha), \xi_2)$. This way a collection of stratified samples will remain stratified in some sense. Note that it is to our advantage to have ξ_1 stratified in one dimension, as well as having the pair (ξ_1, ξ_2) stratified in two dimensions, so that the ℓ_i we choose will be stratified over many (ξ_1, ξ_2) pairs, so some multijittered sampling method may be helpful (e.g [5]).

This basic idea used to estimate $L = (L_1 + L_2)$ can be extended to N_L luminaires by mixing N_L densities

$$p(\mathbf{x}') = \alpha_1 p_1(\mathbf{x}') + \alpha_2 p_2(\mathbf{x}') + \cdots + \alpha_{N_L} p_{N_L}(\mathbf{x}') \tag{23}$$

where the α_i's sum to one, and where each α_i is positive if ℓ_i contributes to the direct lighting. The value of α_i is the probability of selecting a point on the ℓ_i, and p_i is then used to determine which point on ℓ_i is chosen. If ℓ_i is chosen, the we estimate L with $\hat{L}_i/\alpha i$. Given a pair (ξ_1, ξ_2), we choose ℓ_i by enforcing the conditions

$$\sum_{j=1}^{i-1} \alpha_j \; < \xi_1 < \; \sum_{j=1}^{i} \alpha_j$$

And to sample the light we can use the pair (ξ_1', ξ_2) where

$$\xi_1' = \frac{\xi_1 - \sum_{j=1}^{i-1} \alpha_j}{\alpha_i}$$

This basic process is shown in Figure 8. It cannot be over stressed that it is important to "reuse" the random samples in this way to keep the variance low, in the same way we use stratified sampling (jittering) instead of random sampling in the space of the pixel[8]. To choose the point on the luminaire ℓ_i given (ξ_1', ξ_2), we can use the same types of p_i for luminaires as used in the last section. The question remaining is what to use for α_i.

4.1 Constant α_i

The simplest way to choose values for α_i was proposed by Lange [21] (and this method is also implied in the figure on page 148 of [16]), where all weights are made equal: $\alpha_i = 1/N_L$ for all i. This would definitely make a valid estimator because the α_i sum to one and none of them is zero. Unfortunately, in many scenes this estimate would produce a high variance (when the Li are very different as occurs in most night "walkthroughs").

[8] It is also important to not introduce any bais by correlated points in screen space and "luminaire space". Typically a set of N sample pairs is chosen in screen space, and a different set of N sample pairs is chosen for luminaire space, and these are combined into N quadruples in screen-luminaire space. If there is some regularity in how these points are generated (e.g. the first point is in a particular corner of canonical space), then the mapping between pairs should be random [6].

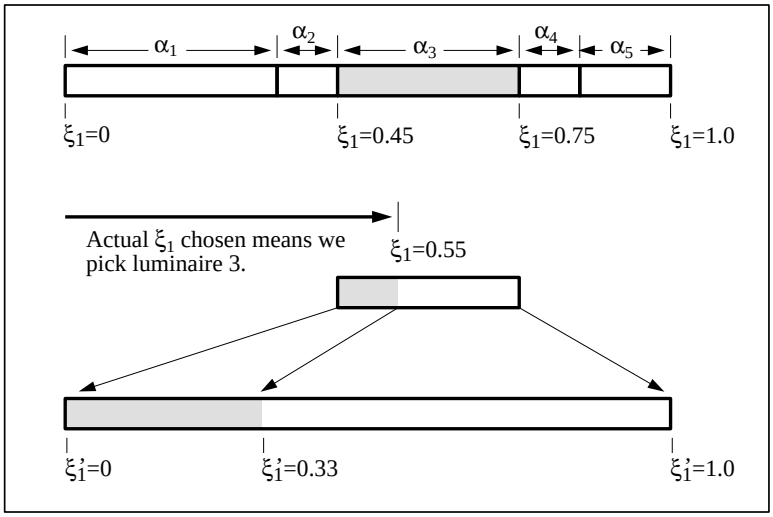

Figure 8: Diagram of mapping ξ_1 to choose ℓ_i and the resulting remapping to new canonical sample ξ_1'.

4.2 Linear α_i

Suppose we had perfect p_i defined for all the luminaires. A zero variance solution would then result if we could set $\alpha_i \propto L_i$, where L_i is the contribution from the ith luminaire. If we can make α_i approximately proportional to L_i, then we should have a fairly good estimator. We call this the *linear method* of setting α_i because the time used to choose one sample is linearly proportional to N_L, the number of luminaires.

To obtain such α_i we get an estimated contribution $\hat{L}_i$ at $\mathbf{x}$ by approximating Equation 13 for ℓ_i with the geometry term $g(\mathbf{x}, \mathbf{x}')$ set to one. These $\hat{L}_i$s (from all luminaires) can be directly converted to α_i by scaling them so their sum is one:

$$\alpha_i = \frac{\hat{L}_i}{\hat{L}_1 + \hat{L}_2 + \cdots + \hat{L}_{N_L}} \tag{24}$$

This method of choosing α_i will be valid because all potentially visible luminaires will end up with positive α_i. We should expect the highest variance in areas where shadowing occurs, because this is where setting the geometry term to one causes α_i to be a poor estimate of α_i.

Figure 10 shows a scene with four rectangular luminaires sampled using the linear method and 49 rays per pixel. Figure 9 shows a scene with 100 rectangular luminaires sampled with 49 rays per pixel using the linear method. This method of setting α_i was first used in [33], and was first theoretically justified in [36].

Implementing the linear α_i method has several subtleties. We implemented a method for each type of luminaire that estimated L_i for a particular $\mathbf{x}$ and $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$. If the entire luminaire is below the tangent plane at $\mathbf{x}$, then the estimate for $\hat{L}_i$ should be zero. An easy mistake to make (which we made in our initial implementation), is to set $\hat{L}_i$ to zero if the center of the luminaire is below the horizon. This will make α_i take the one value that is not allowed: an incorrect zero. Such a bug will become obvious in pictures of spheres illuminated by luminaires that subtend large solid angles, but for many scenes such errors are not noticeable (the figures in [33] had this bug, but it was not noticeable). To overcome this problem, we make sure that for a polygonal luminaires all of its vertices are below the horizon before it is given a zero probability of being sampled. For spherical luminaires, we check that the center of the luminaire is a distance greatew than the sphere radius under the horizon plane before it is given a zero probability of being sampled.

