

Efficient direct lighting calculation for area lights with light portals

J. Romeu Huidobro

Supervisors: M. van de Ruit, E. Eismann

EEMCS, Delft University of Technology, The Netherlands

Abstract. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

1 Introduction

Physically-based Monte Carlo light transport algorithms create photorealistic imagery with applications ranging from videogames to architecture visualization. While they can deliver accurate illumination, their most significant limitation is their high computational demands.

Direct lighting is often one of the most significant components, so most rendering algorithms compute direct illumination separately, allowing for more effective light sampling. Direct lighting is computed by explicitly casting shadow rays from the shading point to the light sources. The probability distribution function (PDF) from which the shadow ray directions are sampled can have a high impact on the convergence rate of the scene. For example, consider a scene with a lamp occluded by a lampshade, all of the shadow rays blocked by the lampshade will have no contribution to the final image and are thus wasted sampling effort.

One widely used technique for sampling direct lighting from environment lights is *light portals*. These are artist-specified regions which indicate an opening to the environment light. This information is then used during rendering to prioritize sampling the directions that go through the portal. Our contribution is extending

the idea of light portals to accelerate the sampling of partially occluded area lights by repurposing concepts from shadow volume calculations.

We begin by reviewing Monte Carlo techniques for direct lighting calculations and provide an overview of existing use cases and sampling strategies for light portals (section 2). Next, we expand on the details of our method (section 3) and its implementation into an existing path tracer (section 4). Lastly, we evaluate our portal sampling strategy against light sampling (section 5), and discuss the results (section 6) before concluding and giving final remarks (section 7).

2 Background and related work

The Light Transport Equation The light transport equation (LTE) [1] is a recursive integral over radiometric quantities (reference), which governs our approximation of light transport.

$$L_o(p, \omega_o) = L_e(p, \omega_o) + \int_{\Omega} f(p, \omega_o, \omega_i) L_o(p, \omega_o, \omega_i) |\cos \theta_i| d\omega_i, \quad (1)$$

The equation states that the exitant radiance at point p

in direction ω_o is equal to the emitted radiance at that point and direction plus the portion of the incident light on p that gets reflected towards ω_o .

The direct lighting integral Repeatedly unfolding the recursive call, the LTE can be reformulated into the path-integral formulation [2] which separates the contributions of each path length into separate terms. This reformulation allows us to calculate direct and indirect independently. The direct lighting term is the one we are concerned with evaluating in this paper, and it is given by the integral:

$$\int_{\Omega} f(p, \omega_o, \omega_i) L_d(p, \omega_i) |\cos \theta_i| d\omega_i, \quad (2)$$

where L_d is the *direct* radiance from direction ω_i , f is the Bidirectional Scattering Function (BSDF) and the cosine term accounts for Lamberts law.

Monte Carlo Integration This integral cannot be solved analytically in general, thus we use *Monte Carlo integration* as a means to numerically estimate it. To compute this estimate, we need to sample N directions ω_i and apply the Monte Carlo estimator [3]:

$$\frac{1}{N} \sum_{i=1}^N \frac{f(p, \omega_o, \omega_i) L_d(p, \omega_i) |\cos \theta_i|}{p(\omega_i)} \quad (3)$$

To reduce variance, we can use *importance sampling* to choose the directions ω_i from a probability density function PDF similar to the integrand. However, when deciding on a distribution to sample from, its clarifying that we are *not* after the distribution providing the lowest variance for a given number of samples, but instead the distribution providing the lowest variance for a given amount of time [4], thus an effective sampling strategy may be worse than an ineffective one if the latter can be sampled more efficiently.

The most common sampling technique is to choose a light to be sampled for based on a precomputed PDF over the light sources in the scene [4], weighted based on one or more heuristics such as light power (emitter sampling citations), and then using multiple importance sampling [2] to draw samples from the PDF of the BSDF and the light’s solid angle, which are then combined according to the contribution of each sample. This approach, however, ignores the visibility of the light source when drawing the samples, which can lead to high variance in scenes where the light is partially occluded, producing many blocked shadow rays.

Light portals *Light portals* are artist-specified regions in the scene which indicate an opening to an environment light, such as a window in an interior scene. During rendering, the portal is then used to focus the environment light sampling to only those directions visible through the portal. They are highly effective for accelerating convergence in many scenes and are implemented in several production renderers such as Cycles [5] and Renderman [6].

The portal area is typically sampled uniformly, although more advanced portal sampling strategies exist [7][8]. Nonetheless, these sampling strategies are designed for sampling *environment lights*, and several challenges and opportunities are presented when applying the same technique to area lights. Unlike area lights, environment lights extend across the whole scene, so any direction sampled through the portal will also be directed to the environment. For this reason, existing portal sampling strategies are unsuitable for area lights.

