A 5D Tree to Reduce the Variance of Monte Carlo Ray Tracing

Eric P. Lafortune and Yves D. Willems

Paper for the 6th Eurographics Workshop on Rendering

Please send any correspondence to:
Eric Lafortune
Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200A
B-3001 Heverlee
BELGIUM
Telephone: ++32 - 16 - 32 70 94
Fax: ++32 - 16 - 32 79 96
E-mail: Eric.Lafortune@cs.kuleuven.ac.be

A 5D Tree to Reduce the Variance of Monte Carlo Ray Tracing

Eric P. Lafortune and Yves D. Willems

Department of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200A, 3001 Heverlee, Belgium
Eric.Lafortune@cs.kuleuven.ac.be

Abstract: In this paper we present a 5D tree structure to cache illumination information gained during Monte Carlo ray tracing. The structure is elegant and simple to use. It is adaptive and makes abstraction of the complexity of the input scene automatically.
We then show how the information in this structure can be used to reduce the variance of the Monte Carlo process. Unlike earlier approaches the techniques presented here do not introduce a bias in the results.

1 Introduction

Monte Carlo ray tracing is widely acknowledged as a general but slow technique for faithfully rendering scenes with global illumination effects. Its stochastic nature entails an uncertainty on the results, which is expressed by the variance of the process. In ray tracing it is visible as uncorrelated noise in the rendered images. Monte Carlo techniques typically reduce the variance by averaging the results from a large number of primary samples. The reduction of the variance is proportional to the number of samples, which means that the standard deviation is only reduced proportionally to the square root of the number of samples.

Most research on Monte Carlo techniques is therefore directed towards finding more efficient sampling schemes which yield a lower variance for the primary estimator. General optimisations [1, 2] can also be applied to the global illumination problem, on the basis of a concise mathematical description [3, 4, 5, 6, 7, 8, 9, 10].

Standard Monte Carlo ray tracing algorithms trace a large number of rays, all independently from one another. Only the eventual radiance values are averaged per pixel and stored. It seems wasteful to throw away all the information gained while tracing rays through the scene. Ward presented what seems the most successful caching scheme to date [11, 12]. He separates the calculations of indirect diffuse reflections from the calculations of the other components. Diffuse illumination is computed separately and cached in an octree. The cached values are interpolated whenever diffuse illumination values are required in their neighbourhoods. In this respect his technique is similar to various two-pass methods that work with an explicit discretisation of the surfaces in the scene; diffuse reflections are interpolated, extrapolated or averaged in some way. The correctness of the results largely depends on the granularity of the discretisations. From a Monte Carlo point of view the results are biased as the expected values of the results of the simulation are no longer the correct values.

In this contribution we propose techniques that do not introduce a bias. The first section presents a tree structure that subdivides the 5D space of positions and directions hierarchically. Computed values for incoming radiance are stored in this tree. The following section explains how these cached values can be used to improve importance sampling and control variates. We then present some results obtained using these optimisations.

2 The 5D tree

Various data structures to store information about the illumination in a scene have been proposed in the literature. Boundary element methods such as the radiosity method store information for each of the elements of the discretised surfaces. The information is usually restricted to the diffuse component of the illumination although extensions to store directional information as well have been proposed, e.g. by discretising the hemisphere [13] or by using spherical harmonics [14]. Ward [11, 12] uses an octree that is independent from the surfaces to store diffuse illumination along with information about the gradients. Finite element methods for rendering participating media discretise the space into regular grids [15, 16, 17] or hierarchical structures [18], mostly without storing directional information.

We propose a 5D tree which is a straightforward extension of the 1D binary tree, the 2D quadtree, the 3D octree, etc. The dimensions are the three regular spatial dimensions and two additional angular dimensions. For convenience a 5D point corresponding to a point (x, y, z) and a direction (θ, ϕ) is represented in coordinates that are normalised to lie between 0 and 1:

$$(x_n, y_n, z_n, s_n, t_n)$$

where:

$$\begin{aligned}
x_n &= (x - x_{min})/(x_{max} - x_{min}), \\
y_n &= (y - y_{min})/(y_{max} - y_{min}), \\
z_n &= (z - z_{min})/(z_{max} - z_{min}), \\
s_n &= (cos\,\theta + 1)/2, \\
t_n &= \phi/(2\pi)
\end{aligned}$$

This parametrisation ensures a uniform subdivision: equally sized 5D volumes at any 5D position correspond to equal 3D volumes and equal spatial angles.

