Making Shaders More Physically Plausible

Robert R. Lewis
bobl@cs.ubc.ca

University of British Columbia
Department of Computer Science

4 March, 1993

Abstract

There is a need to develop shaders that not only "look good", but are more physically plausible. From physical and geometric considerations, we review the derivation of a shading equation expressing reflected radiance in terms of incident radiance and the bidirectional reflectance distribution function (BRDF). We then examine the connection between this equation and conventional shaders used in computer graphics. Imposing the additional physical constraints of energy conservation and Helmholtz reciprocity allows us to create variations of the conventional shaders that are more physically plausible.

1 Introduction

Shading computation is an essential part of any rendering algorithm. Getting an exact physical model of the interaction of light with a surface is, for most surfaces occurring in the real world, a very difficult problem. Consequently, much effort has been expended on finding approximations that are both good-looking and quickly computed. An extensive summary of these shaders is in [hall89].

Looking good and being quickly computable are sufficient criteria for most raytracing renderers. If one of these shaders is used in a radiosity computation, however, it is necessary to additionally ensure that, since radiosity is based on principles of energy conservation (see [kaji86], for example), the shader itself conserves energy.

Another recent technique described in [ward92] constructs shading functions to fit actual bidirectional reflectance distribution function (hereafter, BRDF). Such fits are likely to be more successful if the shading functions themselves are, like the data they fit, consistent with physics.

So a need has arisen for shaders that are not only good-looking and easy to compute, but are also "physically plausible". [1]

In this paper, we'll review the development of the physical shading equation. Then, we'll look at the correspondence between that equation and traditional shaders. Finally, we'll look at ways to modify traditional shaders to make them more physically plausible.

2 Physically-Based Shading

In this section, we'll review the formulation of the fundamental equation describing physically-based shading, presented below as (3). Our presentation follows that given in [cook82] with appropriate syntactic substitutions to bring it in line with [ans86][2].

[1] We use the term *plausible* here in contrast to that of *feasible* in [neum89]. A "feasible" shader is one that we can imagine constructing physically. This is not always possible. The weaker definition of a "plausible" shader is one whose existence does not violate physics.

[2] We use the nomenclature defined in [ans86] throughout, except that we will omit the use of the term "spectral" and the corresponding "λ" subscripts, as all of our considerations will be monochromatic.

symbol	definition
α	angle between $\mathbf{N}$ and $\mathbf{H}$
b_*	scaling constants for various shaders
β	specular "half-angle"
dA	element of surface area
$d\omega_i$	$\sin\theta_i \, d\theta_i \, d\phi_i$
$d\omega_r$	$\sin\theta_r \, d\theta_r \, d\phi_r$
D_*	facet slope distribution functions
E	irradiance
F	Fresnel factor
f_r	bidirectional reflectance distribution function (BRDF)
F_s^*	specular shading functions
ϕ_i	incident azimuthal angle
ϕ_r	reflected azimuthal angle
Φ_L	angle between incident and viewing directions
G	geometrical attenuation factor
$\mathbf{H}$	bisector of source and viewing directions
k_a	ambient reflectance coefficient
k_d	diffuse reflectance coefficient
k_s	specular reflectance coefficient
k_ρ^*	directional-hemispherical reflectances
k_σ	fraction of total energy reflected specularly
L_a	ambient radiance
L_d	directional source radiance
L_i	incident radiance
L_r	reflected radiance
M	exitance
$\Omega_\mathbf{N}$	the hemisphere surrounding $\mathbf{N}$
m	RMS slope of the surface
$\mathbf{N}$	the surface normal
n_s^*	specular exponents
$\mathbf{R}_l$	reflected incident direction
$\mathbf{S}$	incident direction
θ_i	incident polar angle
θ_r	reflected polar angle
$\mathbf{V}$	direction of viewer

Table 1: Table of Commonly-Used Symbols

Figure 1 shows the (usual) geometry of an infinitesimal element of opaque, nonemissive surface area dA being illuminated by an incident radiance L_i coming from an infinitesimal solid angle $d\omega_i$ surrounding direction $\mathbf{S}$. (All vectors presented here are are of unit magnitude.) $\mathbf{N}$ is the surface normal at dA. An observer (or measuring device) is located in direction $\mathbf{V}$.

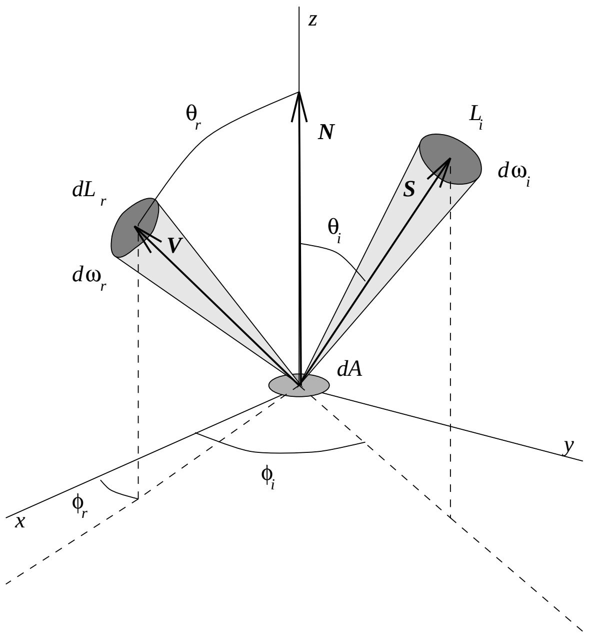

Figure 1: Shading geometry

We assume all reflected radiation passes unobstructed into $\Omega_\mathbf{N}$, the hemisphere surrounding $\mathbf{N}$. (What follows could be extended to account for transmission by taking $\Omega_\mathbf{N}$ to be the entire sphere centered on dA.)

We use $\mathbf{N}$ as the z-axis of a polar coordinate system so that we can specify $\mathbf{S}$ by the usual incident polar and azimuthal angles θ_i and ϕ_i and $\mathbf{V}$ by the usual reflected polar and azimuthal angles θ_r and ϕ_r[3].

To consider a shader from the standpoint of energy balance, we start from the equation of energy balance given in [sieg81]:

$$dE = L_i \, (\mathbf{N} \cdot \mathbf{S}) \, d\omega_i \qquad (1)$$

where dE is the change in irradiance as a result of the illumination of the patch from $d\omega_i$. Using the definition of f_r, the BRDF, this change in irradiance gives rise to a change dL_r in L_r, the resulting radiance of the patch:

$$dL_r = f_r(\mathbf{S}, \mathbf{V}) \, dE \qquad (2)$$

In general, f_r is a function of the incident direction

[3] For the time being, we'll assume ϕ_i and ϕ_r are measured from some locally-defined frame of reference.

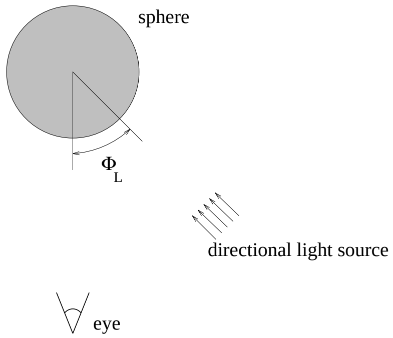

Figure 2: Shader test configuration

(θ_i, ϕ_i) and the reflected direction (θ_r, ϕ_r)[4]. For the sake of brevity, we'll write it as $f_r(\mathbf{S}, \mathbf{V})$.