3 Method

We now present our method. We begin by explaining how the portal is represented and incorporated into the light distribution for a scene with a single light and a single portal. Next, we give an effective yet slow sampling strategy and show how it can be made more efficient by repurposing ideas from shadow calculations. Lastly, we generalize the provided method to multiple portals and multiple lights.

3.1 Sampling a single light portal

Consider a simple scene containing a single light source covered by a lampshade (Figure 1). The core idea behind our method is to have the artist include the geometry of the opening to the light source as a “portal” in the scene description. This portal is then incorporated into the light distribution of the scene. When constructing the light distribution, the portal is prioritized for points in front of it and never sampled for points behind it.

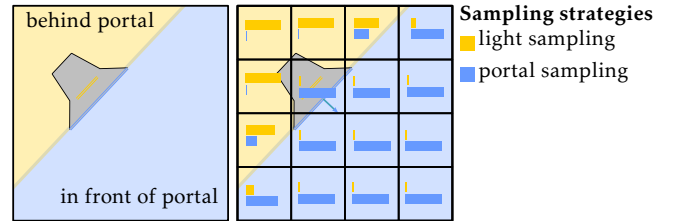


Figure 1: Simple scene (left) showing the region behind and in front of the portal. As well as an illustration of the light distribution for the scene

3.2 Sampling the portal

In (background), we explained why uniformly sampling the portal area is not a good strategy when using light portals with area lights. If we want to only sample directions which go through the portal and are facing the light, the region that needs to be sampled is the intersection of the solid angles subtended by the light source and the light portal.

This region can be sampled by projecting the light source from the perspective of the shading point onto the portal and then clipping the projected region to the portal bounds [9].

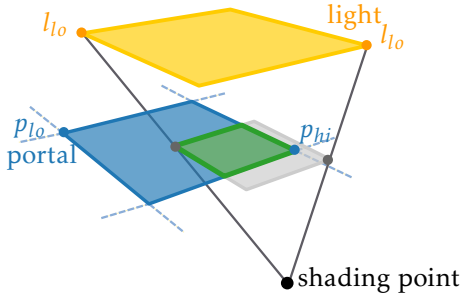


Figure 2: Projection sampling. The light (yellow) is projected onto the portal-plane (blue) and clipped to the portal bounds. Samples are then drawn from the clipped region (green).

This sampling strategy is effective per sample but much slower than light sampling or portal sampling. In the next section, we show how some additional geometry allows us to restrict the use of this sampling strategy to the points where it provides the most significant advantage over more efficient alternatives.

3.3 Thinking in terms of shadows

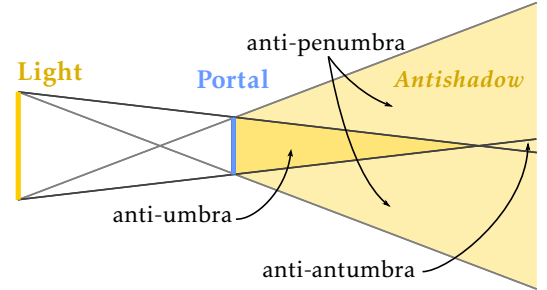
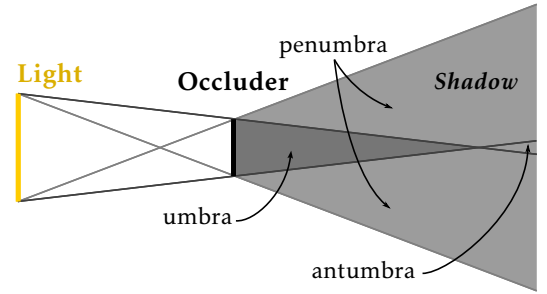
An interesting way to think about light portals is to consider the shadow that the portal would cast if we were to replace it with an occluder. The area in front of the occluder can then be separated into four regions. The *umbra*, *penumbra* and *antumbra* of the shadow, as well as the region out of shadow [10]. These regions are defined as:

- The **umbra** (hard shadow) is the portion of the shadow where the occluder blocks all direct paths to the light source.
- The **penumbra** (soft shadow) is the portion of the shadow where some but not all direct paths to the light are occluded.
- If the occluder is smaller than the light, the umbra will converge, forming the **antumbra**, which is the portion of the penumbra where the occluder is directly in front of the light.

- The **region outside of shadow** is the area where none of the paths to the light is occluded.

Nonetheless, the portal is not an occluder, but instead precisely the opposite, as it indicates an *opening* to the light. Thus, these same regions take on a new meaning when we treat them from the portal's perspective. The portal-equivalent regions are:

- The **anti-umbra** is the region where all direct paths to the light go through the portal.
- The **anti-penumbra** is the region where some but not all paths to the light go through the portal.
- The **anti-antumbra** is the region where the portal is directly in front of the light but does not fully cover it.
- The **region outside of the antishadow** is the area where none of the paths from the shading point to the light to the light pass through the portal



(a) 2D representation of shadow and antishadow regions.