The branching factor of the tree is $2^5 = 32$. It may seem that this would give rise to an explosive growth of the storage requirements, but it is not a problem in practice as only nodes that are effectively needed to store values are created.

Which values are stored in the tree then? The ray tracing algorithm traces random walks through the scene. The process starts with a primary ray from the eye point through a pixel. At the closest intersection point the direct illumination is usually computed by sampling the light sources. Then a random direction is sampled for the reflected ray. The incoming radiance for this ray is estimated by repeating the tracing process recursively. These estimated radiance values for all the intersection points and the sampled directions are cached in the tree, in the nodes determined by their positions (Fig. 1). The values are averaged in the appropriate nodes at each level of the tree. Whenever the number of estimates averaged in a node exceeds a given maximum threshold the node is split; a new node is created if necessary. The building of the tree is in a sense importance-driven; more detailed representations are constructed in regions where more rays are traced.

The best averaged estimate for any point and direction can be retrieved from the tree at any time by descending to the lowest node corresponding to the given coordinates. We impose a minimum threshold on the number of averaged values for the estimate to be reliable. While the representation is not as accurate as representations that are specifically linked to each of the surfaces we now avoid any meshing problems. What is more, the tree automatically makes abstraction of the complexity of the scene. Large groups of small geometric and optical details (e.g. a desk cluttered with books, pencils and paper clips) are treated as a whole by storing only averaged incoming radiance values.

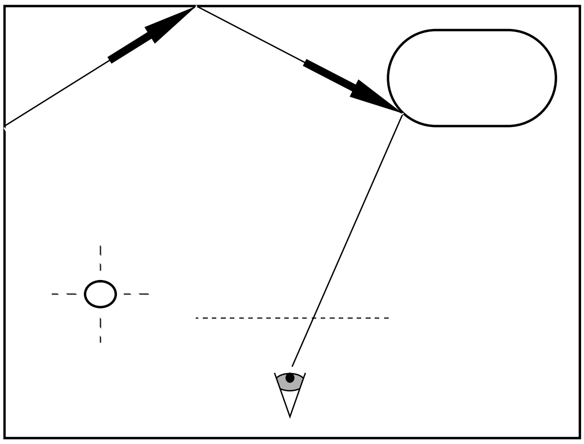

Fig. 1. During the Monte Carlo ray tracing process indirect illumination is estimated by casting reflected rays recursively. The computed estimates for incoming radiance (indicated by the arrows) are stored in the 5D tree.

3 Reduction of the Variance

The information stored in the 5D tree has to be put to good use. The first application is in adaptive importance sampling. This optimisation attempts to send more rays to bright regions in the scene. The second application is in constructing better control variates, approximations of the functions to be integrated that also help to reduce the variance.

3.1 Adaptive importance sampling

Importance sampling is an optimisation over uniform sampling [1, 2]. Rather than taking the random samples uniformly over the integration domain they are concentrated in regions where the integrand is large. For example, consider a one-dimensional function $f(x)$ that is to be integrated over the interval $[0, 1]$:

$$I = \int_0^1 f(x)dx = \int_0^1 p(x)f(x)/p(x)dx$$

If a random sample x is taken according to a uniform probability density function (PDF) the primary estimate for the integral is $f(x)$. If x is sampled according to the PDF $p(x)$ the estimate becomes $f(x)/p(x)$. The variance can be shown to go to 0 when the PDF is chosen proportional to the integrand. The function $p(x)$ must be normalised, and its cumulative function $P(x)$ has to be invertible. One can see that importance sampling in one parameter space corresponds to uniform sampling in another parameter space, by applying the transformation $x = P^{-1}(t)$, which yields:

