Weighted Importance Sampling in Shooting Algorithms

Category: survey

Abstract

This paper proposes the application of a variance reduction technique called weighted importance sampling in shooting type global illumination algorithms. The sampling applied by shooting type Monte-Carlo global illumination algorithms can mimic the power transfer, but not the BRDFs at the visible target of the transfer. Consequently, these algorithms are poor in rendering visible specular surfaces. In order to eliminate these drawbacks, the BRDFs at the visible targets are taken into account as an additional weighting of the sampling density. After discussing the basic concepts we demonstrate the proposed idea with two algorithms. The first one uses conventional rays, while the second one ray-bundles to transfer the light in the scene.

Keywords: Monte-Carlo integration, variance reduction, importance sampling.

1 Introduction

Global illumination algorithm should compute the average of the radiance values on the area visible through a pixel or leaving surface patches in the direction of the eye:

$$C_A = \frac{1}{A} \cdot \int_A L(\vec{x}, \omega_{eye}) d\vec{x},$$

where A is the area of the surface on which the average is computed. We shall call this area as the *area of interest*, which can either be the area visible through a pixel or the area of a patch if finite-element representation is used. According to the rendering equation, the radiance is the sum of the emission and a reflected component that can be obtained by reflecting the radiance of all points that are visible from here. Let us concentrate on the reflected component since the emission is easy to compute. The average of the reflected radiance is:

$$C_A = \frac{1}{A} \cdot \int_A \int_\Omega L^{in}(\vec{x}, \omega') \cdot f_r(\omega', \vec{x}, \omega_{eye}) \cdot \cos\theta'_{\vec{x}} \, d\omega' d\vec{x},$$

where L^{in} is the incoming radiance at point $\vec{x}$, f_r is the BRDF and $\theta'_{\vec{x}}$ is the angle between the surface normal and incoming direction ω'. The product of the BRDF and the cosine of the incoming angle is the *scattering density* $w(\omega', \vec{x}, \omega_{eye})$ that expresses the probability density that scattering connects directions ω_{eye} and ω'.

This integral is often evaluated by Monte-Carlo quadrature. Classical Monte-Carlo quadrature would take M random samples with probability density $p(\vec{x}, \omega')$ and approximate the integral as follows:

$$C_A = \frac{1}{A} \cdot \int_A \int_\Omega L^{in}(\vec{x}, \omega') \cdot w(\omega', \vec{x}, \omega_{eye}) \, d\omega' d\vec{x} \approx$$

$$\frac{1}{M} \cdot \sum_{n=1}^{M} \frac{L^{in}(\vec{x}_n, \omega'_n) \cdot w(\omega'_n, \vec{x}_n, \omega_{eye})}{A \cdot p(\vec{x}_n, \omega'_n)}. \quad (1)$$

In order to reduce the variance, probability density p should mimic the integrand. This approach is called *importance sampling*. Sampling according to a given probability density is carried out by transforming uniformly distributed numbers provided by the pseudo or quasi random number generator. This transformation requires the inverse of the cumulative probability distribution, thus p should be analytically integrable and we should be able to compute the inverse of its integral. These requirements can be met only if p is algebraically simple, which makes it impossible to accurately mimic the integrand. On the other hand, the incoming radiance L^{in} is not available, but we have to use another Monte-Carlo estimation to approximate it.

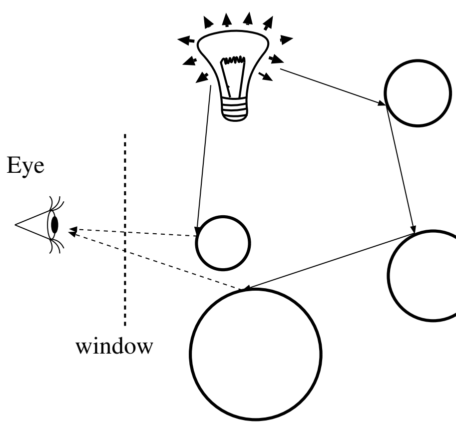

Figure 1: Shooting-type walks

Examining the integrand we can note that this is the product of the incoming radiance and the scattering density. Since it seems hopeless to sample according to this product, Monte-Carlo algorithms try to mimic either the scattering probability at the eye transfer, or the incoming radiance. The first approach is followed in *gathering algorithms*, such as in path tracing, while the second is in *shooting algorithms* (figure 1).

