Real-time polygonal-light shading with linearly transformed cosines

Samuel Laveau, Raphaël Jorel

October 21, 2017

1 Introduction

Physically-based shading with polygonal lights evaluation in real time is not an easy problem to solve, and state of the art techniques are likely to involve either heavy process or unaccurate approximations. Indeed shading with area lights is complicated because it involves integrating a spherical distribution over a spherical polygon. This is complex because of the real time factor. Indeed we could easily compute the integral we need by tracing rays to the polygon and testing the intersections with the unit sphere (Monte Carlo sampling) but it will be too time consuming for real time applications. So what we need is an analytic solution. The first difficulty is to find a distribution that can faithfully represent a Bidirectional Reflectance Distribution Function (BRDF). In other words this distribution has to cover all-frequency signals in order to ensure a result similar to real-world lights with any type of materials. Furthermore some properties from the real world, such as anisotropy or skewness, are tricky to handle with simple distributions. Another related difficulty is to ensure some practical properties such as analytic operators (mainly for integration but product or convolution may be useful).

State-of-the-art real-time polygonal-light shading (before this paper) turns around some specific yet sophisticated distributions that we can integrate (with great efforts). However, this paper proposes a more general and simpler solution. Instead of dealing with an intricate integration they show that it is possible to change it to a simpler one. Throughout the demonstrations the authors prove that the integral of a spherical distribution over a polygon is invariant under simultaneous linear transformations over the direction vectors of the distribution and over the vertices of the polygon. So if we have a spherical distribution which is simple to integrate over a polygon, we know how to integrate all of its linear transformations. From this, the idea is simple: we just choose a well-defined spherical distribution that we can fit to a BRDF by linear transformations. Thereby, when we need to compute the illumination, instead of integrating this complicated new distribution, we just apply the reverse linear transformations to go back to the distribution we can easily integrate (the polygon will change). The authors' choice of concrete implementation is to use a clamped cosine as original distribution, and so they call the result of the BDRF fitting a Linearly Transformed Cosine (LTC).

2 Previous Work

As said earlier, previous work has focused on finding distributions which can be integrated over a polygon with a result close to physical lights in real-time. For the moment, analytic solutions to polygonal lighting have been found only for cosine-like distributions. Integration of a clamped cosine over a polygonal light with constant intensity was already solved by Lambert [1].

Arvo [2] extends Lambert's integration to arbitrary polygon, for Phong distribution. The main issue of this method is the exponent term of the Phong model. For very sharp material, the cost of the integration is too important to ensure a real-time rendering. Moreover, Phong is not the most realistic BRDF model, and Arvo is limited to this.

Lecoq [3] proposes to approximate Arvo integration by using certain functions (namely Loretzian, Pearson and Ellipsoid functions). Using approximations it is now possible to ensure real-time rendering, because the exponent term no longer appears in the computation complexity.
Because this method is based on [2], basically only the Phong model is handled, but it handles microfacet BRDFs too by adding some calculus to adapt the latter. In any case, the method is complicated to understand, and unpleasant artifacts are visible in certain conditions.

Spherical Gaussians. Spherical Gaussians (SGs) have often been used to represent the kind of spherical functions we are looking for. Indeed we can use them to represent all-frequency signals (détailler ? lobe sharpness) and it is possible to integrate them with a closed-form solution. However, despite these convenient properties we can't use a single raw SG to ensure a realistic BRDF because SGs are isotropic by nature when real lights are anisotropic. Yet it is possible to use a mixture of SGs to simulate the anisotropic behavior. This will lead to an appoximation more or less accurate depending on the number of SGs used. **Xu et al. 2013** [4] propose Anisotropic Spherical Gaussians (ASGs) which are inherited from the SGs properties we seek while being more accurate at representing anistropic effects in real time. However, ASGs are meant for environment lights and local point lights, and even there we have to use ASG mixtures to approximate a realistic result. About polygonal lights, **Xu et al.**

2014 [5] introduce a new piecewise linear approximation which allows to integrate an SG over a spherical triangle. Then it is possible to render complex scenes with a composition of triangles. However, the rendering with this method is too slow for real-time applications.

3 Method

3.1 Lighting basics

The general formula to compute the irradiance given a view direction ω_v and a light direction ω_l is given by:

$$I = \int_{\Omega} L(\omega_l)\rho(\omega_v, \omega_l)\cos\theta_l d\omega_l$$

where ρ is a BRDF function and L is the radiance emitted from ω_l. This equation was proposed by [6] and expresses the value of light on each considered point, according to the light and view directions. The BRDF function and cos term are used to weigh the incident light. In the paper, only non-emitting points are considered. Moreover, we want to evaluate the irradiance over the polygon P, therefore the domain Ω is P here.