$$I = \int_0^1 p(x)f(x)/p(x)dx = \int_0^1 f(P^{-1}(t))/p(P^{-1}(t))dt$$

In Monte Carlo ray tracing importance sampling is commonly used when sampling directions for reflected rays, to estimate indirect illumination [5, 7, 8]. The PDF, which is two-dimensional now, is usually chosen proportional to the bidirectional reflection distribution function (BRDF) times a cosine factor. This way more rays are sent in directions for which the incoming illumination is important. Formally, the outgoing radiance (or *surface radiance*) L_o is the integral over the incoming hemisphere Ω_i of the BRDF f_r times the incoming radiance (or *field radiance*) L_i times a cosine factor:

$$L_o(\Theta_o) = \int_{\Omega_i} f_r(\Theta_i, \Theta_0) L_i(\Theta_i) \left|cos\, \theta_i\right| d\omega_i = \int_{\Omega_i} p(\Theta_i) L_i(\Theta_i) d\omega_i$$

Using the BRDF times the cosine factor as a subcritical PDF $p(\Theta_i)$ for sampling the direction Θ_i implicitly entails a transformation to a parameter space $(\xi_1, \xi_2) \in [0,1] \times [0,1]$ where ξ_1 and ξ_2 are sampled uniformly to obtain a $\Theta_i(\xi_1, \xi_2)$ [19]. The expression then becomes:

$$L_o(\Theta_o) = \int_0^1 \int_0^1 L_i(\Theta_i) d\xi_1 d\xi_2$$

This type of importance sampling already yields a major gain for specular surfaces. Most rays are reflected inside the specular cone rather than uniformly over the hemisphere. At diffuse surfaces however rays are only reflected according to a cosine distribution. A large number of samples are usually required to get a reliable estimate.

Additional information about the integrand may help to sample directions more intelligently. In our case the 5D tree structure contains information about the incoming radiance. In order not to loose the advantages of the importance sampling proportional to the BRDF we propose an additional importance sampling scheme in the (ξ_1, ξ_2) parameter space. For each reflection we first resample the stored incoming radiance values on a regular grid in this parameter space. We then construct a piece-wise constant PDF from this discrete representation by normalising the sampled values in the grid. A random pair (ξ_1, ξ_2) is now sampled on the basis of this PDF. As before the values map to a direction for the reflected ray. Figure 2 shows the 2D equivalent of this process schematically. Formally, if $\tilde{L}_i$ is the piece-wise constant approximation for the incoming radiance, with normalisation factor N:

$$L_o(\Theta_o) = \int_0^1 \int_0^1 \frac{\tilde{L}_i(\Theta_i)}{N} \frac{L_i(\Theta_i)}{\tilde{L}_i(\Theta_i)/N} d\xi_1 d\xi_2 = \int_0^1 \int_0^1 p'(\Theta_i) \frac{L_i(\Theta_i)}{p'(\Theta_i)} d\xi_1 d\xi_2$$

The net result is that we sample according to a PDF approximating the entire integrand of the original rendering equation. Both the BRDF with the cosine factor and the approximation of the incoming radiance are taken into account. As with any sampling process the selection of the PDF only influences the variance of the estimate and not its expected value. The expected value therefore remains correct, independent of the quality of the approximations.

3.2 Improved control variates

Control variates are another optimisation for Monte Carlo methods [1, 2, 20]. Consider the one-dimensional integral of $f(x)$ over $[0,1]$ again. Suppose one knows a function $g(x)$ which approximates $f(x)$ and which can be integrated analytically. $g(x)$ can then be used as a control variate for $f(x)$:

$$I = \int_0^1 f(x)dx = \int_0^1 [f(x) - g(x)]dx + \int_0^1 g(x)dx = \int_0^1 [f(x) - g(x)]dx + J$$