As we have assumed an opaque, nonemissive surface, the only contributions to L_r can come from $\Omega_{\mathbf{N}}$. Therefore, we can integrate $d\omega_i$ over $\Omega_{\mathbf{N}}$ to get

$$L_r = \int_{\Omega_{\mathbf{N}}} f_r(\mathbf{S}', \mathbf{V})\, L_i\, (\mathbf{N} \cdot \mathbf{S}')\, d\omega_i' \tag{3}$$

Throughout this paper, we'll add a $'$ to bound variables of integration.

3 Some Typical Shaders

In this section, we'll review several shaders in use in computer graphics. Time does not permit us to consider some of the more elaborate ones (such as [he91]), but as these are not yet in wide use, perhaps the reader will forgive the oversight.

3.1 Phong Shaders

Phong shaders are generally of the form (see, for example, [fole90], equation [16.15]):

$$L_r = k_a\, L_a + [k_d\, (\mathbf{N} \cdot \mathbf{S}) + k_s\, F_s(\mathbf{S}, \mathbf{V})]\, L_d \tag{4}$$

where k_a is the "ambient" reflection coefficient, L_a is an ambient radiance uniformly distributed over $\Omega_{\mathbf{N}}$[5], k_d is the diffuse reflection coefficient, k_s is the specular reflection coefficient, and L_d is the (incident) irradiance from a directional light source.

There are two popular choices for F_s. The original one given in [phon75] corresponds to

$$F_s^P(\mathbf{S}, \mathbf{V}) = \begin{cases} (\mathbf{R}_l \cdot \mathbf{V})^{n_s^P} & \text{if } (\mathbf{R}_l \cdot \mathbf{V}) > 0 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where $\mathbf{R}_l = 2\mathbf{N}(\mathbf{N} \cdot \mathbf{S}) - \mathbf{S}$ is the "reflected" light source direction and n_s^P is the Phong specular reflection exponent.

The other popular choice for F_s comes from [blin77]:

$$F_s^B(\mathbf{S}, \mathbf{V}) = (\mathbf{N} \cdot \mathbf{H})^{n_s^B} \tag{6}$$

where $\mathbf{H} = (\mathbf{S} + \mathbf{V}) / |\mathbf{S} + \mathbf{V}|$ is the unit vector halfway between $\mathbf{S}$ and $\mathbf{V}$ and n_s^B is the Blinn specular reflection exponent.

Let's see how these shaders correspond to (3).

Let the patch be illuminated by the combination of a directional source of radiance L_d in direction $\mathbf{S}$ (i.e., $\theta_i' = \theta_i$, $\phi_i' = \phi_i$) on $\Omega_{\mathbf{N}}$ and a radiance L_a constant over $\Omega_{\mathbf{N}}$. We can model the resulting incident radiance as

$$L_i = L_a + L_d\, \delta(\cos\theta_i' - \cos\theta_i)\, \delta(\phi_i' - \phi_i)$$

If we plug this into (3), we get

$$\begin{aligned} L_r &= L_a \int_{\Omega_{\mathbf{N}}} f_r(\mathbf{S}', \mathbf{V})\, (\mathbf{N} \cdot \mathbf{S}')\, d\omega_i' \\ &\quad + L_d\, f_r(\mathbf{S}, \mathbf{V})\, (\mathbf{N} \cdot \mathbf{S}) \end{aligned} \tag{7}$$

Our goal is to make (4) and (7) equivalent for all possible L_a and L_d. This means that the specular terms of both equations must be equal, so

$$f_r(\mathbf{S}, \mathbf{V}) = k_d + k_s \frac{F_s(\mathbf{S}, \mathbf{V})}{(\mathbf{N} \cdot \mathbf{S})} \tag{8}$$

This will allow us to convert a Phong shader into its corresponding BRDF.

[4] f_r may also vary over the surface, but that variation is usually treated as part of texturing rather than shading, so we'll ignore it here. As mentioned above, we're also ignoring its possible dependence on wavelength.

[5] For the sake of simplicity, we've assumed that dA has an unobstructed view of $\Omega_{\mathbf{N}}$.

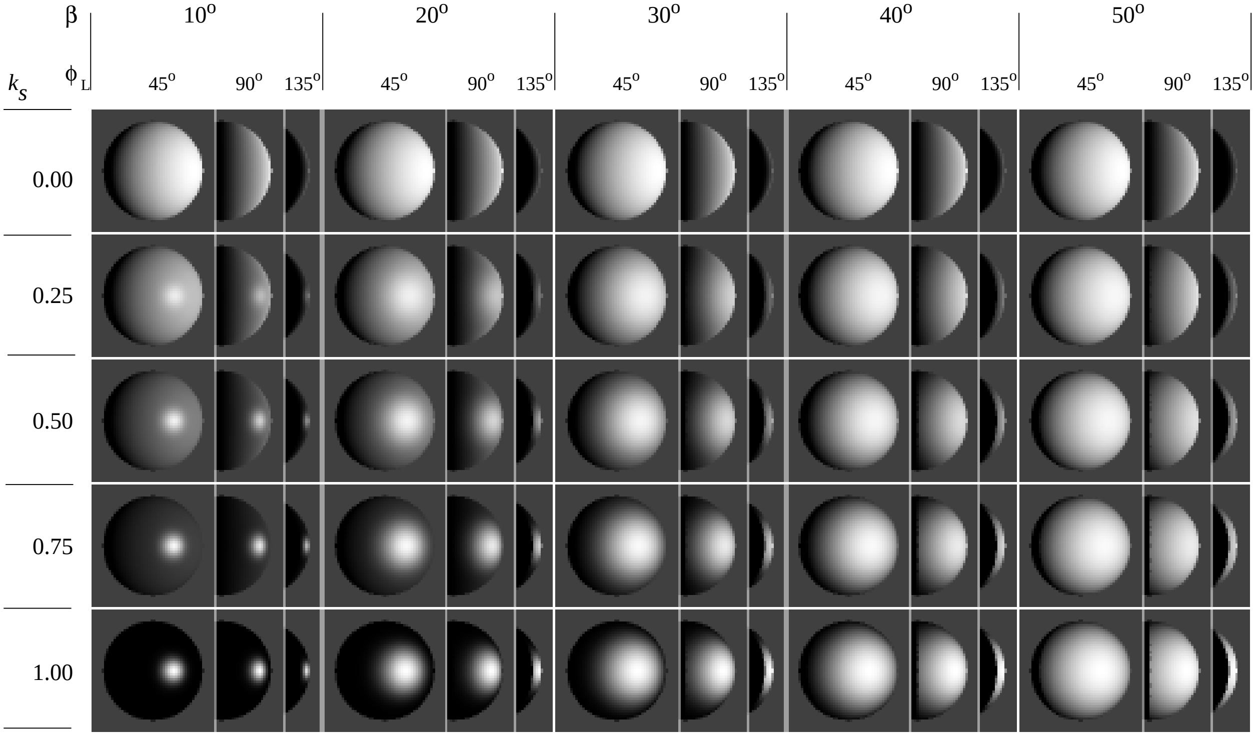

Figure 3: Sphere shaded with a Phong shader, using Blinn's F_s^B, for an assortment of incident angles with respect to the viewer (Φ_L), specular coefficients (k_s), and specular distribution half-angles (β).

For the purposes of comparison of the various models, let us adopt the (common) test configuration whose geometry is shown in Figure 2. A single directional light source shines at a sphere. Viewed from the center of the sphere, the light source is located at angle Φ_L from the viewing direction.