In random walk algorithms paths are initiated at the light sources and are terminated by the rules of Russian roulette. If Russian roulette decides on terminating the walk, then a new walk is initiated. In iteration algorithms, on the other hand, not only the target of the last transfer, but all previous transfers and the light source are potential sources.

Both in random walk and in iteration, an elementary operation of the solution is finding the source of the light transfer, sampling the destination (or the direction towards the destination), then computing the transfer between the source and the destination as the ratio of the integrand and the probability density of sampling. Camera contributions are computed in both cases by connecting each visited point with the eye deterministically. At the end of each transfer, shooting algorithms should check whether or not this transfer has any effects on the camera, i.e. the target is in the area of interest. If this transfer has a camera contribution, then the light is reflected towards the eye using the local BRDF and the estimate is obtained as the product of the estimate of the transfer and the scattering density. The final result will be the average of such estimates. These deterministic connections pose problem if the surfaces are specular since they cannot mimic the important directions of the given BRDFs.

Another problem which needs to be addressed is that we have many areas of interest simultaneously. The color of all pixels or the eye radiance of all patches should be computed at the same time. The calculation of different areas of interest can be considered as a single sampling process if the domain of the integration is extended from the area visible in a pixel to the total surface area S, while the integrand is multiplied by an indicator function $\xi_A(\vec{x})$. This indicator function is 1 if $\vec{x}$ is in area of interest A, otherwise it is zero. Using this indicator function, the Monte-Carlo estimation is the following:

$$\frac{1}{A} \cdot \int_S \int_\Omega L^{in}(\vec{x}, \omega') \cdot w(\omega', \vec{x}, \omega_{eye}) \cdot \xi_A(\vec{x}) \, d\omega' d\vec{x} \approx$$

$$\frac{1}{M} \cdot \sum_{n=1}^{M} \frac{L^{in}(\vec{x}_n, \omega'_n) \cdot w(\omega'_n, \vec{x}_n, \omega_{eye}) \cdot \xi_A(\vec{x})}{A \cdot p(\vec{x}_n, \omega'_n)}, \quad (2)$$

where $p(\vec{x}_n, \omega'_n)$ is a probability density in $S \times \Omega$ and not only in $A \times \Omega$. In image synthesis we should evaluate these integrals for all areas of interest. We concluded that the integrand of the visible colors is a product of the transferred radiance and the scattering density at the target. As we shall see, shooting algorithms can sample according to the transferred radiance, but not according to the scattering probability at the target and not according to indicator ξ_A. This is exactly the place where the proposed technique, the application of weighted importance sampling comes into play.

Weighted importance sampling is a rather old method [6] which has received just little attention in rendering so far. The exception is the pioneering paper [2], which applied this technique in stochastic iteration of the radiosity equation. However, we believe that this technique has more potential in other algorithms, and particularly in the non-diffuse setting.

The structure of this paper is the following. In the next section we review the theory of weighted importance sampling then discuss how it can be incorporated in shooting algorithms. In order to demonstrate its application, a ray-based and a ray-bundle based algorithms are equipped with this improvement.