4.3 Spatial Subdivision

In the linear method, choosing α_i based on estimated contribution requires querying every luminaire in the scene. This is acceptable for many scenes, but if N_L is large (thousands or millions), even that might be too slow. In such scenes at most a few hundred luminaires (and usually at most tens) will contribute significantly

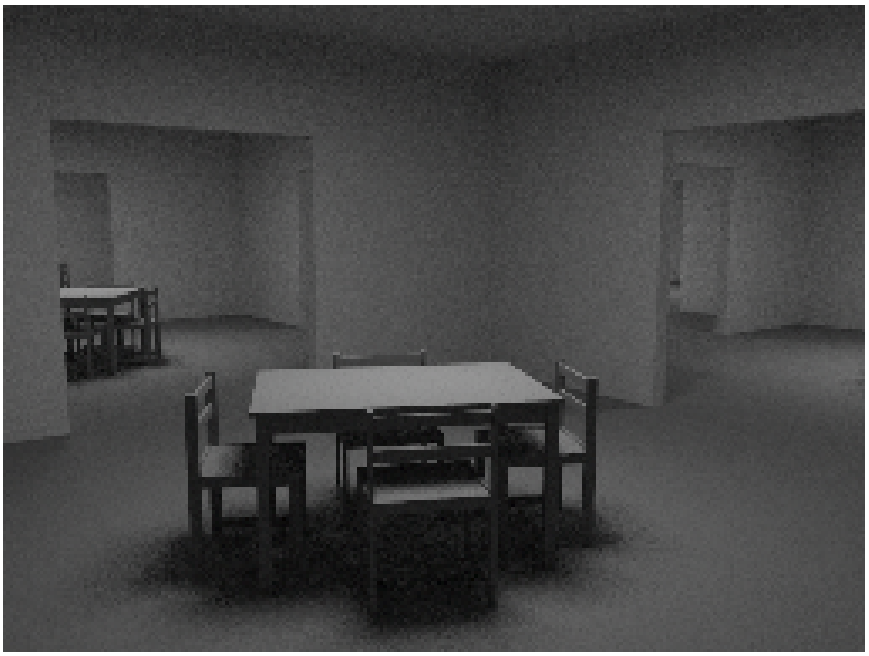

Figure 9: An image that used the "linear" method with 100 luminaires.

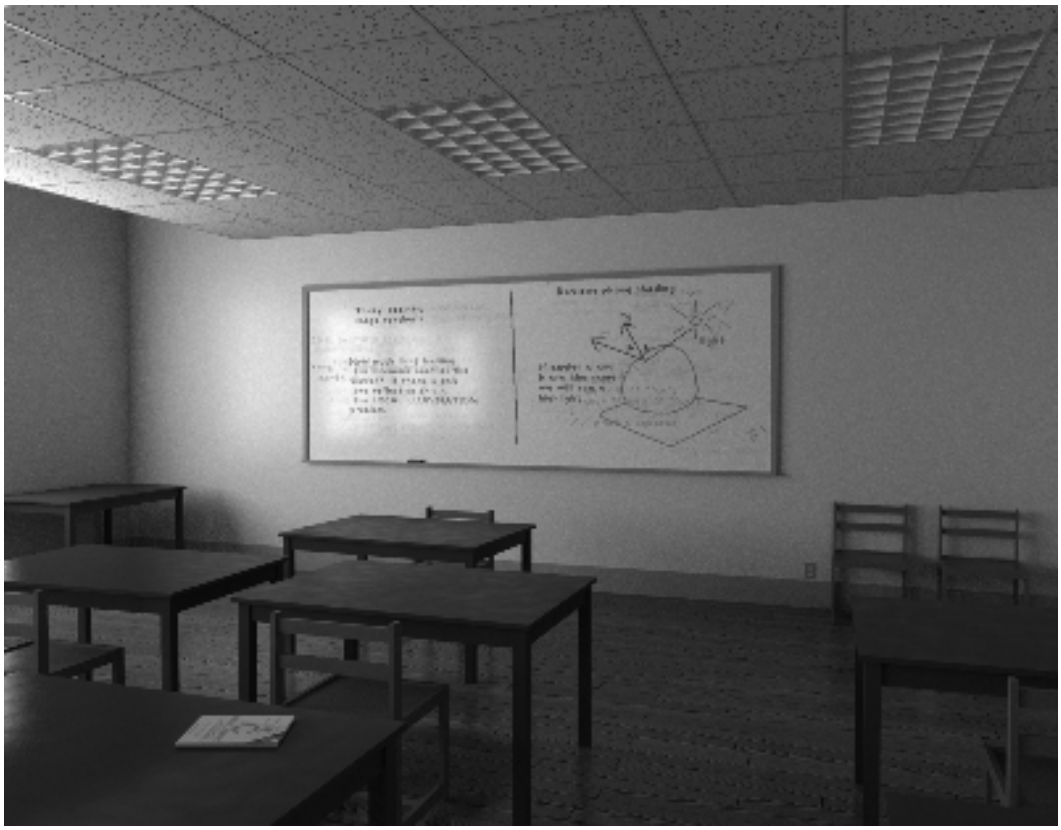

Figure 10: An image that used the "linear" method with four luminaires.

to the radiance at any particular point. Suppose we can partition $\mathcal{L} = \{\ell_1, \ldots, \ell_{N_L}\}$ into two subsets $\mathcal{L}_{\text{bright}}$ and $\mathcal{L}_{\text{dim}}$, where $\mathcal{L}_{\text{bright}}$ is the set of luminaires that are "important" at $\mathbf{x}$ (i.e. they contribute significantly to $L_s(\mathbf{x}, \hat{\omega})$ for some $\hat{\omega}$), and $(\mathcal{L}_{\text{dim}} = (\mathcal{L} - \mathcal{L}_{\text{bright}})$. With these two subsets of $\mathcal{L}$ we can construct low cost α_i.