Figure 3 shows a series of images generated with the test configuration with a Phong shader using F_s^B. In this series, k_s varies between 0 and 1 and k_d is taken to be $1-k_s$. The angular distribution of the specular peak is qualitatively characterized[6] by the specular half-angle β, defined by

$$F_s^B = \frac{1}{2} = \cos^{n_s^B} \beta$$

Hence, for a given β,

$$n_s^B = -\frac{\ln 2}{\ln \cos \beta}$$

The ambient terms of (4) and (7) must also be equal[7]. If we equate these and notice that $\int_{\Omega_{\mathbf{N}}} (\mathbf{N} \cdot \mathbf{S}') d\omega_i' = \pi$, we get

$$k_a = k_d \, \pi + k_s \, G_s(\mathbf{V}) \qquad (9)$$

where we have defined

$$G_s(\mathbf{V}) = \int_{\Omega_{\mathbf{N}}} F_s(\mathbf{S}', \mathbf{V}) d\omega_i'$$

So consistency demands that we have only two degrees of freedom in selecting k_a, k_d, and k_s. In what follows, we'll take k_a to be dependent upon k_d and k_s.

[6] Only in the case of Phong shading with F_s^P does β have a direct and obvious geometrical interpretation, but, as we'll see (and [blin77] points out), it's qualitatively useful in other cases.

[7] We can look at this as a consistency constraint: The same BRDF we use to shade a directional light source must also shade an ambient light source.

3.2 Torrance-Sparrow Shaders

The other major class of shaders was first proposed in [torr67] and applied to computer-generated imagery in [blin77]. We shall, however, follow the development given in [cook82].

Torrance-Sparrow shaders can be formulated directly in terms of their BRDF:

$$f_r(\mathbf{S},\mathbf{V}) = \frac{FGD}{\pi\,(\mathbf{N}\cdot\mathbf{S})\,(\mathbf{N}\cdot\mathbf{V})} \tag{10}$$

where F is the Fresnel coefficient, G is the geometrical attenuation factor, and D is the facet slope distribution function.

The Fresnel coefficient for unpolarized light and zero extinction ([cook82] ignores extinction) is

$$F = \frac{(g-c)^2}{2(g+c)^2}\left[1 + \frac{(c\,(g+c)-1)^2}{(c\,(g-c)+1)^2}\right]$$

where $c = (\mathbf{V}\cdot\mathbf{H})$, $g^2 = n^2 + c^2 - 1$, and n is the index of refraction.

The geometrical attenuation factor is

$$G = \min\left\{1, \frac{2\,(\mathbf{N}\cdot\mathbf{H})\,(\mathbf{N}\cdot\mathbf{V})}{(\mathbf{V}\cdot\mathbf{H})}, \frac{2\,(\mathbf{N}\cdot\mathbf{H})\,(\mathbf{N}\cdot\mathbf{S})}{(\mathbf{V}\cdot\mathbf{H})}\right\}$$

There are several choices for the facet slope distribution function. [blin77] suggests three of them. The first corresponds to a Phong shader:

$$D_1 = b_1 \cos^{c_1}\alpha$$

where $\cos\alpha = (\mathbf{N}\cdot\mathbf{H})$. The second is the Gaussian one originally used in [torr67]:

$$D_2 = b_2 e^{-(c_2\alpha)^2}$$

The third is from [trow67]:

$$D_3 = b_3\left[\frac{c_3^2}{\cos^2\alpha(c_3^2-1)+1}\right]^2$$

In all of these, the b's are arbitrary constants analogous to the k's in Phong shaders. The c's (empirically) determine the width of the spectral lobe. As [blin77] observes, if we define β to be the value of α at which a distribution drops to half its peak value, we have

$$\begin{aligned} c_1 &= -\frac{\ln 2}{\ln\cos\beta} \\ c_2 &= \frac{\sqrt{\ln 2}}{\beta} \\ c_3 &= \sqrt{\frac{\cos^2\beta - 1}{\cos^2\beta - \sqrt{2}}} \end{aligned}$$

[cook82] considers an additional possibility originating with [beck63], which we'll include here as

$$D_4 = \frac{1}{4m^2\cos^4\alpha} e^{-\left(\frac{1-\cos^2\alpha}{m^2\cos^2\alpha}\right)}$$

where m is the RMS slope of the surface. Unlike D_1 - D_3, there is no arbitrary b constant for this distribution. The relationship of m to the corresponding value of β is

$$m = \frac{\tan\beta}{\sqrt{\ln 2 - 4\ln\cos\beta}}$$

3.3 Neumann-Neumann Shaders

In [neum89], Neumann and Neumann discuss "separable" shaders (i.e., those whose BRDF is of the form $a(\mathbf{S})\,r(\mathbf{V})$ for some functions a and r) and how their use can speed up radiosity computation in non-diffuse environments. As an example, they describe a "lacquer model" of a purely diffuse material covered by a semi-transparent "lacquer" that absorbs but does not scatter light that passes through it. The resulting BRDF they derive is

$$f_r(\mathbf{S},\mathbf{V}) = c\exp\left\{-s\left(\frac{1}{(\mathbf{N}\cdot\mathbf{S})} + \frac{1}{(\mathbf{N}\cdot\mathbf{V})}\right)\right\}$$

where c and s are constants that characterize the model. We can make this comparable with (8) and (10) by defining b_N as the value of f_r at $\mathbf{S} = \mathbf{V} = \mathbf{N}$. The equation then becomes:

$$f_r(\mathbf{S},\mathbf{V}) = b_N\exp\left\{-s\left(\frac{1}{(\mathbf{N}\cdot\mathbf{S})} + \frac{1}{(\mathbf{N}\cdot\mathbf{V})} - 2\right)\right\}$$

As we did before, we can relate s to a more geometrically meaningful quantity β that qualitatively

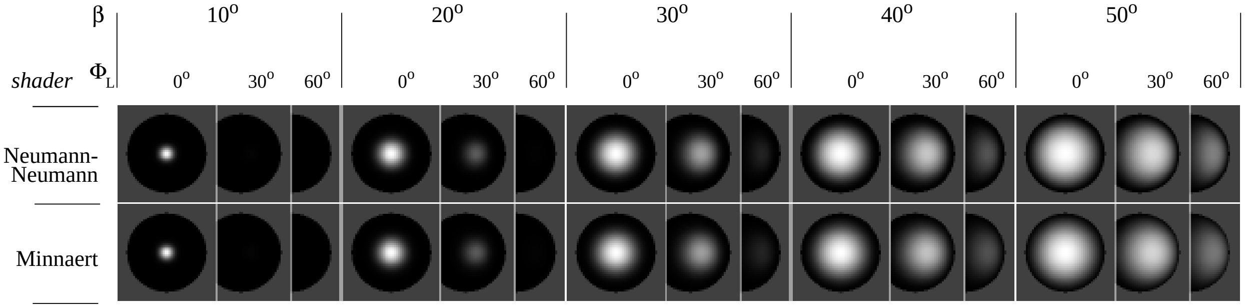

Figure 4: Sphere shaded with Neumann-Neumann and Minnaert shaders for an assortment of incident angles with respect to the viewer (Φ_L) and specular distribution half-angles (β).

measures the width of the spectral peak. Keep the illumination normal ($\mathbf{S} = \mathbf{N}$) and increase the angle between $\mathbf{V}$ and $\mathbf{N}$ until f_r drops to half of its maximum (i.e. $\mathbf{V} = \mathbf{N}$) value. We define the resulting angle to be β. We can relate s to β:

$$s = -\frac{\ln 2}{1 + \cos\beta}$$

Figure 4 shows what Neumann-Neumann shaders (and the subsequently-discussed Minnaert shaders) look like when applied to a sphere. Incident light for each shader is scaled to produce a peak unsaturated radiance at normal incidence ($\Phi_L = 0$) and then held constant as Φ_L and β are varied.