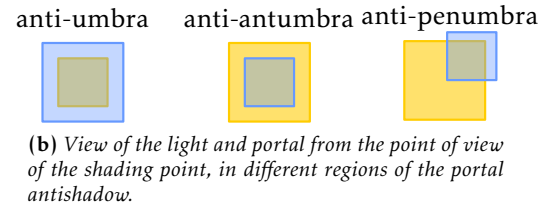


Figure 3: Two-dimensional representation of the regions of a shadow. If we replace the occluder with a light portal, the same geometric volumes take different roles.

Returning to our problem of sampling the light going through a portal, having an idea of which region of the antishadow the shading point is in can be used to choose a sampling strategy. Firstly, if the shading point is **outside of the antishadow**, there is no need to do any sam-

pling whatsoever, as there are no direct paths through the portal that reach the light. Thus if the shading point is in this region, we can simply return zero without any computation.

If the shading point is in the **anti-umbra**, all direct paths to the light go through the portal, so visibility is guaranteed. Thus, the optimal sampling strategy is to sample the light source itself. Conversely, when the shading point is in the **anti-antumbra**, all of the paths through the portal hit the light, making portal sampling the optimal strategy. Figure (antumbra b) gives a good intuition on why this works.

Lastly, points in the **anti-penumbra** are more complex, as some but not all direct paths to the light go through the portal. In this case, we resort to using the effective, but slow projection sampling method explained in [subsection 3.2](#).

Returning to the simple scene from figure [Figure 1](#), we can now partition the space further by including the antishadow volumes. Now, we add the sampling strategies discussed above to the light distribution, weighing the PDF of each strategy based on which region of the antishadow the shading point is contained in.

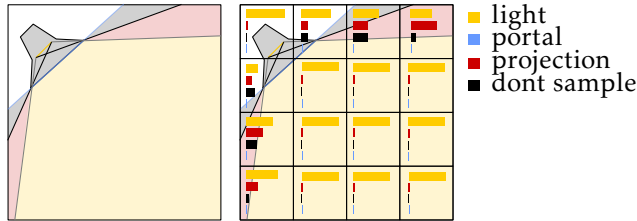


Figure 4: Simple scene employing antishadow volumes to weight the choice of sampling strategy depending on the location of the shading point.

In doing this, we restrict the slow projection sampling technique to only the points in the anti-penumbra and use faster strategies for shading points where there is no additional benefit gained by projecting.

3.4 Computing the antishadow regions

In ??, we explained the concept of a light portal "antishadow" and how information about which region of the antishadow the shading point is in can be used to sample the portal efficiently. This section discusses how these antishadow volumes are computed and used to weigh sapling strategies.

For polygonal objects, the shape of the umbra and penumbra regions can be embedded in a discontinuity mesh constructed from the edges and vertices of the light, and the portal [11]. However, we implement a simplified version of this algorithm limited to rectangular aligned light and portal.

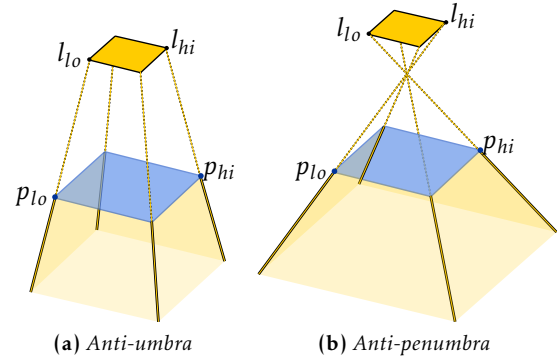


Figure 5: Computing the anti-umbra and anti-penumbra for aligned rectangles. The anti-antumbra can be derived from the Anti-umbra.

The anti-umbra is the frustum produced by connecting the adjacent vertices from the portal and the light (5a). Likewise, the anti-penumbra is the frustum produced by connecting the *opposite* vertices from the light to the portal (5b). The anti-antumbra does not need to be explicitly computed, as it is defined by the same planes that make up the anti-umbra, but with their normals inverted.

To check which region a point is in, we can check if the point is on the right side of all of the planes defining the region. This process is quite fast as, in the worst case, it requires only eight dot products and comparisons. However, a slower implementation would be equally suitable since the light distribution is precomputed.