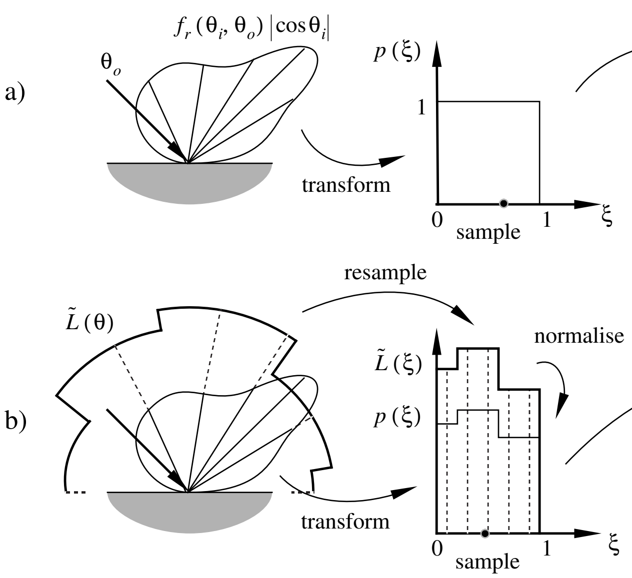

Fig. 2. (a) Importance sampling as it is commonly used with path tracing. For a given direction θ_o a reflected direction θ_i has to be sampled according to a PDF that is proportional to the BRDF times the cosine factor. This is implemented by taking a uniform random sample ξ in a parameter space for which the transformation can be computed analytically, and then transforming that sample to a direction θ_i.
(b) Adaptive importance sampling as presented here also uses the estimates for incoming radiance that are stored in the 5D tree. The estimates are resampled at regular intervals in the ξ parameter space. The resulting piece-wise constant approximation is normalised to obtain a PDF. The sample ξ is now taken according to this PDF and again transformed to a direction θ_i. As a result reflections will be sampled according to the BRDF but also preferentially in bright and thus important directions.

For a uniform sample x the primary estimate becomes $[f(x) - g(x)] + J$. If the function $g(x)$ approximates the function $f(x)$ closely enough this estimate will have a lower variance than the original estimate.

In an earlier paper we have already applied control variates to the rendering equation [21]. The incoming radiance could be approximated by a constant ambient radiance yielding a simple control variate. Thanks to the 5D tree we now have more detailed information about the incoming radiance. These estimates can be expected to be more effective. Formally, in the (ξ_1, ξ_2) parameter space where we have constructed the piece-wise constant approximation $\tilde{L}_i$:

$$L_o(\Theta_o) = \int_0^1 \int_0^1 (L_i(\Theta_i) - \tilde{L}_i(\Theta_i)) d\xi_1 d\xi_2 + \int_0^1 \int_0^1 \tilde{L}_i(\Theta_i) d\xi_1 d\xi_2$$

The latter integral of the piece-wise constant approximation can easily be computed analytically. In fact, it comes for free with the importance sampling explained in the previous paragraph as it is the normalisation factor of the PDF.

4 Results

We have tested the proposed optimisations on an implementation of a path tracing algorithm which is already optimised by means of multi-dimensional stratified sampling and classical importance sampling of the BRDFs. Russian roulette is used to terminate a random walk. The BRDF model is a modified Phong model which is both reciprocal and energy-conserving [19]. For BRDF models such as this one which consist of a sum of a diffuse and a specular component the proposed importance sampling technique can be easily extended to select between specular and diffuse reflection in a more effective way.

Figure 3a shows a test scene which is mostly indirectly illuminated by lamp shades against the walls. This is a particularly difficult scene for stochastic ray tracing. Figure 4 shows images rendered without and with using the 5D tree respectively, at 100×100 pixels. Only 50 samples per pixel were used in order to clearly demonstrate the differences. Figure 5 shows more precisely how the optimisations reduce the RMS error (compared to a reference image). The reduction varies between 20 and 25%.