Note that as Φ_L increases, the image radiance decreases, unlike Phong shaders. Also notice that the limb of the sphere ($(\mathbf{N} \cdot \mathbf{V}) = 0$) is always dark.

Neumann-Neumann shaders exhibit undesirable behavior when being applied to a specular surface. Although, as Figure 4 shows, they produce an acceptable specular peak, for a given incident angle the resulting radiance always peaks in the normal direction. Especially for a highly specular surface, we should expect the radiance to peak somewhat closer to the reflected direction.

3.4 Minnaert Shaders

In [minn41], Minnaert describes a shader derived from observations of the Moon. His original model is

$$L_r = b_M (\mathbf{N} \cdot \mathbf{S})^k (\mathbf{N} \cdot \mathbf{V})^{k-1} L_i$$

For some constants b_M and k. This corresponds to the BRDF

$$f_r(\mathbf{S}, \mathbf{V}) = b_M ((\mathbf{N} \cdot \mathbf{S})\,(\mathbf{N} \cdot \mathbf{V}))^{k-1}$$

We can relate k to an angle β defined as in the previous section:

$$k = 1 - \frac{\ln 2}{\ln \cos\beta}$$

Figure 4 contrasts Minnaert shaders with Neumann-Neumann shaders. It is difficult to tell them apart. Their numerical values in these images differ by no more than 2%. (As a computational aside, since Minnaert shaders are also separable, this suggests that a Minnaert shader should be able to take the place of a Neumann-Neumann shader with fewer arithmetic operations in most cases, especially if k is an integer.)

Minnaert shaders exhibit the same undesirable behavior as Neumann-Neumann shaders applied to a specular surface: the resulting radiance peaks in the normal direction.

4 Energy Conservation

The first physical constraint we'll examine with respect to shaders is that of energy conservation. Physically plausible shaders must obey energy conservation. In a steady-state situation, energy conservation

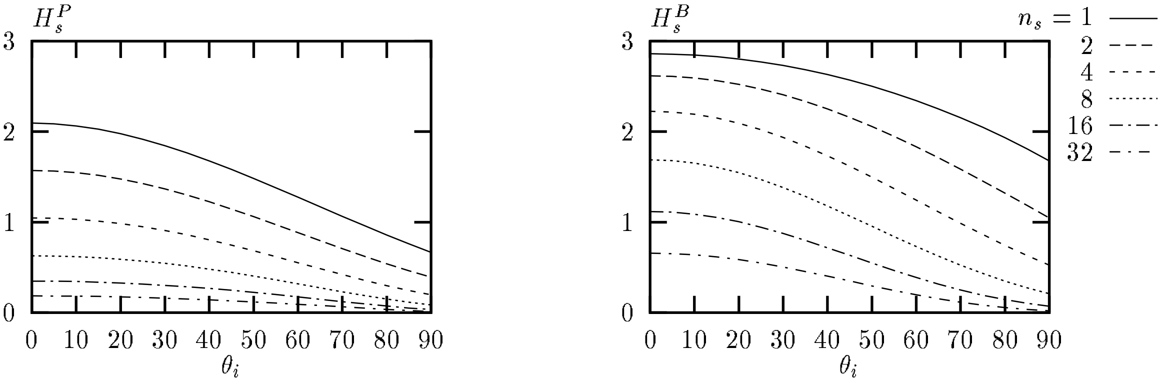

Figure 5: Specular integrals $H_s(\mathbf{S})$ for Phong's F_s^P (left) and Blinn's F_s^B (right)

is synonymous with power conservation. The total amount of power reflected, i.e., $M\,dA$, where M is the exitance, must be less than or equal to the total power incident $E\,dA$, where E is the irradiance. Hence,

$$M \leq E \tag{11}$$

From [sieg81], we have an equation similar to (1) describing the change in exitance dM due to a reflected radiance L_r radiated into an infinitesimal solid angle $d\omega_r$ around a direction $\mathbf{V}$:

$$dM = (\mathbf{N} \cdot \mathbf{V})\, L_r \, d\omega_r$$

We substitute (3) into this and integrate over $\Omega_{\mathbf{N}}$ to get

$$M = \int_{\Omega_{\mathbf{N}}} \int_{\Omega_{\mathbf{N}}} f_r(\mathbf{S}', \mathbf{V}')\, L_i\, (\mathbf{N} \cdot \mathbf{S}')\, (\mathbf{N} \cdot \mathbf{V}')\, d\omega_i'\, d\omega_r' \tag{12}$$

We can also integrate (1) to get

$$E = \int_{\Omega_{\mathbf{N}}} L_i\, (\mathbf{N} \cdot \mathbf{S}')\, d\omega_i' \tag{13}$$

So, if we make the trivial assumption that $E > 0$, we divide both sides of (11) by E to get

$$\frac{\int_{\Omega_{\mathbf{N}}} \int_{\Omega_{\mathbf{N}}} f_r(\mathbf{S}', \mathbf{V}')\, L_i\, (\mathbf{N} \cdot \mathbf{S}')\, (\mathbf{N} \cdot \mathbf{V}')\, d\omega_i'\, d\omega_r'}{\int_{\Omega_{\mathbf{N}}} L_i\, (\mathbf{N} \cdot \mathbf{S}')\, d\omega_i'} \leq 1 \tag{14}$$

Energy conservation does not depend upon the particular L_i distribution. Given any L_i, (14) must hold, so, as we did with the Phong shaders, let's try a δ-function for L_i of the form

$$L_i = L_d\, \delta(\cos\theta_i' - \cos\theta_i)\, \delta(\phi_i' - \phi_i)$$

to represent a directional source of radiance L_d in a direction $\mathbf{S}$. According to [ans86], M/E in this case becomes the "directional-hemispherical reflectance", which we'll refer to as k_ρ[8]. Integrating the δ-functions and cancelling out common factors, we get

$$k_\rho = \int_{\Omega_{\mathbf{N}}} f_r(\mathbf{S}, \mathbf{V}')\, (\mathbf{N} \cdot \mathbf{V}')\, d\omega_r' \leq 1 \tag{15}$$

4.1 Making Phong Shaders Conserve Energy

Let's apply these results to a Phong shader to see what constraint(s) energy conservation leads to. Noting that

$$\int_{\Omega_{\mathbf{N}}} (\mathbf{N} \cdot \mathbf{V}')^\gamma \, d\omega_r' = \frac{2\pi}{\gamma + 1} \tag{16}$$

(12) becomes

$$M = L_d\, [k_d\, \pi\, (\mathbf{N} \cdot \mathbf{S}) + k_s\, H_s(\mathbf{S})] \tag{17}$$

[8] In [neum89], this is referred to as "albedo", but that usage is imprecise as the definition of that term does not require a unidirectional source. In addition, "albedo" is not defined in [ans86].