Suppose that the size of $\mathcal{L}_{\text{bright}}$ is N_b and that $\mathcal{L}_{\text{bright}} = \{b_1, b_2, \ldots, b_{N_b}\}$, and that $\mathcal{L}_{\text{dim}} = \{d_1, d_2, \ldots, d_{N_d}\}$, where $N_d = N_L - N_b$. If N_L is large and we have partitioned $\mathcal{L}$ correctly, then N_b should be much less than N_d. For each b_i in $\mathcal{L}_{\text{bright}}$, we estimate L_i in the same way we did in the linear method. We now assume that all members of $\mathcal{L}_{\text{dim}}$ contribute approximately the same amount $\hat{L}_d$ as a random element d_r of $\mathcal{L}_{\text{dim}}$. To pick a random element in $\mathcal{L}_{\text{dim}}$ we use a rejection method where we repeatedly choose a luminaire at random from $\mathcal{L}$ (which we store in a linear array) until we find a light that is not in $\mathcal{L}_{\text{bright}}$ (that is faster than finding one that is in $\mathcal{L}_{\text{dim}}$ because we expect that $N_d >> N_b$). If we want a better estimate for the average contribution of a luminaire in $\mathcal{L}_{\text{dim}}$, we can take more than one sample. The appropriate number of such samples to take in $\mathcal{L}_{\text{dim}}$ is scene dependent, and is something that should be studied further. For our implementation we have taken one sample. This gives estimates $\hat{L}_i$ for contributions from all luminaires without consulting more than one (or a few if more than one sample is taken) element of $\mathcal{L}_{\text{dim}}$, and will be reasonably accurate for the important luminaires. We can now construct a probability density function by setting α_i equal to the normalized value of $\hat{L}_i$:

$$\alpha_i = \frac{\hat{L}_i}{\hat{L}_1 + \hat{L}_2 + \cdots + \hat{L}_{N_L}} = \frac{\hat{L}_i}{\sum_{\ell_j \in \mathcal{L}_{\text{bright}}} \hat{L}_j + N_d \hat{L}_r} \tag{25}$$

where $\hat{L}_r$ is our estimate for the contribution of the random luminaire from $\mathcal{L}_{\text{dim}}$.

The difficult part of this method is deciding which luminaires are in $\mathcal{L}_{\text{bright}}$ for a particular $\mathbf{x}$. As pointed out by Kok and Jansen [19], a luminaire that is responsible for a large fraction of the radiance of $\mathbf{x}$ is likely to be responsible for a large fraction of the radiance of the neighboring points of $\mathbf{x}$. This implies we can use a spatial subdivision scheme to precompute a $\mathcal{L}_{\text{bright}}$ list for each spatial cell in the spatial subdivision structure. For a particular cell a luminaire is put in $\mathcal{L}_{\text{bright}}$ if it might contribute more than a threshold average spectral radiance to a diffuse surface within the cell. A simple way to determine whether the maximum potential contribution of a luminaire is above the threshold is to evaluate Equation 13 with $g(\mathbf{x}, \mathbf{x}')$ and $(-\hat{\omega}' \cdot \hat{n})$ each set to one, and $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ set to $\rho_{\max}$ for all points (implementationally a large number of points) on the boundary of the spatial subdivision cell. Because specular surfaces are usually handled using reflection rays (and thus Equation 12. See [37] for details) rather than shadow rays, we can expect $\rho_{\max}$ to not be extremely large. In Figure 11, a set of luminaires and their regions of importance are shown. The stratification of the scene into boxes is arbitrary.

An easy way to choose the subdivision cells is to use the leaf cells of the conventional subdivision structure (e.g. the octree leaves of a Glassner style octree used for ray intersection acceleration[9]), and maintain a separate $\mathcal{L}_{\text{bright}}$ list at each leaf. This is a finer than needed subdivision for scenes where the number of objects is much greater than the number of luminaires. An advantage of this is that no luminaire lists need to be constructed for empty cells, and the characteristics of reflective objects can be used to construct the lists (because the objects in each cell are known in advance). Instead, we have implemented a separate *light octree* that recursively subdivides itself until each leaf is at a maximum allowed depth or when the size of $\mathcal{L}_{\text{bright}}$ for that cell is below a specified limit. The depth and size limits are similar to those in a conventional octree, and their values are even less well-understood by us so far. To avoid excess subdivision, we check whether the minimum contribution of an important luminaire to a cell is above the threshold. If all members of $\mathcal{L}_{\text{bright}}$ are thus determined to be in $\mathcal{L}_{\text{bright}}$ for any possible descendant of that cell, we do not subdivide.

Another difficulty in building the light octree is what to use for the average radiance threshold. For our program we assume we know what radiance L_w would map to white on the display device (white would be an rgb triple of [255,255,255] on a typical 24 bit frame-buffer driven CRT). We set the radiance threshold T to be some fraction of L_w[9]. Because our display device typically has eight bits per channel, we usually make the threshold a few percent of L_w. Such a threshold will ensure that any luminaire that can change

[9] L_w should be chosen according to a perceptual viewer model, such as the model implemented by Tumblin and Rushmeier [42]. L_w may also vary from pixel to pixel if spatially varying mappings are used [4]. Such models will become increasingly important as physically based rendering becomes more popular.