The authors propose to compute irradiance value by using an approximation of the BRDF term, based on a spherical distribution. The relation is:

$$D \approx \rho(\omega_v, \omega_l)cos\theta_l$$

where D is the distribution. It transforms the formula as follows:

$$I \approx \int_P L(\omega_l)D(\omega_l)d\omega_l$$

The idea is therefore to replace the BRDF function and the cos term by a spherical distribution, because then it is possible to get constant-time evaluation of irrandiance with this reformulation. The only problem comes from the approximation, because the method can not provide perfect BRDF fitting.

Now, we need to know how this D term is computed.

3.2 A story of distribution

We consider an original spherical distribution D_0, which is oriented by ω_0. The authors propose to apply a 3×3 matrix M on ω_0, the direction vector of D_0. We get a new distribution D with the direction vector ω, given by $\omega = M\omega_0/\|M\omega_0\|$ (the transformed direction is normalized).

From that, the authors give the following equation:

$$D(\omega) = D_0(\omega_0)\frac{\partial \omega_0}{\partial \omega}$$

The term $\partial\omega_0/\partial\omega$ is the determinant of the Jacobian, that measures the change of the solid angle due to the distortion of the linear transformation. Since the authors do not detail the jacobian computation, we provide the complete derivation in appendix A.

To finish, the authors play with integration with a change of variable to retrieve the Jacobian term. This gives :

$$\int_P D(\omega)d\omega = \int_{M^{-1}P_0} D_0(\omega_0)\frac{\partial \omega_0}{\partial \omega} = \int_{P_0} D(\omega_0)d\omega_0 \quad (1)$$

Here we see the very important result to remember exposed in the paper. This result is really intuitive with images, and as a good picture is better than a long discussion, let us just take a look on figure 1.

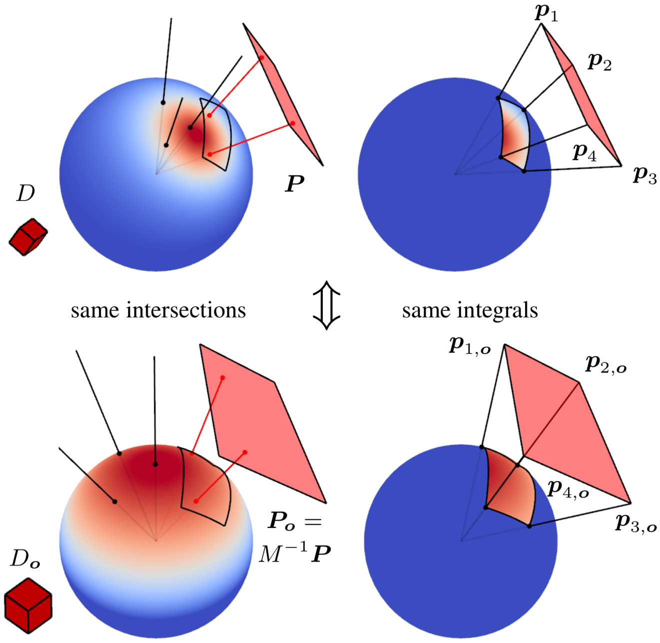

Figure 1: The bottom configurations are the top ones multiplied by M^{-1}. We see that the probability to intersect polygon stay the same, before and after the multiplication. Intuitively, we find that the integral over P is the same than over P_0.

3.3 Usage in practice

BRDF fitting. Now we have the method to approximate the BRDF term in equation of irradiance, we need to fit an existing BRDF. The authors fit the GGX microfacet BRDF, because it is the most realistic BRDF for the moment.
For any combination (ω_0, α), respectively the view direction and the roughness, the GGX BRDF is fitted with a Linear Transform Cosine. The authors show that the

transformation matrix can be represented with only 4 parameters, written as:

$$M = \begin{bmatrix} a & 0 & b \\ 0 & c & 0 \\ d & 0 & 1 \end{bmatrix}$$

So, for any pair (ω_0, α), only 4 parameters need to be stored. As we need the reverse transformation matrices in the equation 1, the authors store directly M^{-1}. The norm $\int_\Omega \rho(\omega_v, \omega_l) cos\theta_l d\omega_l$ of the fitted BRDF is added as a fifth parameter, and all this data requires 80KB. The use of this latter parameter is not specified in the paper. That seems reasonable, but we have to keep in mind that we will use 80KB every time we will want to fit GGX for parameters different from ω_0 and α.