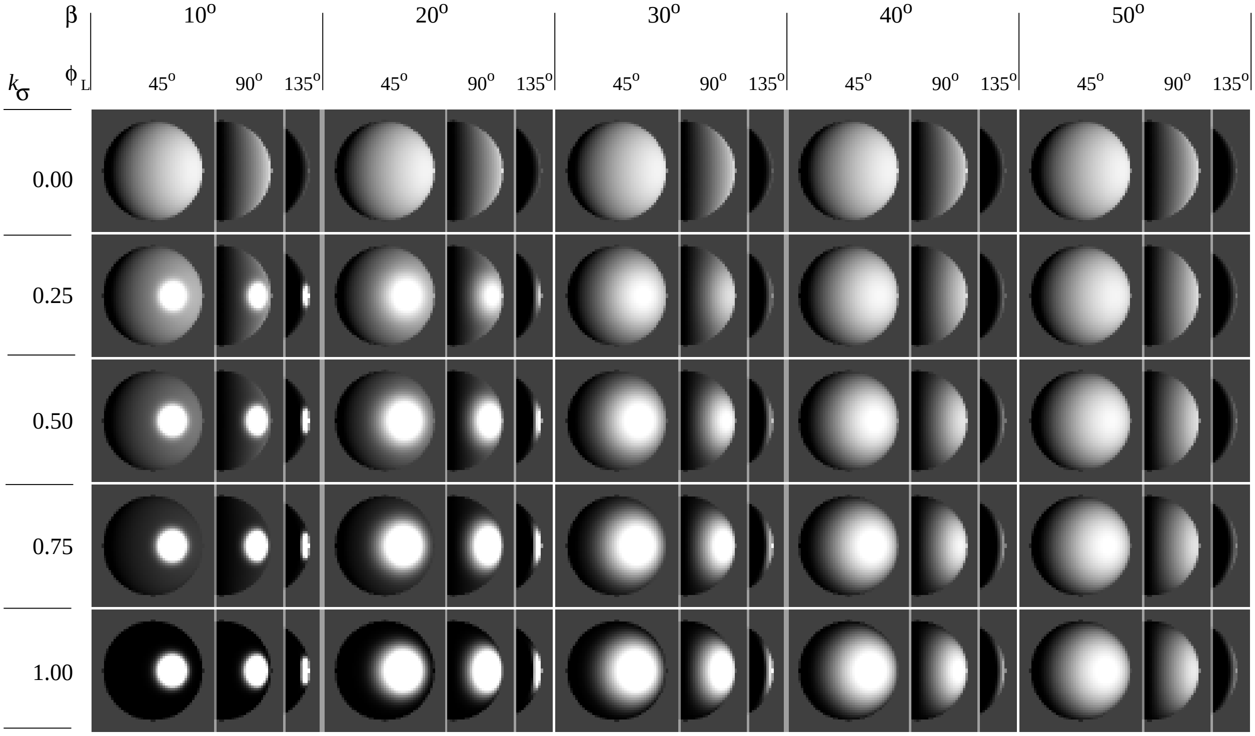

Figure 6: Sphere shaded with an energy-conserving Phong shader, using Blinn's F_s^B, for an assortment of incident angles with respect to the viewer (Φ_L), specular fractions (k_σ), and specular distribution half-angles (β).

where we have defined

$$H_s(\mathbf{S}) = \int_{\Omega_\mathbf{N}} F_s(\mathbf{S}, \mathbf{V}')\,(\mathbf{N} \cdot \mathbf{V}')d\omega'_r$$

Figure 5 shows H_s evaluated numerically using both (5) and (6) for a variety of specular exponents. Note that H_s is a function of the incident direction and specular exponent only and that it can be thought of as an integral operator acting on a given F_s.

As we might expect, (13) becomes

$$E = L_d\,(\mathbf{N} \cdot \mathbf{S}) \qquad (18)$$

so that

$$k_\rho = k_d\,\pi + k_s\,\frac{H_s(\mathbf{S})}{(\mathbf{N} \cdot \mathbf{S})} \qquad (19)$$

To guarantee energy conservation regardless of illumination geometry, it is necessary to guarantee that $k_\rho \leq 1$ for all incident directions. But there's a problem here. Given the F_s's in (5) and (6) and regardless of $\mathbf{S}$, it is always the case that $F_s \geq 0$ and, furthermore, there is always some nonvanishing region of $\Omega_\mathbf{N}$ over which $F_s > 0$. That means that H_s is always > 0, as Figure 5 illustrates. So that if $k_s > 0$, it is always possible to choose θ_i close enough to 90^o that k_ρ will be greater than one. We therefore conclude that the specular terms of Phong shaders do not conserve energy at sufficiently large incident angles.

After [neum89], let's consider a different formulation of a shader. Start from (4), but suppose that, instead of being constant, we allowed k_s to vary with $\mathbf{S}$ in such a way that energy conservation was maintained. (As (19) shows, we're not getting any trouble from the diffuse term, so we'll leave it alone.)

Let k_σ be the fraction of exitance that is reflected specularly:

$$k_\sigma \equiv \frac{\int_{\Omega_\mathbf{N}} dM_{spec}}{M} = \left(1 + \frac{k_d\,\pi\,(\mathbf{N} \cdot \mathbf{S})}{k_s\,H_s(\mathbf{S})}\right)^{-1} \qquad (20)$$

We can solve (19) and (20) for k_d and k_s:

$$k_d = \pi^{-1}\,k_\rho\,(1 - k_\sigma) \qquad (21)$$

$$k_s = \frac{k_\rho \, k_\sigma \, (\mathbf{N} \cdot \mathbf{S})}{H_s(\mathbf{S})}$$

so we can rewrite the BRDF for the new shader as

$$f_r(\mathbf{S}, \mathbf{V}) = k_\rho \left[\frac{1 - k_\sigma}{\pi} + \frac{k_\sigma \, F_s(\mathbf{S}, \mathbf{V})}{H_s(\mathbf{S})} \right] \qquad (22)$$

We can also construct the analogue of (4) to express this result in terms of radiance:

$$L_r = k_\alpha L_a + k_\rho \, (\mathbf{N} \cdot \mathbf{S}) \left[\frac{1 - k_\sigma}{\pi} + \frac{k_\sigma \, F_s(\mathbf{S}, \mathbf{V})}{H_s(\mathbf{S})} \right] L_d$$

where

$$k_\alpha = k_\rho \left[1 + k_\sigma \left(\int_{\Omega_\mathbf{N}} (\mathbf{N} \cdot \mathbf{S}') \frac{F_s(\mathbf{S}', \mathbf{V})}{H_s(\mathbf{S}')} d\omega_i' - 1 \right) \right]$$

corresponds to (9).

Figure 6 shows what such a shader looks like when applied to a sphere with a single directional light source and no ambient radiance. For this figure, we've used Blinn's F_s^B. We've also taken $k_\rho = 1$, since any other value would just be a uniform reduction by a constant factor in image radiance. Notice that, unlike Figures 3 and 4, the highly specular parts of the printed images are necessarily saturated in order to show the diffuse parts.

4.2 Do Torrance-Sparrow Shaders Conserve Energy?

Figure 7 shows some numerical integrations of (15), contrasting Phong shaders with Torrance-Sparrow shaders. All Torrance-Sparrow shaders were computed with a Fresnel factor $F = 1$ (i. e. a large index of refraction) to show the worst case.