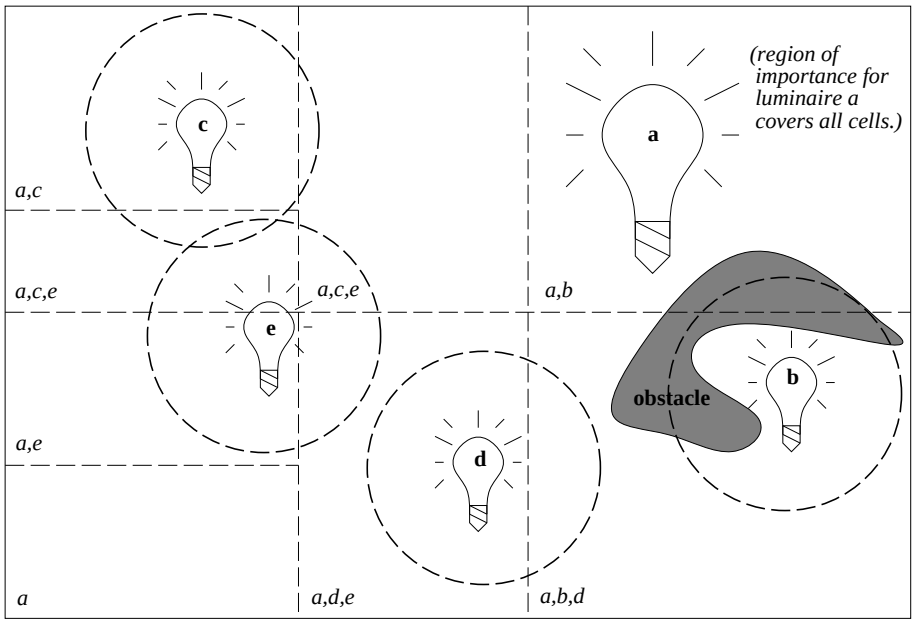

Figure 11: A subdivision of space where luminaire is a member of $\mathcal{L}_{\text{bright}}$ for a cell if it has the potential to contribute significantly to the surface radiance of a point in that cell.

the pixel color more than a few steps will be included in $\mathcal{L}_{\text{bright}}$. If L_w is chosen correctly, then the sum of contributions of luminaires should be no more than L_w, and thus the number of important luminaires should be less than one hundred for most scenes.

To characterize important vs unimportant luminaires, we associate an axis aligned *influence box* with each luminaire that includes all points that might include that luminaire in its $\mathcal{L}_{\text{bright}}$ list. When deciding whether a luminaire is important to a cell, we just check whether the cell and the influence box overlap, and if so, then the luminaire is treated as an important light source. To avoid an important luminaire being overlooked, the influence box must contain every point whose reflected radiance might be larger than a predefined threshold T when receiving the light only from that luminaire. Although this gives the chance of a non-important luminaire being selected as an important one, that is preferred to risk missing an important luminaire because our estimate is $\hat{L}_i/\alpha_i$ which will blow up if $\hat{L}_i$ is large and α_i is small, but will just change a small component if $\hat{L}_i$ is small, regardless of α_i.

To decide where a luminaire might cause an object to change its radiance more than T, we first examine a point source at point $\mathbf{x}'$ with radiant intensity[10] distribution $I(\hat{\omega}')$. The potential contribution due to this source can be bounded by replacing $\rho(\mathbf{x}, \hat{\omega}, \hat{\omega}')$ with $\rho_{\max}$, and $(-\hat{\omega}' \cdot \hat{n})$ with one:

$$L_s(\mathbf{x}, \hat{\omega}) \leq \frac{\rho_{\max} I(\hat{\omega}')}{\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\|^2} \tag{26}$$

To make sure this potential contribution is below T, we make sure the distance is above a certain quantity:

$$\|\overrightarrow{\mathbf{x} - \mathbf{x}'}\| \geq \sqrt{\frac{\rho_{\max} I(\hat{\omega}')}{T}} \tag{27}$$

Note that the set of points $\mathbf{x}$ that have a potential above T are the inside of an *influence isosurface* that depends only on $I(\hat{\omega}')$ (see Figure 12). The box that bounds this influence isosurface is that luminaire's influence box.

[10] The radiant intensity, $I(\hat{\omega}')$, is the power per unit solid angle in direction $\hat{\omega}'$.

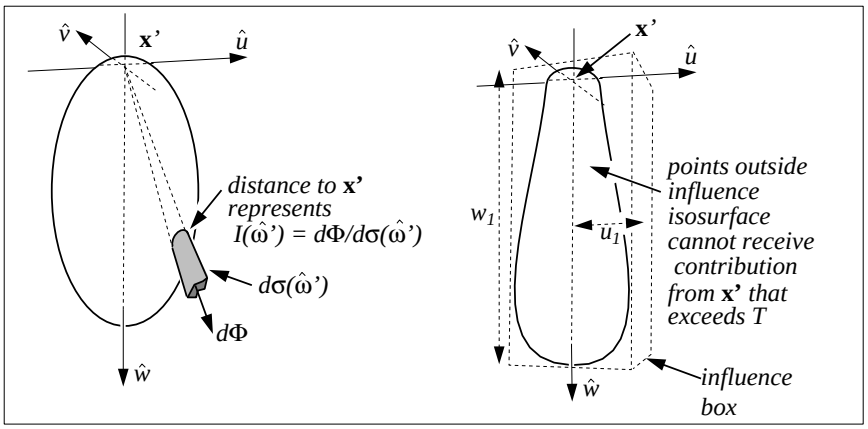

Figure 12: The goniometric diagram of radiant intensity of a point source (left) and the associated influence isosurface where the maximum radiance contribution is T.