2 Weighted Importance Sampling

Suppose that integral $F = \int_V f(\mathbf{z}) \, d\mathbf{z}$ needs to be evaluated by Monte-Carlo quadrature. The classical Monte-Carlo approach would compute the following sum:

$$F = \int_V f(\mathbf{z}) \, d\mathbf{z} \approx \frac{1}{M} \cdot \sum_{n=1}^{M} \frac{f(\mathbf{z}_n)}{p(\mathbf{z}_n)},$$

where p is the sampling density, which should mimic integrand f. In practical cases p cannot mimic f accurately and be appropriate for sample generation at the same time. Weighted importance sampling [6, 2] attacks this problem by working with two probability densities simultaneously. Suppose we have probability density $g(z)$ that is quite good in mimicking integrand f but we are unable to sample according to this density due to its algebraic complexity. On the other hand, we also have another probability density $p(z)$ which is possibly poorer in mimicking f but is appropriate to construct a sampling scheme. Weighted importance sampling proposes the following quadrature formula to estimate the integral:

$$\int_V f(\mathbf{z}) \, d\mathbf{z} \approx \frac{\sum_{n=1}^{M} f(\mathbf{z}_n)/p(\mathbf{z}_n)}{\sum_{n=1}^{M} g(\mathbf{z}_n)/p(\mathbf{z}_n)},$$

where samples $\mathbf{z}_n$ are obtained with probability density p. In order to demonstrate that this quadrature is asymptotically equivalent to the original Monte-Carlo quadrature, let us divide both the numerator and the denumerator by the number of samples M:

$$\frac{\frac{1}{M} \cdot \sum_{n=1}^{M} f(\mathbf{z}_n)/p(\mathbf{z}_n)}{\frac{1}{M} \cdot \sum_{n=1}^{M} g(\mathbf{z}_n)/p(\mathbf{z}_n)}.$$

The new numerator is the same as the original integral quadrature, thus it converges to the integral. The denumerator, on the other hand, is the Monte-Carlo estimate of integral $\int_V g(\mathbf{z}) \, d\mathbf{z}$. Since g is a probability density function, its integral is 1, thus the new quadrature converges to the same value as the old quadrature.

The question is whether or not this new estimate is better than the old one. This depends on whether or not density g is better in mimicking f than p. The formal analysis

[6] results in the following formulae. The mean square error after M samples obtained with weighted importance sampling, including both the bias and the variance, is

$$\varepsilon_{WIS} \approx \frac{1}{M} \cdot \int_V \left(\frac{f(\mathbf{z})}{p(\mathbf{z})} - F \cdot \frac{g(\mathbf{z})}{p(\mathbf{z})} \right)^2 \cdot p(\mathbf{z}) \, d\mathbf{z}. \quad (3)$$

For the sake of comparison, we also present the mean square error of the classical Monte-Carlo estimator:

$$\varepsilon_{CMC} \approx \frac{1}{M} \cdot \int_V \left(\frac{f(\mathbf{z})}{p(\mathbf{z})} - F \right)^2 \cdot p(\mathbf{z}) \, d\mathbf{z}. \quad (4)$$

Instead of repeating the proof, we provide an intuitive explanation. Suppose that p is poor in sampling a particular sub-domain $\mathcal{S}$, i.e. it does not generate samples here as frequently as would be required by the large integrand values (if classical Monte-Carlo method is used, when we are lucky enough to generate a sample in sub-domain $\mathcal{S}$, we get a large integrand value that is divided by a small probability, which results in a huge term in the average approximating the integral). These infrequent but huge values are responsible for large fluctuations. However, if this method is used with a probability density g according to weighted importance sampling, then the approximating sum is also divided by the sum of $g(\mathbf{z}_n)/p(\mathbf{z}_n)$ terms. When we are not lucky to sample the important regions, g will also be small, thus the denumerator will be smaller than one. Dividing by the denumerator, the approximation will be scaled up. However, when the sample is in important sub-domain $\mathcal{S}$, the Monte-Carlo estimate and the denumerator will be increased simultaneously by a larger value. Since the fluctuations of the numerator and the denumerator are thus synchronized, the fluctuation of their ratio is decreased.

However, weighted importance sampling may have not only advantages but disadvantages as well. It has a small bias, which disappears quickly. On the other hand, if p is already good enough to mimic integrand f, then the numerator will be stable. The fluctuation of the denumerator around 1 appears just as an additional noise.