One way to produce an energy-conserving Torrance-Sparrow shader suggests itself: simply choose any value of b_j such that

$$b_j < \frac{1}{\left[\frac{k_\rho^{Tj}}{b_j} \right]_{\max}}$$

where $\left[\frac{k_\rho^{Tj}}{b_j} \right]_{\max}$ is the maximum value as shown in Figure 7. (The Beckmann distribution is not a problem as long as its integral is always less than unity, and it has no b-coefficient to adjust anyway.)

Nevertheless, doing this would probably be a mistake. To see why, look at the plot for k_ρ^{T1}/b_1, the Torrance-Sparrow shader with the Phong microfacet distribution. Notice that it does not diverge as $\theta_i \rightarrow 90^o$, even though k_ρ^P/k_s, the corresponding Phong shader with a Phong specular term, *does* diverge. The same is true for k_ρ^{T2}/b_2 compared to k_ρ^B/k_s.

Why should this be? The answer lies in the geometrical attenuation factor G. As $\theta_i \rightarrow 90^o$, G is guaranteed to be less than or equal to unity and, if $(\mathbf{V} \cdot \mathbf{H}) > 0$ (i.e., $\mathbf{V}$ and $\mathbf{S}$ are not antiparallel), it will vanish in the limit.

But what does this really mean? If we go back to the derivation of the geometrical attenuation factor in [torr67], we see that G is designed to compensate for the blocking of light that falls on a facet and the masking of light that the facet reflects. The blocking and masking agents are themselves other facets.

This leads to a critical question for Torrance-Sparrow shaders and energy conservation: What happens to the light that gets blocked or masked? The shader does not treat secondary reflection. Instead, it acts as though the blocked or masked light were completely absorbed by the surface. This is unlikely.

For this reason, while it may be reasonable to consider the use of Torrance-Sparrow shaders as *ad hoc* basis functions to fit empirical data, as was done in [ward92], we should do so realizing that it's not really "fair" to use Torrance-Sparrow shaders in an energy-conserving context. Basis functions that properly account for blocked and masked light are needed, but we will not attempt to derive them here.

4.3 Making Neumann-Neumann and Minnaert Shaders Conserve Energy

Figure 8 shows some numerical integrations of (15) for Neumann-Neumann and Minnaert shaders. Contrast these with those of Figure 7.

Like the Torrance-Sparrow shaders, k_ρ is bounded in both cases, so we can put a limit on b_N or b_M to assure energy conservation.

In the case of a Minnaert shader, we can go a bit further and note that k_ρ can be determined analytically

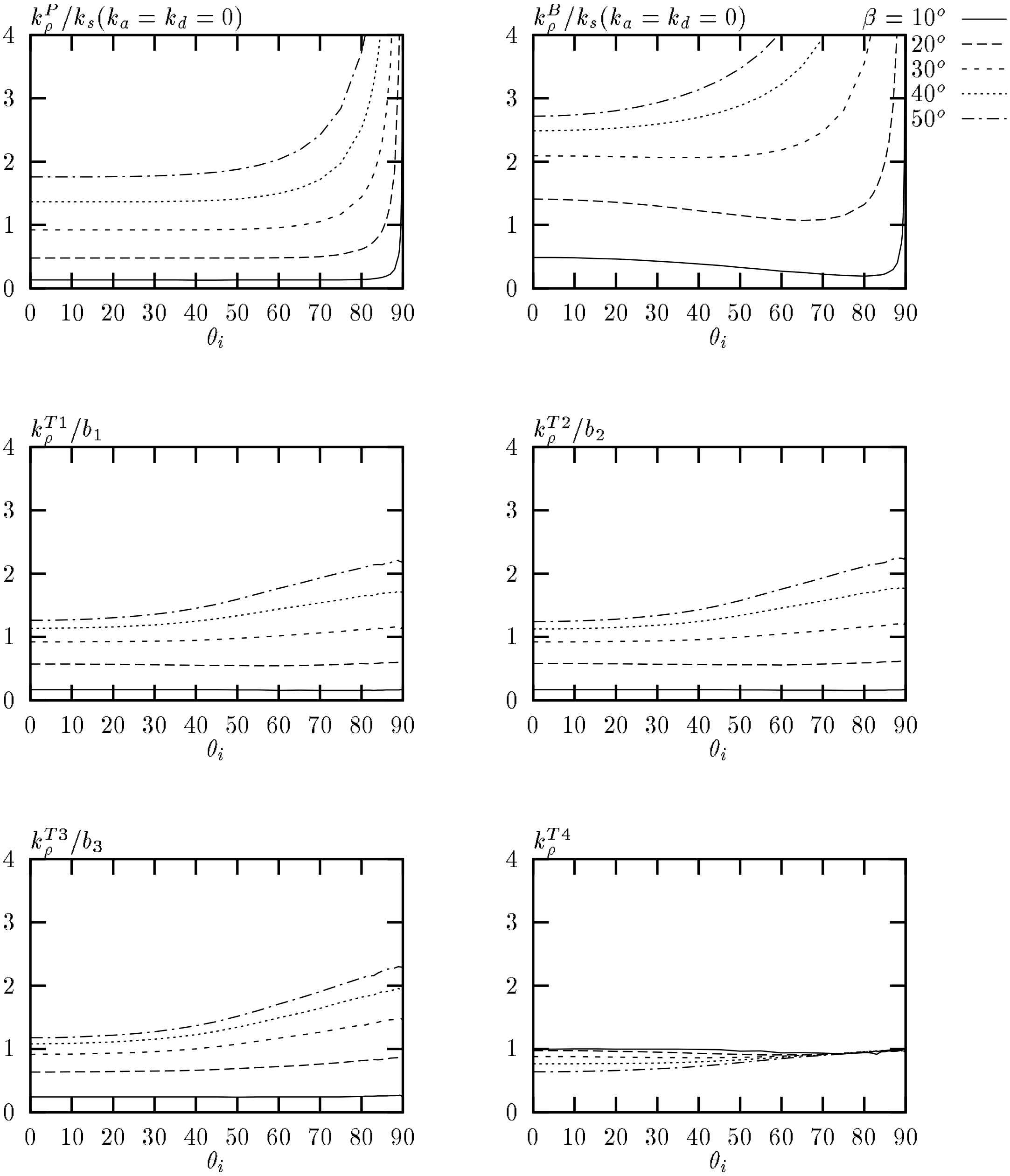

Figure 7: Directional-hemispherical reflectance for a Phong shader (k_ρ^P), a Blinn shader (k_ρ^B), and Torrance-Sparrow shaders with Phong (k_ρ^{T1}), original Torrance-Sparrow (k_ρ^{T2}), Trowbridge (k_ρ^{T3}), and Beckmann (k_ρ^{T4}), microfacet distributions

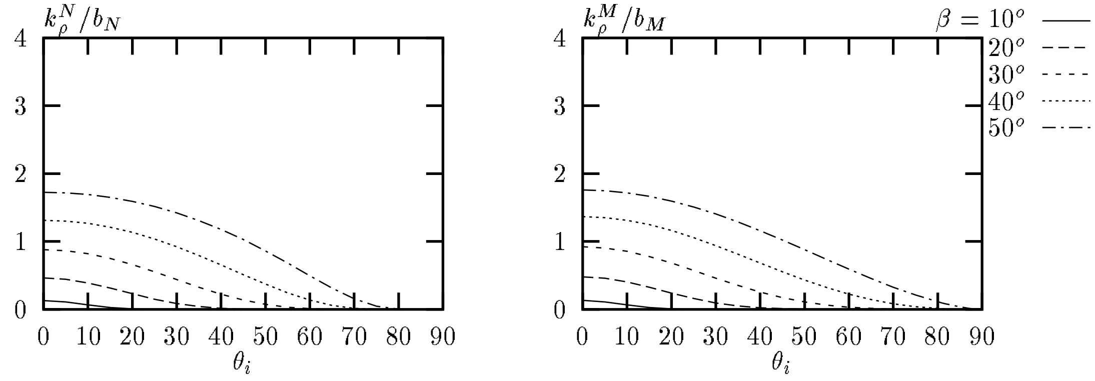

Figure 8: Directional-hemispherical reflectance for a Neumann-Neumann shader (k_ρ^N) and a Minnaert shader (k_ρ^M),

(using (16), as was done in [wood85]). The resulting BRDF can be formulated directly in terms of k_ρ:

$$f_r(\mathbf{S},\mathbf{V}) = k_\rho \frac{(k+1)}{2\pi}((\mathbf{N}\cdot\mathbf{S})\,(\mathbf{N}\cdot\mathbf{V}))^{k-1}$$

where, as always, any value of k_ρ between 0 and 1 will guarantee energy conservation.