As an example of finding an influence isosurface and the resulting influence box of a finite sized luminaire, we now consider a spherical luminaire with a directionally isotropic emission pattern and emitted power Φ, and a resulting far-field radiant intensity of $I(\hat{\omega}') = \Phi/(4\pi)$. Allowing for a finite radius r just makes the radius r_i of the sphere's influence isosurface expand by r:

$$r_i = \sqrt{\frac{\rho_{\max}\Phi}{4\pi T}} + r \tag{28}$$

If $I(\hat{\omega}')$ of the luminaire has $\hat{w}$ as an axis of symmetry, then $I(\hat{\omega}')$ will depend only on $(\hat{w} \cdot \hat{\omega}')$. If we define ϑ to be the angle between $\hat{w}$ and $\hat{\omega}'$ then we have a function $I(\vartheta)$ that fully describes the emission of the luminaire. To find the influence box which contains points $[u_0, u_1] \times [v_0, v_1] \times [w_0, w_1]$, we first note that because of symmetry, $u_0 = v_0 = -u_1 = -v_1$. For simplicity, we assume that $I(\vartheta) = 0$ for $\vartheta \geq \pi/2$, so $w_0 = 0$ (when this is not true, the analysis for $\vartheta \geq \pi/2$ is analogous to the analysis for $\vartheta < \pi/2$). To find w_1 we note that on the potential isosurface at angle ϑ, the value of w can be found from Equation 27:

$$w = \cos\vartheta \sqrt{\frac{\rho_{\max} I(\hat{\omega}')}{T}}$$

To find w_1 we can differentiate w^2 (which will be maximum when w is maximum) with respect to ϑ and find where this derivative is zero. This occurs when:

$$-2\sin\vartheta I(\vartheta) + \cos\vartheta \frac{dI(\vartheta)}{d\vartheta} = 0 \tag{29}$$

Similar analysis for u_1 yields that The largest u inside the influence isosurface occurs at ϑ such that

$$2\cos\vartheta I(\vartheta) + \sin\vartheta \frac{dI(\vartheta)}{d\vartheta} = 0 \tag{30}$$

Equations 29 and 30 can help us find the influence box of a point "phong"' luminaire with power Φ located at $\mathbf{x}'$ with

$$I(\vartheta) = \frac{(e+1)\Phi\cos^e\vartheta}{2\pi}$$

Differentiating $I(\vartheta)$ yields

$$\frac{dI(\vartheta)}{d\vartheta} = -\frac{e(e+1)\Phi\cos^{e-1}\vartheta\sin\vartheta}{2\pi}$$

This implies that w on the potential isosurface will be a maximum when $\vartheta = 0$, so

$$w_1 = \sqrt{\frac{\rho_{\max}(e+1)\Phi}{2\pi T}}$$

Applying Equation 30 implies that u will be maximum on the potential isosurface when $(2\cos^2\vartheta = e\sin^2\vartheta)$, so

$$u_1 = \left(\frac{e}{e+2}\right)^{\frac{e}{2}} \sqrt{\frac{\rho_{\max}\Phi}{\pi(e+2)T}}$$

This will give us an influence box about a point phong source in the $\hat{u}\hat{v}\hat{w}$ coordinate system. To extend ths result to polygonal luminaires, we can apply the point source analysis to a given vertex, and the add to u_1 the maximum u distance to any other vertex. To find the importance box in cartesian coordinates, we apply the conservative strategy of finding the cartesian axis aligned bounding box of the $\hat{u}\hat{v}\hat{w}$ influence box.

Figure 13 show two images of a pair of "sphereflakes" (generated using Eric Haines' procedural database software), with each sphereflake is composed of 7381 spheres. In the image with 7381 luminaires, the light octree was used. In both cases, individual luminaires had were sampled with a $(\hat{\omega}'\cdot\hat{n}')\|\overrightarrow{\mathbf{x}-\mathbf{x}'}\|^{-2}$ density function. The single luminaire case used ten samples per pixel, and the 7381 luminaire case used forty samples per pixel, and took about eight times as long as the ten sample single luminaire image. This means the overhead of choosing the ray using the light octree does not swamp the cost of sending the shadow ray, even in this extreme case where half of the objects are luminaires. In less pathological scenes, the light octree should be smaller relative to the number of primitives, and performance should be even better.

Clearly, the linear and light octree methods are most useful for scenes with a large number of luminaires. Such scenes are becoming increasingly important. In outdoor scenes, especially in urban settings, scenes with thousands and even hundreds of thousands of luminaires are commonplace. In opera and theater applications, hundreds or thousands of luminaires are common [8]. In infrared scenes, almost all surfaces are luminaires, so something such as the light octree is crucial. These methods also make it easy to use luminaires defined by many polygons or parametric patches. No matter how many patches define a light bulb surface, it will receive only one shadow ray.

5 Conclusion

We have presented techniques for constructing probability spaces on luminaire surfaces for the direct lighting calculation. These techniques make the calculation of direct lighting from thousands of luminaires feasible. The chief limitation of the work is that it does not take into account the geometry term (visibility) or the BRDF. Extending the methods to include these terms is possible, but it will not be easy. The geometry term in particular should probably be attacked using probabilistic methods, because the visibility problem in high complexity scenes is very difficult, and exact solutions will take too long.

The basic rationale for the techniques presented in this paper is that direct lighting should not be calculated to a higher accuracy than necessary. This is very similar in concept to Kajiya's argument that we should not expend much work for deep parts of the ray tree [16]. It is certainly not always true that one shadow ray per viewing ray is optimal, however. For the 100 luminaire case, one shadow ray is better than 100 shadow rays, but two or three might be better still. This issue requires further investigation.

Even in scenes with only a few luminaires, the linear or light octree techniques presented in Section 4 can be useful. Many of the recent rendering techniques can be viewed in terms of replacing surfaces with *imposters*. An example of an imposter is a luminaire with zero reflectivity that is shaped like a reflecting patch and *emits* light in roughly the same distribution and intensity as the reflecting patch *reflects* light. This concept has been used by Kok and Jansen[19], Chen et al.[3], Rushmeier et al. [29] and Ward [46]. Adding an imposter reduces the number of patches whose reflected light needs to be calculated and increases the number of luminaires. Using imposters becomes more attractive if the cost of direct lighting does not increase linearly with N_L, so we believe the techniques in this paper will be used with both radiosity and ray tracing programs.