Thus we can conclude that weighted importance sampling should be used carefully, since it can reduce and increase the variance depending on the mimicking capabilities of the two probability densities. Let us examine the difference of the mean square errors of the estimator of classical Monte-Carlo and the estimator of weighted Monte-Carlo:

$$\varepsilon_{CMC} - \varepsilon_{WIS} =$$
$$\frac{1}{M} \cdot \int_V F \cdot \left(\frac{g(\mathbf{z})}{p(\mathbf{z})} - 1 \right) \cdot \left(\frac{2f(\mathbf{z})}{p(\mathbf{z})} - F \left(\frac{g(\mathbf{z})}{p(\mathbf{z})} + 1 \right) \right) \cdot p(\mathbf{z}) \, d\mathbf{z}$$

In order to get improvement, this difference should be positive. Note that the integrand of the error difference is a product where the first factor cannot be negative, but the second and the third factors can. Thus improvement is guaranteed if the second and the third terms change their signs simultaneously. There are two cases. The quadrature overestimates in $\mathbf{z}$ if $f(\mathbf{z})/p(\mathbf{z}) > F$. On the other hand, the quadrature underestimates in $\mathbf{z}$ if $f(\mathbf{z})/p(\mathbf{z}) < F$.

In case of overestimation the new probability density g should meet the following requirement in order to make the integrand of the error difference positive:

$$\frac{f(\mathbf{z})}{F} \geq g(\mathbf{z}) > p(\mathbf{z}).$$

It means that g should also result in overestimation but its level should be decreased. This statement can be proven by checking that in this case both the second and the third factors are positive.

In case of underestimation, the requirement of the integrand of the error difference being positive is

$$\frac{f(\mathbf{z})}{F} \leq g(\mathbf{z}) < p(\mathbf{z}),$$

thus the level of underestimation should be decreased. In this case the second and third factors are both negative.

Generally these requirements are not easy to met. We shall consider a particularly important case when this problem can be solved. Suppose that there is a considerable sub-domain in V where integrand f and density g are zero but p is not. That is, g can mimic these zero integrand points but p cannot. In this sub-domain the integrand is underestimated since $f(\mathbf{z})/p(\mathbf{z}) = 0 < I$. When f is non-zero then f/p should usually be larger than I in order to compensate the zero values. Thus when f is non-zero, then usually overestimation happens. Density g should also be usually larger than p here since their integrands are 1 and g has a smaller domain where it is non zero.

3 Application of weighted importance sampling in shooting algorithms

Suppose that we use a sampling method (i.e. a shooting algorithm) that transfers light to point $\vec{x}$ from direction ω' with probability density $p(\vec{x}, \omega')$. As we shall see, shooting algorithms can make p more or less proportional to the radiance or power of this transfer and the cosine of the incoming angle, but the sampling density is unable to mimic BRDF $f_r(\omega', \vec{x}, \omega_{eye})$ and indicator ξ_A at the target of the shooting. Thus these factors appear as additional weighting that can be responsible for a large variation. The variation is especially high if the target surface is specular or the probability of hitting the area of interest is small. In such situations, gathering would be better, which mimics the indicator of the area of interest and the scattering probability of the receiver point. We intend to use shooting, but to incorporate the more efficient density of gathering when these difficult situations happen. Weighted importance sampling allows us to pretend we follow gathering even when the sample was obtained with shooting. Whether or not it is worth doing may change from area of interest to area of interest. It is definitely worth doing when the surfaces are specular, thus we shall use this approach for the specular component of the reflection. We note that weighted importance sampling might also be also good for diffuse surfaces if the probability of hitting the area of interest is small since it can eliminate the variation of caused by the indicator, as recognized in [2]. However, we shall not analyze this possibility in this paper.