5 Making Shaders Reciprocal

The second physical constraint we'll examine with respect to shaders is that of Helmholtz reciprocity. A physically plausible shader ought to obey Helmholtz reciprocity (see [sieg81]). In terms of the BRDF, this means that

$$f_r(\mathbf{S},\mathbf{V}) = f_r(\mathbf{V},\mathbf{S}) \qquad (23)$$

for all $\mathbf{V}$ and $\mathbf{S}$ in $\Omega_\mathbf{N}$.

5.1 Are Phong Shaders Reciprocal?

Using the BRDF of a Phong shader given in (5), and expressing F_s in the functional form $F_s(\mathbf{S},\mathbf{V})$, we see that such a shader will be reciprocal if

$$\frac{F_s(\mathbf{S},\mathbf{V})}{(\mathbf{N}\cdot\mathbf{V})} = \frac{F_s(\mathbf{V},\mathbf{S})}{(\mathbf{N}\cdot\mathbf{S})}$$

Substitution of both F_s^P from (5) and F_s^B from (6) reveals that neither of these shaders is reciprocal[9].

Is our energy-conserving modified Phong shader reciprocal? Applying (23) to (22), we are asking if

$$\frac{F_s(\mathbf{S},\mathbf{V})}{H_s(\mathbf{S})} = \frac{F_s(\mathbf{V},\mathbf{S})}{H_s(\mathbf{V})}$$

Again, the answer is no for both F_s's.

5.2 Are Torrance-Sparrow Shaders Reciprocal?

By inspection, it's easy to see that the Torrance-Sparrow shaders are all reciprocal. This should come as no surprise, as the assumption of reciprocity was part of their derivation in [torr67]. Unfortunately, the arguments made above about their energy conservation still limits their plausibility.

[9] Even though $F_s^B(\mathbf{S},\mathbf{V}) = F_s^B(\mathbf{V},\mathbf{S})$.

5.3 Are Separable Shaders Reciprocal?

It's also easy to see by inspection that both Neumann-Neumann and Minnaert shaders are reciprocal. Again, this is because reciprocity was part of their derivation.

As with Torrance-Sparrow shaders, separable shaders could be used as *ad hoc* basis functions. Care needs to be taken, though, to retain reciprocity. Given two separable BRDF's $f_{r1}(\mathbf{S},\mathbf{V}) = a_1(\mathbf{S})r_1(\mathbf{V})$ and $f_{r2}(\mathbf{S},\mathbf{V}) = a_2(\mathbf{S})r_2(\mathbf{V})$, a simple linear combination of the form $f_r(\mathbf{S},\mathbf{V}) = c_1 f_{r1}(\mathbf{S},\mathbf{V}) + c_2 f_{r2}(\mathbf{S},\mathbf{V})$ for some constants c_1 and c_2 is not, in general, reciprocal.

One way to guarantee reciprocity is to linearly combine the separable parts rather than their product:

$$f_r(\mathbf{S},\mathbf{V}) = (c_{1a}a_1(\mathbf{S})+c_{2a}a_2(\mathbf{S}))(c_{1r}r_1(\mathbf{V})+c_{2r}r_2(\mathbf{V}))$$

The trouble with this is that no matter how many terms we add, the resulting BRDF will always have the property of having the reflected radiance peak in the normal direction. If the data we attempt to fit does not have this property, we should not expect a good fit.

6 An Energy-Conserving, Reciprocal Shader

Objections can be raised to all of the shaders we're presented so far, either on the grounds of implausibility (Phong) or of behavior that, while plausible, is unlikely to fit a real BRDF (Torrance-Sparrow, Neumann-Neumann, Minnaert).

Consider instead a Phong shader formulated like (8), but using Blinn's F_s^B and omitting the $(\mathbf{N}\cdot\mathbf{S})$ in the denominator of the specular term:

$$f_r(\mathbf{S},\mathbf{V}) = k_d + k_s F_s^B(\mathbf{S},\mathbf{V}) \tag{24}$$

Obviously, since F_s^B is reciprocal, this BRDF is reciprocal.

Figure 9 shows the resulting k_ρ. It is bounded, so we can always conserve energy by limiting k_s and k_d. (Unfortunately, we can't formulate the shader in terms of k_ρ and k_σ as we did above, since doing this makes the shader non-reciprocal.)

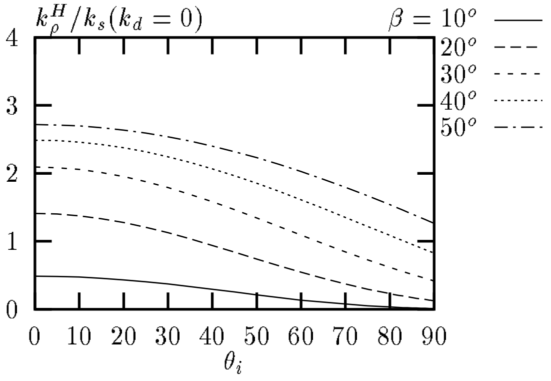

Figure 9: Directional-hemispherical reflectance for a reciprocal Phong shader, using Blinn's F_s^B (k_ρ^H)

Figure 10 shows some images produced with a reciprocal Phong shader. As in Figure 3, k_s varies between 0 and 1 and k_d is taken to be $1 - k_s$.

While resembling Figure 3, the images for large Φ_L are dimmer, as we might expect from the absence of the $(\mathbf{N}\cdot\mathbf{S})$ in the specular denominator. Nevertheless, they are not as diminished as those of the separable shaders in Figure 4 (which doesn't even bother showing $\Phi_L > 60^o$).

7 Summary

We have examined a number of shaders commonly used in graphics, looking at their plausibility in terms of energy conservation and reciprocity. Our results are summarized in Table 2.

As originally defined, Phong shaders fail on both counts. It is possible to modify a Phong shader to conserve energy and even, as shown in (22), have an energy-based parameterization, but this rules out satisfying reciprocity.