Acknowledgements

The authors wish to thank Jim Arvo, Randy Bramley, Andrew Glassner, Holly Rushmeier, Ken Torrance, Bruce Walter and Greg Ward for their help on algorithms and terminology, Ken Chiu for help on the code

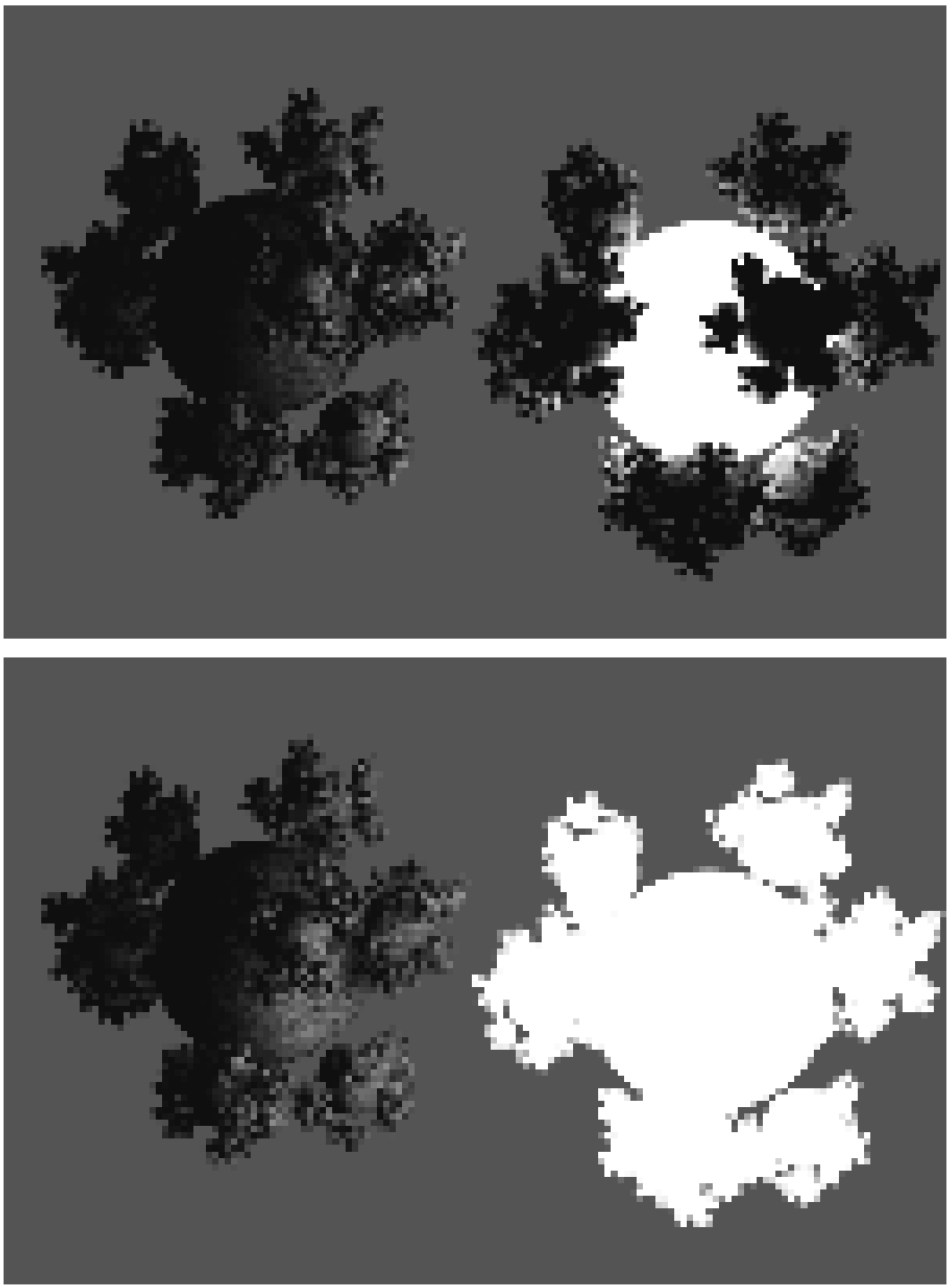

Figure 13: Single luminaire, 10 rays per pixel (top), 7381 luminaires, 40 rays per pixel (bottom).

involved in the project, Bill Kubitz, Dennis Gannon, Bob Skeel, Don Hearn for initial suggestions directions the work should be pushed. We also are grateful to the reviewers helped greatly improve a particularly difficult-to-read draft version of the paper.

This work was supported by Indiana University Faculty Start-up Funds, and the National Science Foundation under Grant No. NSF-CCR-92-09457.

Appendix: Proof of Optimal Sampling for Weighted Integrands

In image synthesis often have integrands that take the form of a product of a strictly nonnegative function $w : \mathcal{S} \to \mathbb{R}^+$ and a strictly nonnegative function $f : \mathcal{S} \to \mathbb{R}^+$ whose values are only attainable through point sampling:

$$I = \int_{\mathcal{S}} w(\psi) f(\psi) d\mu(\psi)$$

To solve this by Equation 2, the optimal choice for the probability function is $\wp(\psi) \propto w(\psi)f(\psi)$, but as is often pointed out, this choice requires us to already know the value of I. Instead, people often either choose uniform $\wp$, or set $\wp(\psi) = w(\psi)$[6, 26, 21]. This section extends arguments in [37] that letting $\wp(\psi) = w(\psi)$ is in some sense optimal.

In graphics we usually repeatedly perform an integral for every member of a set $F = \{f_1, f_2, \ldots, f_n\}$. To decide which $\wp$ to use, we could try to minimize the average variance of our estimate across all $f \in F$. We cannot do this with such a vague definition of F, but we can approximate our situation in graphics by assuming that we are no more likely to have large values of f in a particular point in S than any other. This makes sense when thinking of $\mathcal{S}$ being the support of a filter w on a pixel on the image plane, and f as being the radiance hitting the film plane; we do not expect the exact location of features in the luminance function to be correlated with the details of how the pixels are arranged. This is especially true if we will use the same w for many different images in an animation.