Lighting. After that, the use is quite simple. From a given ω_v and a roughness value α, we get the precomputed transformation matrix. The former formula becomes:

$$I = \int_P L(\omega_l) D(\omega_l) d\omega_l = \int_{P_0} L(\omega_l) D_0(\omega_0) d\omega_0$$

In the simplest case, i.e., constant polygonal light, the L term is constant, so it can be removed from the integral. So

$$I = \int_P L(\omega_l) D(\omega_l) d\omega_l = L \int_{P_0} D_0(\omega_0) d\omega_0 \qquad (2)$$

Now, the way to compute I depends on the chosen original distribution.

Original distribution. The choice of the original distribution allows to create distributions with base shapes. But every distribution doesn't always give good fitting. For instance, GGX is impossible to fit with a uniform distribution. Thus, for fitting this BRDF, the choice of original distribution is really important.

In this paper, the authors use a clamped cosine distribution, because (as as they themselves admit), “that yield to a good approximation to physically based BRDF”. The validation of this assertion is purely visual, it cannot be formally proven.

So, as a clamped cosine distribution is used as original distribution, we need to know how to integrate this kind of distribution over a polygon. Fortunately, Baum et al. [7] already found the closed-form solution (remember it is important in real-time rendering to have constant computation time expression).

Given that, equation 2 is rewritable as $E(P_0)$ that denotes the irradiance over the polygon P_0.

$$\begin{aligned} E(P_0) &= E(p_1, ..., p_n) \\ &= \frac{1}{2\pi} \sum_{i=1}^{n} acos(\langle p_i, p_j \rangle) \left\langle \frac{p_i \times p_j}{\|p_i \times p_j\|}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\rangle \end{aligned}$$

So, we can – for a given view and light directions, a material roughness value and a polygon – compute the irradiance I in constant time (defined by the vertex indices of the polygon).

4 Results

Because the BRDF fitting is based on GGX, the authors compare their results with GGX ray traced.