Figure 3b shows the same scene, now illuminated by light boxes against the ceiling. For this scene we have not used shadow rays to sample direct illumination separately, but rather added lighting contributions from the reflected rays only. This approach may be interesting for large or complex luminaires which cannot be sampled efficiently. The effect of the optimisations presented here will increase because they improve on the reflected rays. In this case they yield a reduction of the RMS error of about 30%.

a) b)

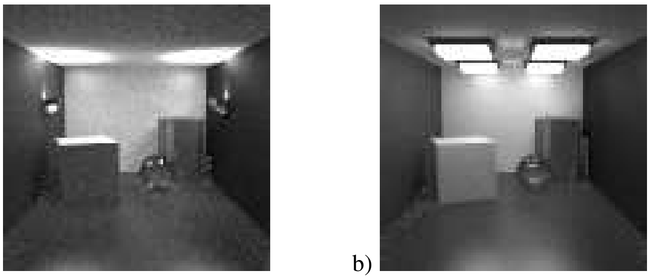

Fig. 3. The test scenes with different types of luminaires.

In the tests we have set the threshold for splitting nodes of the 5D tree proportional to the total number of primary samples. This resulted in about 11.000 nodes for the first test quickly levelling to about 86.000 nodes for the latter tests (requiring between 2 MBytes and 14 MBytes respectively). However, additional experiments have shown that the improvements are only affected slightly if the number of nodes is kept small. Figure 6 gives an impression of the radiance estimates stored in the 5D tree after rendering an image with 50 samples per pixel.

If the image is computed pixel per pixel the variance for the first pixels will be larger than the variance for the last pixels, as the data structure will have been refined in the meantime. Scanning over the image repeatedly, taking a single sample per pixel at a time removes this effect. Experiments have shown that the overall improvement is small though.

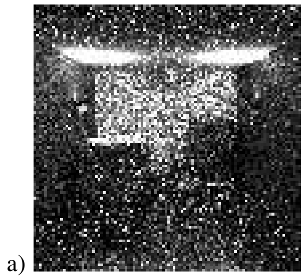

Fig. 4. The first test scene rendered with 50 samples per pixel, (a) without and (b) with the optimisation respectively.

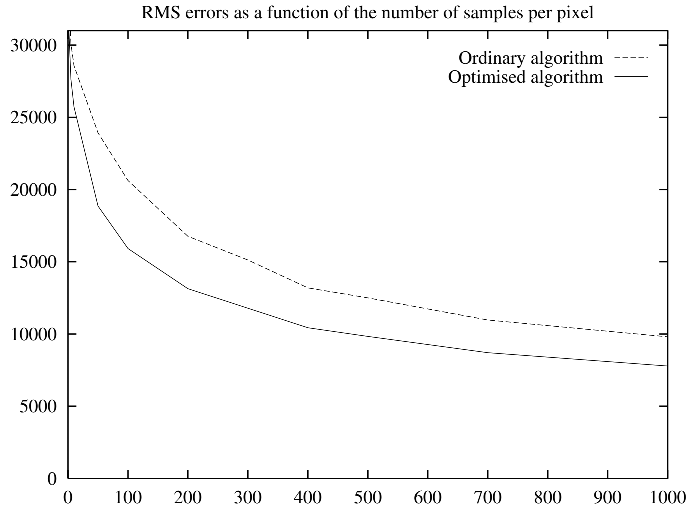

Fig. 5. The RMS error as a function of the number of samples per pixel N, for the first test scene. It decreases as $1/\sqrt{N}$, but for the optimised algorithm it is 20 to 25% smaller.

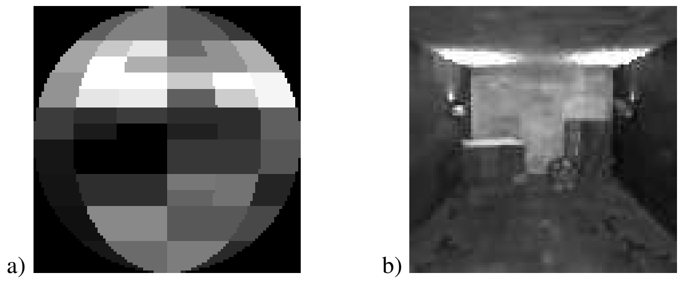

Fig. 6. An impression of the 5D tree structure after rendering the actual images with 50 samples per pixel. (a) A fish eye view of the radiance values stored at a point in the middle of the white wall. This image shows the directional subdivision of the 5D tree. (b) An image rendered with the stored radiance values integrated locally at each first intersection point. This image gives an idea of the spatial subdivision of the 5D tree. Note that the glossy reflections are a result of the directional information captured in the tree structure.