The average variance $\overline{V}$ of the estimator $I' = w(\psi)f(\psi)/\wp(\psi)$ is:

$$\overline{V} = \frac{1}{n} \sum_{i=1}^{n} \left[\int_{\mathcal{S}} \frac{w^2(\psi) f_i^2(\psi)}{\wp(\psi)} d\mu(\psi) - \left(\int_{\mathcal{S}} w(\psi) f - i(\psi) d\mu(\psi) \right)^2 \right] \tag{31}$$

Because the righthand (squared) term is $\sum I_i$ (a constant), and because we assume that average values of members of F are not correlated with the evaluation point in $\mathcal{S}$, then as n becomes large we should expect the average value of $f^2(\psi)$ at any given ψ is some constant $\overline{f^2}$. The average variance of the estimator I' is then just:

$$\overline{V} = \frac{1}{n} \sum_{i=1}^{n} \left[\int_{\mathcal{S}} \frac{w^2(\psi)\overline{f^2}}{\wp(\psi)} d\mu(\psi) - I^2 \right]$$

Because the I^2 term is a constant, the sum and the $1/n$ term will cancel each other (there are no more f_i terms, minimizing $\overline{V}$ is equivalent to finding $\wp$ that minimizes

$$I_v = \int_{\mathcal{S}} \frac{w^2(\psi)}{\wp(\psi)} d\mu(\psi) \tag{32}$$

with the constraint that $\wp$ is strictly nonnegative, and $\mu(p) = 1$.

To find a constrained function that minimizes a quantity, we use the calculus of variations with Lagrange multipliers. We first replace $\wp(\psi)$ with $\wp(\psi) + \alpha\gamma(\psi)$, where α is a constant and $\gamma(\psi)$ is a pertubation function. Because $\wp(\psi) + \alpha\gamma(\psi)$ must be a valid probability density function, we restrict γ to have zero volume: $\mu(\gamma) = 0$. We now define a new integral $I_c(\alpha)$ to be

$$I_c(\alpha) = \int_{\mathcal{S}} \left[\frac{w^2(\psi)}{\wp(\psi) + \alpha\gamma(\psi)} + \lambda(\wp(\psi) + \alpha\gamma(\psi)) \right] d\mu(\psi) \tag{33}$$

where λ is the Lagrange multiplier (a constant). The calculus of variations tells us that I_v will be stationary (an extremum) when the partial derivative of I_c evaluated at zero is zero. This is:

$$\frac{\partial I_c(\alpha)}{\partial \alpha} = \int_{\mathcal{S}} \left[\frac{-\gamma(\psi) w^2(\psi)}{(\wp(\psi) + \alpha\gamma(\psi))^2} + \lambda\gamma(\psi) \right] d\mu(\psi) \tag{34}$$

Setting $\partial I_c(0)/\partial\alpha = 0$ yields

$$\frac{\partial I_c(0)}{\partial \alpha} = \int_{\mathcal{S}} \left[\frac{\gamma(\psi) w^2(\psi)}{\wp^2(\psi)} - \lambda\gamma(\psi) \right] d\mu(\psi) = 0 \tag{35}$$

Because the integral must vanish for any valid $\gamma(\psi)$, we get

$$w^2(\psi) = \lambda \wp^2(\psi) \tag{36}$$

Since both w and $\wp$ are nonnegative, this implies that the intuitively appealing choice of $\wp = w$ has some theoretical justification as well. the second derivative $\partial^2 I_c(0)/\partial\alpha^2$ is always positive, so setting $\wp = w$ will ensure a minimum variance.

References

[1] John M. Airey and Ming Ouh-young. Two adaptive techniques let progressive radiosity outperform the traditional radiosity algorithm. Technical Report TR89-20, University of North Carolina at Chapel Hill, August 1989.

[2] James Arvo, Kenneth Torrance, and Brian Smits. A framework for the analysis of error in global illumination algorithms. *Computer Graphics*, 28(3), July 1994. ACM Siggraph '94 Conference Proceedings.

[3] Shenchang Eric Chen, Holly Rushmeier, Gavin Miller, and Douglass Turner. A progressive multi-pass method for global illumination. *Computer Graphics*, 25(4):165–174, July 1991. ACM Siggraph '91 Conference Proceedings.

[4] K. Chiu, M. Herf, P. Shirley, S. Swamy, C. Wang, and K. Zimmerman. Spatially nonuniform scaling functions for high contrast images. In *Graphics Interface '93*, pages 245–244, May 1993.

[5] Kennth Chiu, Peter Shirley, and Changyaw Wang. Multi-jittered sampling. In Paul Heckbert, editor, *Graphics Gems 4*. Academic Press, New York, NY, 1993.

[6] Robert L. Cook. Stochastic sampling in computer graphics. *ACM Transactions on Graphics*, 5(1):51–72, January 1986.

[7] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. *Computer Graphics*, 18(4):165–174, July 1984. ACM Siggraph '84 Conference Proceedings.

[8] Julie O'B. Dorsey, François X. Sillion, and Donald P. Greenberg. Design and simulation of opera lighting and projection effects. *Computer Graphics*, 25(4):41–50, July 1991. ACM Siggraph '91 Conference Proceedings.

[9] Andrew S. Glassner. Space subdivision for fast ray tracing. *IEEE Computer Graphics and Applications*, 4(10):15–22, 1984.

[10] Andrew S. Glassner. *Principles of Digital Image Synthesis*. Morgan-Kaufman, San Francisco, 1994.

[11] Eric Haines. Essential ray tracing algorithms. In Andrew S. Glassner, editor, *An Introduction to Ray Tracing*, pages 33–77. Academic Press, San Diego, CA, 1989.

[12] John H. Halton. A retrospective and prospective of the monte carlo method. *SIAM Review*, 12(1):1–63, January 1970.

[13] J. M. Hammersley and D. C. Handscomb. *Monte Carlo Methods*. Wiley, New York, N.Y., 1964.

[14] David S. Immel, Michael F. Cohen, and Donald P. Greenberg. A radiosity method for non-diffuse environments. *Computer Graphics*, 20(4):133–142, August 1986. ACM Siggraph '86 Conference Proceedings.

[15] American National Standard Institute. Nomenclature and definitions for illumination engineering. ANSI Report, 1986. ANSI/IES RP-16-1986.