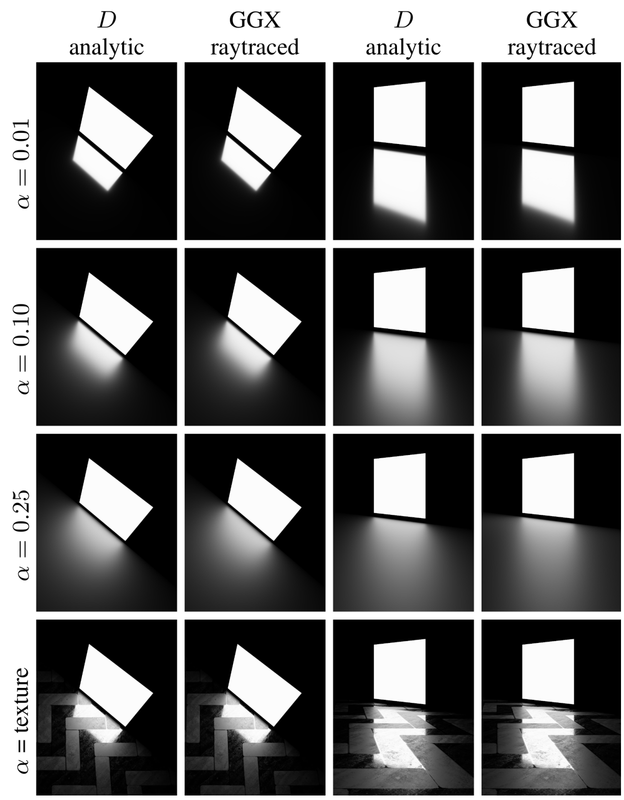

Figure 2: Constant light is several cases

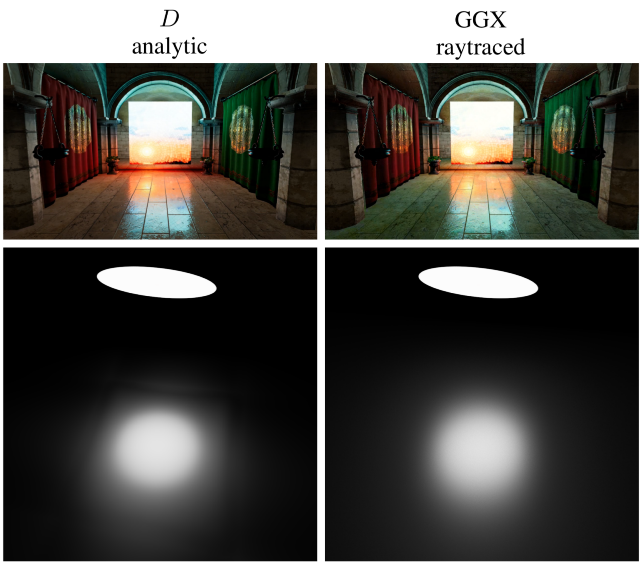

Figure 3: Failure cases

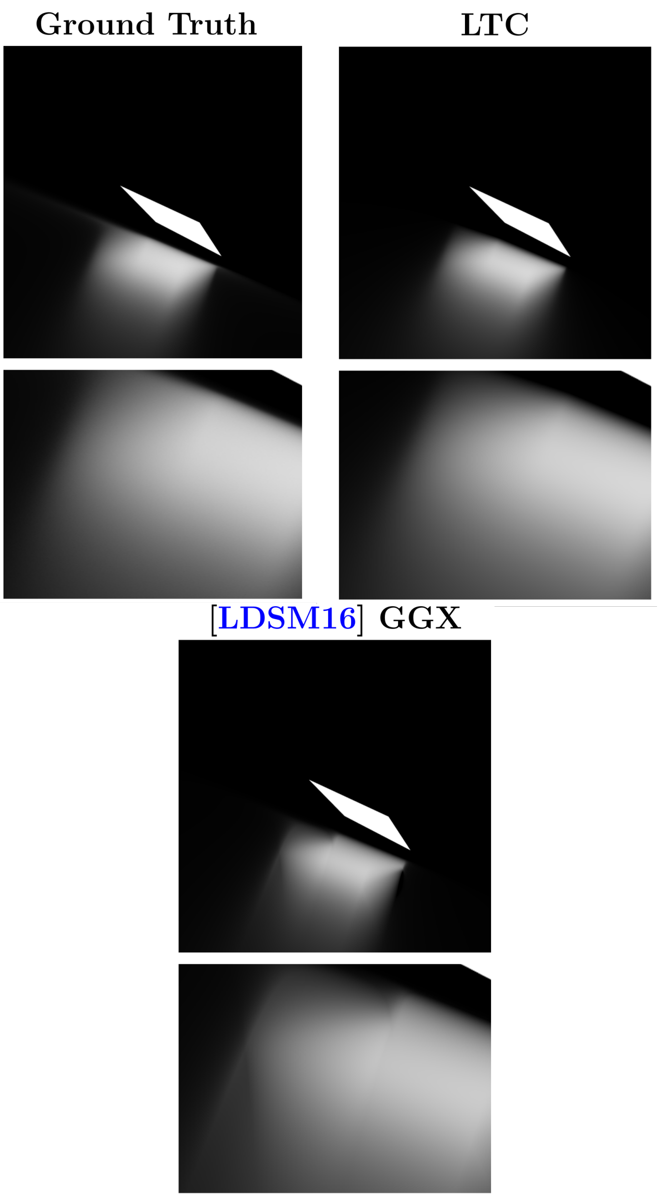

Figure 4: Comparison with [3] method

In figure 2, we can see that results for a rectangular light and simple constraints (different roughnesses) are really close to the "ground truth" (GGX raytraced).

In figure 3, we observe failures, the LTC method gives some slightly different results, but still plausible for the human eye. We are able to detect the differences because the reference is just aside, but without this latter, it will be impossible to find the issues.

By contrast, in figure 4, we clearly see the failures with the method proposed by [3], because some unpleasant artifacts appear. Even if the LTC method lets see some differences, the results are more pleasant to the eye and plausible.

5 Conclusion

Limitations. The main advantage of this method is the relative simplicity to understand it and thus implement it, but the main issue is the required memory to store the precomputed matrices. The degree of liberty is really restricted, we are able to just vary roughness. For each other parameter of the GGX BRDF, we need to compute new matrices and store them. That could be really memory greedy if we use GGX to simulate many materials. This issue doesn't appear in the [3] method, because this one uses functions instead of pre-computed matrices to approximate irradiance value. But this one is really more complex and hard to understand.

Future Work. To improve the method, the authors propose 3 things :

- Allowing integration over other 3D shapes, and not only over polygons. According to them, it should be not difficult if an original distribution can be integrated over a new shape,
- Using other analytic operators, like inner product or convolution,
- Improving BRDF fitting by avoiding nonlinear optimization, which is an expensive operation.