5 Conclusions

We have presented a 5D data structure to store information about incoming radiance for illumination simulations. The tree is independent from the surfaces in the scene and therefore somewhat less accurate than boundary element subdivisions of the surfaces. However, this spatial approach also has some distinct advantages:

– The structure is uniform and simple to maintain. All meshing problems inherent to boundary element methods are avoided.
– The tree is adaptive; only spatial and directional regions that are important for the simulation are refined. Empty spaces and uninteresting directions are not considered because no values are ever stored there.
– The top-down refinement automatically makes abstraction of the complexity of the scene. Because incoming radiance values are stored large sets of small objects can be grouped in an effective way, even if they have complex geometries or optical properties.
– The memory requirements are simple to control. More available memory allows for more detailed representations of the radiance function and faster convergence.
– The structure also seems to lend itself perfectly for use with participating media since it can store radiance values in the 3D space and not only at surfaces.
– We also intend to use it to store potential (or *importance*) for light tracing and bidirectional path tracing algorithms.

The 5D tree has the same kind of attractiveness that Monte Carlo methods have: while it is not overly efficient it is extremely general in its application.

We have then shown how the information stored in the 5D tree can be used to improve the efficiency of Monte Carlo ray tracing. More specifically indirect illumination can be computed more accurately thanks to two optimisations:

– An adaptive importance sampling scheme can send more reflected rays in bright directions.
– Better control variates can be constructed for the rendering equation on the basis of the information about incoming radiance.

The improved importance sampling scheme and control variates reduce the variance of the simulation. Practically, a lower RMS error of the image is obtained for the same number of samples per pixel. Moreover, they do not introduce a bias in the result, unlike most techniques that cache illumination values. This is of some theoretical importance and may be vital in practice when computing reference images to verify other techniques for instance. It comes at a cost however. Other methods like in Ward's system will produce better images if some bias is no object.

The computational overhead of the proposed optimisations is relatively large but it does not depend on the complexity of the scene. While the extra work does not pay off for simple scenes, it will become worthwhile as the complexity increases and tracing rays inevitably becomes increasingly expensive.

Compared to bidirectional path tracing [22, 8, 10, 23] this technique may offer an advantage if the sources of indirect illumination are relatively large and bright, or if there is a large number of light sources. In those cases seeking light starting from the eye will be more effective than distributing light starting from a random light source.

Elsewhere in these proceedings Dutré *et al.* [24] and Jensen [25] present similar adaptive importance sampling schemes, without the control variate optimisation. Dutré *et al.* apply it to light tracing, which is the dual algorithm of path tracing. They store approximated potential functions at discrete patches in the scene and sample the hemisphere without taking the BRDF into account. Jensen applies it to path tracing in virtually the same way as presented here. However, he uses information that is stored as a fixed set of reflected photons from a first light tracing pass. Our approach has the advantage that it does not require any extra rays and that computational effort and memory are only spent on regions of the scene that are important for the image.

6 Acknowledgements

The first author would like to acknowledge the financial support by a grant from the Flemish 'Institute for the promotion of Scientific and Technological Research in the Industry' (IWT #944045).