The reflected radiance, the BRDF and the scattering density are decomposed to diffuse and specular terms:

$$\begin{aligned} L^r &= L^d + L^s, \\ f_r(\omega', \vec{x}, \omega) &= f_d + f_s(\omega', \vec{x}, \omega), \\ w(\omega', \vec{x}, \omega) &= f_d \cdot \cos\theta'_{\vec{x}} + w_s(\omega', \vec{x}, \omega). \end{aligned}$$

We propose density g to mimic the specular scattering density and the indicator of the area of interest when the specular component is computed, that is the densities of the corresponding gathering algorithm. If the light is transferred on several wavelengths simultaneously, then the specular BRDF and the specular scattering density are functions of the wavelength. In this case, we can use the luminance of the specular scattering density. The luminance function is denoted by $\mathcal{L}$. Function g should be a probability density, thus we have to ensure that its integral is 1. To meet this requirement, the specular scattering density is divided by its integral and the indicator by the size of the area of interest. The integral of the specular scattering density is the specular albedo:

$$a_s(\vec{x}, \omega) = \int_{\Omega} w_s(\omega', \vec{x}, \omega)\, d\omega'.$$

In fact the albedo is only a function of the incoming angle θ' for isotropic materials. Thus an appropriate g, which is a probability density, is

$$g(\vec{x}, \omega') = \frac{\mathcal{L}(w_s(\omega', \vec{x}, \omega_{eye}))}{\mathcal{L}(a_s(\vec{x}, \omega_{eye}))} \cdot \frac{\xi_A(\vec{x})}{A}.$$

Substituting this density into the formula of weighted importance sampling we obtain :

$$C_A = \frac{\sum_{n=1}^{M} \frac{L^{in}(\vec{x}_n, \omega'_n) \cdot w_s(\omega'_n, \vec{x}_n, \omega_{eye}) \cdot \xi_A(\vec{x}_n)}{p(\vec{x}_n, \omega'_n)}}{A \cdot \sum_{n=1}^{M} \frac{g(\vec{x}_n, \omega'_n)}{p(\vec{x}_n, \omega'_n)}} =$$

$$\frac{\sum_{n=1}^{M} \frac{L^{in}(\vec{x}_n, \omega'_n) \cdot w_s(\omega'_n, \vec{x}_n, \omega_{eye}) \cdot \xi_A(\vec{x}_n)}{p(\vec{x}_n, \omega'_n)}}{\sum_{n=1}^{M} \frac{\mathcal{L}(w_s(\omega'_n, \vec{x}_n, \omega_{eye})) \cdot \xi_A(\vec{x}_n)}{\mathcal{L}(a_s(\vec{x}, \omega_{eye})) \cdot p(\vec{x}, \omega'_n)}}.$$

Note that the size of area of interest disappears from the formula, since the effect of averaging is compensated by making g a probability density. In order to apply this formula in practice, an accumulating specular radiance L_A and an accumulating probability d_A are assigned to each area of interest. These values are incremented only if this area of interest is the target of the current transfer. The accumulating radiance is incremented by

$$\frac{L^{in}(\vec{x}_n, \omega'_n) \cdot w_s(\omega'_n, \vec{x}_n, \omega_{eye})}{A \cdot p(\vec{x}_n, \omega'_n)}.$$

The accumulating probability is incremented by

$$\frac{\mathcal{L}(w_s(\omega'_n, \vec{x}_n, \omega_{eye}))}{\mathcal{L}(a(\vec{x}_n, \omega_{eye})) \cdot A \cdot p(\vec{x}_n, \omega'_n)}.$$

The estimate of the eye radiance is expressed as the sum of the specular reflected radiance and diffuse radiance L^d:

$$L^r(\omega_{eye}) \approx \frac{L_A^s}{d_A} + L^d.$$

In the following subsections we examine two different shooting like algorithms and discuss how the presented weighted importance sampling method can be applied to them. These algorithms apply random transport operators $\mathcal{T}^*$ to the radiance estimate of the given iteration step in a way that its expected value gives back the effect of the application of the original light transport operator. Iterating the radiance such a way, the iterated functions will fluctuate around the expected radiance function. Computing the averages from the estimates of different iteration steps, the final, converged result can be obtained.