References

[1] Klett, E. Witwe, Detleffsen, C. Peter, *et al.*, *IH Lambert... Photometria sive de mensura et gradibus luminis, colorum et umbrae*. sumptibus viduae Eberhardi Klett, 1760.

[2] J. Arvo, "Applications of irradiance tensors to the simulation of non-lambertian phenomena," in *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 335–342, ACM, 1995.

[3] P. Lecocq, G. Sourimant, and J.-E. Marvie, "Accurate analytic approximations for real-time specular area lighting," in *ACM SIGGRAPH 2015 Talks*, SIGGRAPH '15, (New York, NY, USA), pp. 68:1–68:1, ACM, 2015.

[4] K. Xu, W.-L. Sun, Z. Dong, D.-Y. Zhao, R.-D. Wu, and S.-M. Hu, "Anisotropic spherical gaussians," *ACM Transactions on Graphics*, vol. 32, no. 6, pp. 209:1–209:11, 2013.

[5] K. Xu, Y.-P. Cao, L.-Q. Ma, Z. Dong, R. Wang, and S.-M. Hu, "A practical algorithm for rendering interreflections with all-frequency brdfs," *ACM Trans. Graph.*, vol. 33, pp. 10:1–10:16, Feb. 2014.

[6] J. T. Kajiya, "The rendering equation," in *ACM Siggraph Computer Graphics*, vol. 20, pp. 143–150, ACM, 1986.

[7] D. R. Baum, H. E. Rushmeier, and J. M. Winget, "Improving radiosity solutions through the use of analytically determined form-factors," *SIGGRAPH Comput. Graph.*, vol. 23, pp. 325–334, July 1989.

A Jacobian derivation

In the paper, this equation is not clearly derived. Here is the complete proof.

The formula is:

$$\partial\omega = \partial\omega_0 A \frac{\cos\theta}{r^2}$$

By using the given value $A = \|M\omega_1 \times M\omega_2\|$, $\cos\theta = \left\langle \frac{M\omega_0}{\|M\omega_0\|}, \frac{M\omega_1 \times M\omega_2}{\|M\omega_1 \times M\omega_2\|} \right\rangle$ and $r = \|M\omega_0\|$, we get:

$$\frac{\partial\omega_0}{\partial\omega} = \frac{\|M\omega_1 \times M\omega_2\|}{\|M\omega_0\|^2} \left\langle \frac{M\omega_0}{\|M\omega_0\|}, \frac{M\omega_1 \times M\omega_2}{\|M\omega_1 \times M\omega_2\|} \right\rangle$$

.

Because $M\omega_1 = M\omega_2 = |M| M^{-T}\omega_0$, we can rewrite this latter formula as:

$$\frac{\partial\omega_0}{\partial\omega} = \frac{\||M|M^{-T}\omega_0\|}{\|M\omega_0\|^2} \left\langle \frac{M\omega_0}{\|M\omega_0\|}, \frac{|M|M^{-T}\omega_0}{\||M|M^{-T}\omega_0\|} \right\rangle$$

and after simplification,

$$\frac{\partial\omega_0}{\partial\omega} = \frac{|M| \cdot \|M^{-T}\omega_0\|}{\|M\omega_0\|^2} \left\langle \frac{M\omega_0}{\|M\omega_0\|}, \frac{M^{-T}\omega_0}{\|M^{-T}\omega_0\|} \right\rangle$$

Now,

$$\begin{aligned} \left\langle \frac{M\omega_0}{\|M\omega_0\|}, \frac{M^{-T}\omega_0}{\|M^{-T}\omega_0\|} \right\rangle &= \frac{(M\omega_0)^T M^{-T}\omega_0}{\|M\omega_0\|\|M^{-T}\omega_0\|} \\ = \frac{\omega_0^T M^T M^{-T}\omega_0}{\|M\omega_0\|\|M^{-T}\omega_0\|} &= \frac{\langle\omega_0, \omega_0\rangle}{\|M\omega_0\|\|M^{-T}\omega_0\|} \\ &= \frac{1}{\|M\omega_0\|\|M^{-T}\omega_0\|} \end{aligned}$$

then

$$\frac{\partial\omega_0}{\partial\omega} = \frac{|M|}{\|M\omega_0\|^3}$$

.

and reciprocally

$$\frac{\partial\omega}{\partial\omega_0} = \frac{|M^{-1}|}{\|M^{-1}\omega\|^3}$$