3.5 Multiple portals and multiple lights

Finally, generalizing to multiple portals and multiple lights is relatively straightforward. Since we incorporate the portals into the light distribution, we can assign the probability density of each portal based on

will write this section later as I want to try some quick heuristics such as portal area, portal distance etc. to set the pdfs of each portal

4 Implementation

still some work to do in this regard unfortunately

5 Results

should I evaluate with direct lighting integrator or path? direct lighting would show the results more clearly, but ideally I show results for actual rendering right? Also could do both of course, what do you think?

We evaluate the method presented for a scene containing a variety of lights each with their corresponding portals. We compare portal sampling, using antishadow volumes to set the portal sampling distributions. The reference image was produced using an unbiased unidirectional path tracer, using light sampling, and x samples per pixel. We then compare the mean squared error over time of portal sampling and light sampling.

6 Discussion

Incomplete due to missing results.

Things to potentially include (depending on if I implement the feature or not):

- **Limitation:** Requires additional artist input so not great, but mention that since standard light portals are already included in most pipelines it should be feasible to implement it.
- **Limitation:** Implementation presented is limited to AAplanes, give guidance on generalizing

7 Conclusion

Incomplete since still on results

References

- [1] James T Kajiya. “THE RENDERING EQUATION”. In: 20.4 (1986), p. 8.
- [2] Eric Veach. “ROBUST MONTE CARLO METHODS FOR LIGHT TRANSPORT SIMULATION”. In: (), p. 432.
- [3] Erik Lafortune. *Monte Carlo Techniques and Optimisations for Physically Based Rendering*. Dagstuhl Seminar on Rendering. Wadern, Germany, June 1996.
- [4] Peter Shirley, Changyaw Wang, and Kurt Zimmerman. “Monte Carlo Techniques for Direct Lighting Calculations”. In: *ACM Transactions on Graphics* 15.1 (Jan. 1996), pp. 1–36. issn: 0730-0301, 1557-7368. doi: [10.1145/226150.226151](https://doi.org/10.1145/226150.226151). URL: <https://dl.acm.org/doi/10.1145/226150.226151> (visited on 05/11/2022).
- [5] *Light Settings — Blender Manual*. URL: https://docs.blender.org/manual/en/latest/render/cycles/light_settings.html?highlight=light%20portals (visited on 05/06/2022).
- [6] *PxrPortalLight*. Renderman Documentation. URL: <https://rmanwiki.pixar.com/display/REN24/PxrPortalLight> (visited on 05/06/2022).
- [7] Shinji Ogaki. “Generalized Light Portals”. In: *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 3.2 (Aug. 26, 2020), pp. 1–19. issn: 2577-6193. doi: [10.1145/3406176](https://doi.org/10.1145/3406176). URL: <https://dl.acm.org/doi/10.1145/3406176> (visited on 04/23/2022).
- [8] Benedikt Bitterli, Jan Novák, and Wojciech Jarosz. “Portal-Masked Environment Map Sampling”. In: *Computer Graphics Forum* 34.4 (July 2015), pp. 13–19. issn: 0167-7055, 1467-8659. doi: [10.1111/cgf.12674](https://doi.org/10.1111/cgf.12674). URL: <https://onlinelibrary.wiley.com/doi/10.1111/cgf.12674> (visited on 05/06/2022).
- [9] Ivan E. Sutherland and Gary W. Hodgman. “Reentrant Polygon Clipping”. In: *Communications of the ACM* 17.1 (Jan. 1974), pp. 32–42. issn: 0001-0782, 1557-7317. doi: [10.1145/360767.360802](https://doi.org/10.1145/360767.360802). URL: <https://dl.acm.org/doi/10.1145/360767.360802> (visited on 05/31/2022).
- [10] J. -M. Hasenfratz et al. “A Survey of Real-time Soft Shadows Algorithms: Real-time Soft Shadows”. In: *Computer Graphics Forum* 22.4 (Dec. 2003), pp. 753–774. issn: 01677055. doi: [10.1111/j.1467-8659.2003.00722.x](https://doi.org/10.1111/j.1467-8659.2003.00722.x). URL: <https://onlinelibrary.wiley.com/doi/10.1111/j.1467-8659.2003.00722.x> (visited on 06/08/2022).
- [11] George Drettakis and Eugene Fiume. “A Fast Shadow Algorithm for Area Light Sources Using Backprojection”. In: *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '94*. The 21st Annual Conference. Not Known: ACM Press, 1994, pp. 223–230. isbn: 978-0-89791-667-7. doi: [10.1145/192161.192207](https://doi.org/10.1145/192161.192207). URL: <http://portal.acm.org/citation.cfm?doid=192161.192207> (visited on 06/09/2022).



Figure 6: *Image is a bit sketchy (and data is fake), but feedback on the composition might be useful* We compare the convergence rates for light sampling (Light) and portal sampling (Portal). The listed errors are for the highlighted image insets. As seen, portal sampling is the most robust method, regardless of the size ratio between the light and the portal