3.1 Method 1: Ray-shooting

In order to solve the global illumination problem this algorithm simulates the light transfer by random rays. A temporary random radiance approximation is stored on the patches obtained tessellating the original surfaces. From this temporary radiance approximation, we can estimate the output radiance of each patch in each direction. Ray-shooting transfers the radiance between two random points connected by a ray. In order to use importance sampling, the source point and the direction are sampled proportional to the cosine weighted radiance.

Let us sample $\vec{y}$ and ω' proportional to $L(\vec{y}, \omega') \cos\theta_{\vec{y}}$ and then $\vec{x}$ by the ray shooting process. This can be realized by first finding patch i of $\vec{y}$ proportional to its power, i.e. the selection probability is the ratio of the power of patch j:

$$\Phi_j = A_j \cdot \int_{\Omega} L_j(\omega') \cos\theta_{\vec{y}}\, d\omega',$$

and the total power of the scene:

$$\Phi = \int_S \int_{\Omega} L(\vec{y}, \omega') \cos\theta_{\vec{y}}\, d\omega' d\vec{y} = \sum_k \Phi_k.$$

Then source $\vec{y}$ is sampled on this patch uniformly, finally, direction ω' is sampled proportional to directional distri-

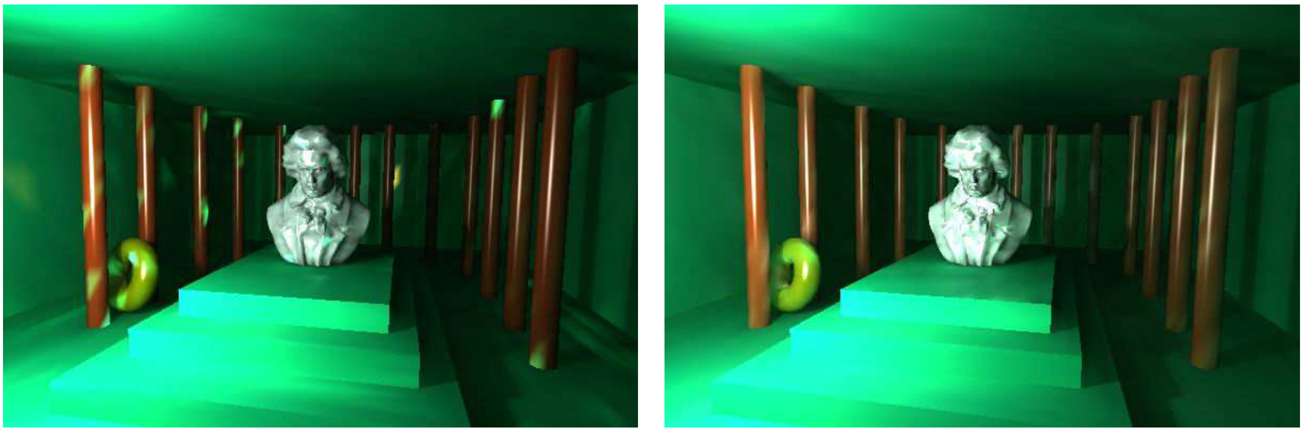

Figure 2: A Beethoven scene rendered by classical Monte-Carlo (left) and weighted Monte-Carlo (right) using 54 seconds on a P4 1.4 Ghz computer. The scene consists of 21 thousand patches. We used stochastic ray-shooting for the calculation of the indirect illumination while the direct illumination was obtained by a deterministic method.

bution $L(\vec{y}, \omega')$. The final sampling density of $\vec{y}$ and direction ω' is

$$p_{\vec{y}}(\vec{y}, \omega') = \frac{\mathcal{L}(L(\vec{y}, \omega')) \cos\theta_{\vec{y}}}{\mathcal{L}(\Phi)}.$$

It often happens that we are unable to sample exactly with this distribution, thus $p_{\vec{y}}(\vec{y}, \omega')$ is only roughly proportional to the radiance.