Torrance-Sparrow shaders are reciprocal and appear to conserve energy, but their underlying derivation fails to account for blocked and masked energy. They

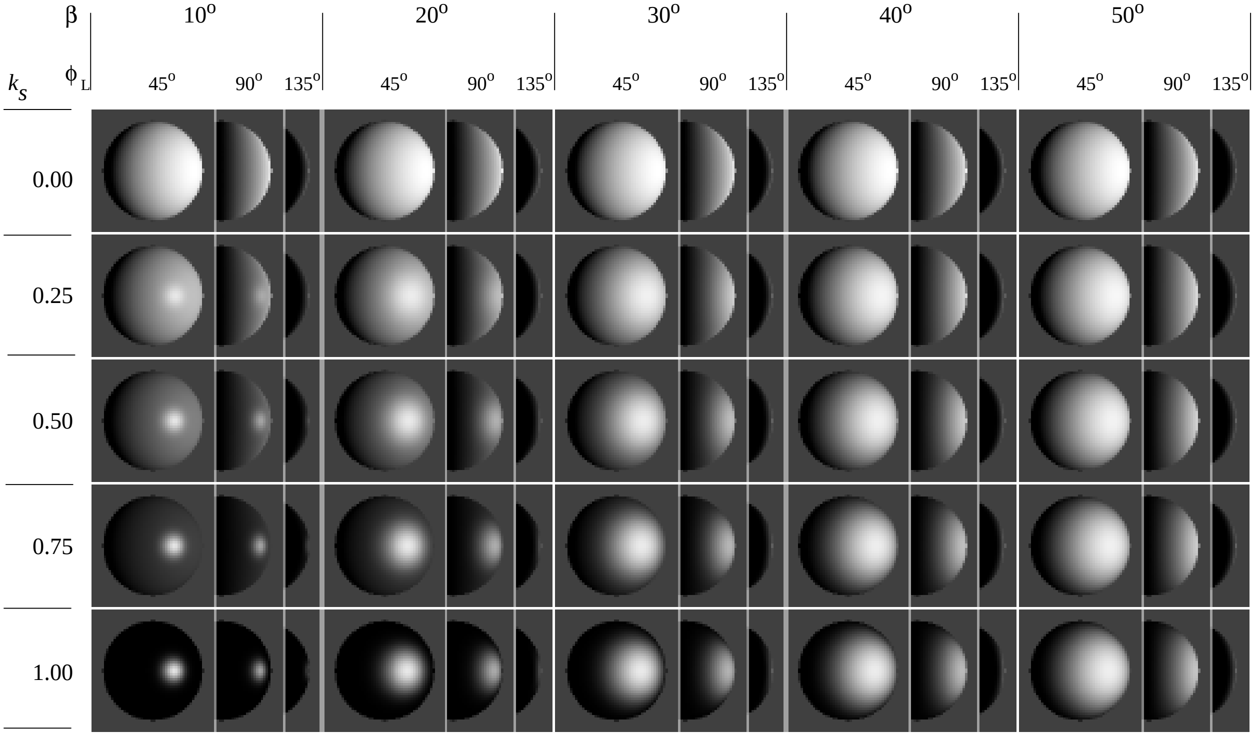

Figure 10: Sphere shaded with a reciprocal Phong shader for an assortment of incident angles with respect to the viewer (Φ_L), specular coefficients (k_s), and specular distribution half-angles (β).

may still be useful, however, as *ad hoc* basis functions.

Neumann-Neumann and Minnaert shaders are similar. Both are plausible: they conserve energy and are reciprocal. Minnaert shaders have been used successfully to fit radiometric data. While it would be worth trying one of them as a basis, we expect that they will prove less useful with highly specular surfaces because both shaders peak undesirably in the normal direction.

A differenly-modified Phong shader given in (24) is reciprocal and can be constrained to conserve energy. In fact, we can observe that a diffuse surface is the special case $n_s^B = 0$ to suggest that it would be useful to attempt to fit real data with a power series in $(\mathbf{N} \cdot \mathbf{H})$ of the form:

$$f_r(\mathbf{S}, \mathbf{V}) = \sum_{i=0}^{i_{max}} k_n (\mathbf{N} \cdot \mathbf{H})^{n_i}$$

for some suitable sequence $\{n_0 ... n_{i_{max}}\}$.

References

[ans86] American National Standards Institute/Illuminating Engineering Society of North America. *Nomenclature and Definitions for Illuminating Engineering*, ansi/ies rp-16-1986 edition, June 1986.

[beck63] P. Beckmann and A. Spizzichino. *The scattering of electromagnetic waves from rough surfaces.* MacMillan, 1963.

[blin77] James F. Blinn. "Models of Light Reflection For Computer Synthesized Pictures". *Computer Graphics (SIGGRAPH '77 Proceedings)*, Vol. 11, No. 2, pp. 192–198, July 1977.

[cook82] R.L. Cook and K.E. Torrance. "A Reflectance Model for Computer Graphics". *ACM Transactions on Graphics*, Vol. 1, No. 1, pp. 7–24, January 1982.

[fole90] J.D. Foley, A. van Dam, Steven K. Feiner, and John F. Hughes. *Computer Graphics:*

Shader	Plausiblity		Other
Class	Conserves Energy?	Reciprocal?	Objections
Phong	no	no	
Energy-Conserving Phong	yes	no	
Torrance-Sparrow	yes	yes	no secondary reflection
Neumann-Neumann	yes	yes	L_r always peaks at $\theta_r = 0$
Minnaert	yes	yes	L_r always peaks at $\theta_r = 0$
Reciprocal Phong-Blinn	yes	yes	

Table 2: Summary of Results

Principles and Practice. Addison-Wesley Publishing Company, second edition, 1990.

[hall89] R. Hall. *Illumination and Color in Computer Generated Imagery.* Springer-Verlag, 1989.

[he91] Xiao D. He, Kenneth E. Torrance, Francois X. Sillion, and Donald P. Greenberg. "A comprehensive physical model for light reflection". *Computer Graphics (SIGGRAPH '91 Proceedings)*, Vol. 25, No. 4, pp. 175–186, July 1991.

[kaji86] J.T. Kajiya. "The Rendering Equation". *Computer Graphics (SIGGRAPH '86 Proceedings)*, Vol. 20, No. 4, pp. 143–150, August 1986.

[minn41] M. Minnaert. "The Reciprocity Principle in Lunar Photometry". *Astrophysical Journal*, Vol. 93, pp. 403–410, May 1941.

[neum89] Laszlo Neumann and Attila Neumann. "Photosimulation: interreflection with arbitrary reflectance models and illumination". *Computer Graphics Forum*, Vol. 8, No. 1, pp. 21–34, March 1989.

[phon75] Bui-T. Phong. "Illumination for Computer Generated Pictures". *Communications of the ACM*, Vol. 18, No. 6, pp. 311–317, June 1975.

[sieg81] Robert Siegel and John R. Howell. *Thermal Radiation Heat Transfer.* Hemisphere Publishing Corporation, 1981.

[torr67] K.E. Torrance and E.M. Sparrow. "Theory for Off-Specular Reflection from Roughened Surfaces". *Journal of Optical Society of America*, Vol. 57, No. 9, 1967.

[trow67] T.S. Trowbridge and K.P. Reitz. "Average Irregularity Representation of a Roughened Surfaces for Ray Reflection". *Journal of Optical Society of America*, Vol. 65, No. 3, 1967.

[ward92] Gregory J. Ward. "Measuring and modeling anisotropic reflection". *Computer Graphics (SIGGRAPH '92 Proceedings)*, Vol. 26, No. 2, pp. 265–272, July 1992.

[wood85] R. J. Woodham and T. K. Lee. "Photometric Method for Radiometric Correction of Multispectral Scanner Data". *Canadian Journal of Remote Sensing*, Vol. 11, No. 2, pp. 132–161, December 1985.