[16] James T. Kajiya. The rendering equation. *Computer Graphics*, 20(4):143–150, August 1986. ACM Siggraph '86 Conference Proceedings.

[17] Malvin H. Kalos and Paula A. Whitlock. *Monte Carlo Methods*. John Wiley and Sons, New York, N.Y., 1986.

[18] David Kirk and James Arvo. Unbiased variance reduction for global illumination. In *Proceedings of the Second Eurographics Workshop on Rendering*, 1991.

[19] A. Kok and F. Jansen. Source selection for the direct lighting calculation in global illumination. In *Proceedings of the Second Eurographics Workshop on Rendering*, 1991.

[20] Arjan J. F. Kok and Frederik W. Jasen. Adaptive sampling of area light sources in ray tracing including diffuse interreflection. *Computer Graphics forum*, 11(3):289–298, 1992. Eurographics '92.

[21] Brigitta Lange. The simulation of radiant light transfer with stochastic ray-tracing. In *Proceedings of the Second Eurographics Workshop on Rendering*, 1991.

[22] Mark E. Lee, Richard A. Redner, and Samuel P. Uselton. Statistically optimized sampling for distributed ray tracing. *Computer Graphics*, 19(3):61–68, July 1985. ACM Siggraph '85 Conference Proceedings.

[23] Thomas J. V. Malley. A shading method for computer generated images. Master's thesis, University of Utah, June 1988.

[24] Don P. Mitchell. Spectrally optimal sampling for distribution ray tracing. *Computer Graphics*, 25(4), July 1991. Siggraph '91 Conference Proceedings.

[25] S. N. Pattanaik. *Computational Methods for Global Illumination and Visualization of Complex 3D Environments*. PhD thesis, Birla Institute of Technology & Science, February 1993.

[26] Werner Purgathofer. A statistical method for adaptive stochastic sampling. *Computers & Graphics*, 11(2):157–162, 1987.

[27] Mark S. Rea, editor. *The Illumination Engineering Society Lighting Handbook*. Illumination Engineering Society, New York, NY, 8th edition, 1993.

[28] Jon Rokne. The area of a simple polygon. In James Arvo, editor, *Graphics Gems II*, pages 5–6. Academic Press, San Diego, CA, 1991.

[29] Holly Rushmeier, Charles Patterson, and Aravindan Veerasamy. Geometric simplification for indirect illumination calculations. In *Graphics Interface '93*, pages 227–236, May 1993.

[30] Holly E. Rushmeier. *Realistic Image Synthesis for Scenes with Radiatively Participating Media*. PhD thesis, Cornell University, May 1988.

[31] Christophe Schlick. The acne problem. *Ray Tracing News*, 4(1), March 1991. e-mail Edition, available under anonymous ftp from weedeater.math.yale.edu.

[32] Peter Shirley. *Physically Based Lighting Calculations for Computer Graphics*. PhD thesis, University of Illinois at Urbana-Champaign, November 1990.

[33] Peter Shirley. A ray tracing algorithm for global illumination. In *Graphics Interface '90*, pages 205–212, May 1990.

[34] Peter Shirley. Discrepancy as a quality measure for sampling distributions. In *Eurographics '91*, pages 183–193, September 1991.

[35] Peter Shirley, Kelvin Sung, and William Brown. A ray tracing framework for global illumination. In *Graphics Interface '91*, pages 117–128, June 1991.

[36] Peter Shirley and Changyaw Wang. Direct lighting by monte carlo integration. In *Proceedings of the Second Eurographics Workshop on Rendering*, 1991.

[37] Peter Shirley and Changyaw Wang. Distribution ray tracing: Theory and practice. In *Proceedings of the Third Eurographics Workshop on Rendering*, pages 200–209, 1992.

[38] Y. A. Shreider. *The Monte Carlo Method*. Pergamon Press, New York, N.Y., 1966.

[39] François X. Sillion and Claude Puech. A general two-pass method integrating specular and diffuse reflection. *Computer Graphics*, 23(3):335–344, July 1989. ACM Siggraph '89 Conference Proceedings.

[40] Jerome Spanier and Earl H. Maize. Quasi-random methods for estimating integrals using relatively small samples. *SIAM Review*, 36(1):18–44, March 1994.

[41] D. M. Titterington, A. F. M. Smith, and U. E. Makov. *The Statistical Analysis of Finite Mixture Distributions*. John Wiley & Sons, New York, NY, 1985.

[42] Jack Tumblin and Holly Rushmeier. Tone reproduction for realistic computer generated images. *IEEE Computer Graphics and Applications*, 13(7), 1993.

[43] Changyaw Wang. Physically correct direct lighting for distribution ray tracing. In David Kirk, editor, *Graphics Gems 3*. Academic Press, New York, NY, 1992.

[44] Changyaw Wang. *The Direct Lighting calculation in Global Illumination Methods*. PhD thesis, Indiana University, November 1993.

[45] Greg Ward. Adaptive shadow testing for ray tracing. In *Proceedings of the Second Eurographics Workshop on Rendering*, 1991.

[46] Gregory J. Ward. The radiance lighting simulation and rendering system. *Computer Graphics*, 28(2), July 1994. ACM Siggraph '94 Conference Proceedings.

[47] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. *Computer Graphics*, 22(4):85–92, August 1988. ACM Siggraph '88 Conference Proceedings.

[48] H. Wozniakowski. Average case complexity of multivariate integration. *Bulliten (New Series) of the American Mathematical Society*, 24(1):185–193, January 1991.

[49] Sidney J. Yakowitz. *Computational Probability and Simulation*. Addison-Wesley, New York, N.Y., 1977.

[50] S. K. Zeremba. The mathematical basis of monte carlo and quasi-monte carlo methods. *SIAM Review*, 10(3):303–314, July 1968.

[51] Kurt Zimmerman. Direct lighting models for ray tracing with cylindrical lamps. In *Graphics Gems V*. Academic Press, Inc., 1995.