References

1. M. Kalos and P. Whitlock, *Monte Carlo Methods*. Wiley & Sons, 1986.
2. J. Hammersly and D. Handscomb, *Monte Carlo Methods*. Chapman and Hall, 1964.
3. M. Lee, R. Redner, and S. Uselton, "Statistically optimized sampling for distributed ray tracing," *Computer Graphics*, vol. 19, pp. 61–67, July 1985.
4. P. Shirley, *Physically Based Lighting Calculations for Computer Graphics*. PhD thesis, University of Illinois, Nov. 1990.
5. B. Lange, "The simulation of radiant light transfer with stochastic ray-tracing," in *Proceedings of the Second Eurographics Workshop on Rendering*, (Barcelona, Spain), May 1991.
6. G. Ward, "Adaptive shadow testing for ray tracing," in *Proceedings of the Second Eurographics Workshop on Rendering*, (Barcelona, Spain), May 1991.
7. S. Pattanaik, *Computational Methods for Global Illumination and Visualisation of Complex 3D Environments*. PhD thesis, Birla Institute of Technology & Science, Pilani, India, Feb. 1993.
8. E. Lafortune and Y. Willems, "A theoretical framework for physically based rendering," *Computer Graphics Forum*, vol. 13, pp. 97–107, June 1994.
9. P. Shirley, C. Wang, and K. Zimmerman, "Monte carlo techniques for direct lighting calculations," *ACM Transactions on Graphics*, to appear.
10. E. Lafortune and Y. Willems, "Reducing the number of shadow rays in bidirectional path tracing," in *Proceedings of WSCG 95*, (Pilsen, Czech Republic), pp. 384–392, Feb. 1995.
11. G. Ward, F. Rubinstein, and R. Clear, "A ray tracing solution for diffuse interreflection," *Computer Graphics*, vol. 22, pp. 85–92, July 1988.
12. G. Ward and P. Heckbert, "Irradiance gradients," in *Proceedings of the Third Eurographics Workshop on Rendering*, (Bristol, UK), pp. 85–98, May 1992.
13. D. Immel, M. Cohen, and D. Greenberg, "A radiosity method for non-diffuse environments," *Computer Graphics*, vol. 20, pp. 133–142, Aug. 1986.
14. F. Sillion, J. Arvo, S. Westin, and D. Greenberg, "A global illumination solution for general reflectance distributions," *Computer Graphics*, vol. 25, pp. 187–196, July 1991.

15. C. Patmore, “Illumination of dense foliage models,” in *Proceedings of the Fourth Eurographics Workshop on Rendering*, (Paris, France), pp. 63–71, June 1993.
16. E. Languénou, K. Bouatouch, and M. Chelle, “Global illumination in presence of participating media with general properties,” in *Proceedings of the Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 69–85, June 1994.
17. N. Max, “Efficient light propagation for multiple anisotropic volume scattering,” in *Proceedings of the Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 87–104, June 1994.
18. F. Sillion, “Clustering and volume scattering for hierarchical radiosity calculations,” in *Proceedings of the Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 105–117, June 1994.
19. E. Lafortune and Y. Willems, “Using the modified phong brdf for physically based rendering,” Technical Report CW197, Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium, Nov. 1994.
20. J. Halton, “On the relative merits of correlated and importance sampling for monte carlo integration,” *Proceedings of the Cambridge Philisophical Society*, vol. 61, pp. 497–498, 1965.
21. E. Lafortune and Y. Willems, “The ambient term as a variance reducing technique for monte carlo ray tracing,” in *Proceedings of the Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 163–171, June 1994.
22. E. Lafortune and Y. Willems, “Bi-directional path tracing,” in *Proceedings of CompuGraphics*, (Alvor, Portugal), pp. 145–153, Dec. 1993.
23. E. Veach and L. Guibas, “Bidirectional estimators for light transport,” in *Proceedings of the Fifth Eurographics Workshop on Rendering*, (Darmstadt, Germany), pp. 147–162, June 1994.
24. Ph. Dutré and Y. Willems, “Potential-driven monte carlo particle tracing for diffuse environments with adaptive probability density functions,” in *Proceedings of the Sixth Eurographics Workshop on Rendering*, (Dublin, Ireland), June 1995.
25. H. Jensen, “Importance driven path tracing using the photon map,” in *Proceedings of the Sixth Eurographics Workshop on Rendering*, (Dublin, Ireland), June 1995.