When the area of interest is examined, probability density $p_{\vec{y}}(\vec{y}, \omega')$ should be transformed to the probability density of hitting $\vec{x}$ from direction ω':

$$p(\vec{x}, \omega') = \frac{p_{\vec{y}}(\vec{y}, \omega') \cdot \cos\theta_{\vec{x}}}{\cos\theta_{\vec{y}}}.$$

The random transfer operator results in non-zero radiance just at the hit point $\vec{x}$. The contribution to the area of interest is

$$C_A = \frac{1}{A} \cdot \frac{L(\vec{y}, \omega') \cos\theta_{\vec{y}}}{p_{\vec{y}}(\vec{y}, \omega)} \cdot f_s(\omega', \vec{x}, \omega_{eye}) \cdot \xi_A(\vec{x}).$$

Note that this estimator is quite smooth except for the BRDF and indicator ξ_A since the variation of the incoming radiance approximation is compensated by the corresponding factor in the sampling probability. Thus when the variation of the BRDF exceeds the variation of the incoming illumination, then weighted importance sampling can improve the estimate.

Weighted importance sampling will maintain an accumulating eye radiance and an accumulating probability at each areas of interest. The accumulating eye radiance L_A^s is incremented by

$$\frac{1}{A} \cdot \frac{L(\vec{y}, \omega') \cos\theta_{\vec{y}}}{p_{\vec{y}}(\vec{y}, \omega')} \cdot f_s(\omega', \vec{x}, \omega_{eye}).$$

when the ray hit this area (i.e. when $\xi_A(\vec{x}) = 1$). At the same time, the accumulated probability is incremented by

$$\frac{\mathcal{L}(w_s(\omega'_n, \vec{x}_n, \omega_{eye}))}{\mathcal{L}(a_s(\vec{x}, \omega')) \cdot A \cdot p(\vec{x}_n, \omega'_n)} =$$

$$\frac{\mathcal{L}(f_s(\omega'_n, \vec{x}_n, \omega_{eye}))}{\mathcal{L}(a_s(\vec{x}_n, \omega_{eye})) \cdot A \cdot p_{\vec{y}}(\vec{y}, \omega')}.$$

The estimate of the eye radiance is expressed as the ratio of the accumulated eye radiance and accumulated probability:

$$L^r(\omega_{eye}) \approx \frac{L_A^s}{d_A} + L^d.$$

Let us interpret the results. Classical Monte-Carlo estimate would use the estimate L_A^s/M, and in our case d_A converges to M, thus asymptotically, the two estimates are equivalent. Suppose that we have two areas of interest having similar specular reflectance and similar illumination conditions. The number of hits is expected to be proportional to the incoming power. On the other hand, the direction of the hits follow the directional variation of the illumination. If we have moderate number of samples, it can happen, for example, that due to the random nature of sampling, the first patch gets more samples and from the important directions of the specular reflection, while the second patch less samples from the unimportant directions. Both estimates are divided by the same M in classical Monte-Carlo. Thus, although their expected values are similar, the first patch can be much brighter than the second, which is responsible for large variance (left of figure 2). However, when weighted importance sampling is used, the accumulating probability will also be much larger for the first patch since it is incremented more times and by larger values. Thus when dividing with accumulating probability d_A, the difference between the lucky and the unlucky patches disappears (right of figure 2).

3.2 Method 2: Ray-bundle shooting

In this second algorithm we use bundle of rays to transfer the radiance in the scene.

Parallel ray-bundle tracing transfers the radiance of all patches parallel to a randomly selected global line in each iteration cycle [8]. Bundles carry out surface integration on the fly, thus only the directional integral should be computed. If the global directions are sampled from a uniform distribution and the radiance is transferred into two opposite directions, then the directional density is:

$$p(\omega') = \frac{1}{2\pi}.$$

The random transport operator generates a random approximation of the reflected radiance at each step:

$$L^r \approx \mathcal{T}^*_{par} L = \frac{L(\vec{y},\omega') \cdot f_r(\omega',\vec{x},\omega) \cdot \cos\theta'_{\vec{x}}}{p(\omega')} =$$

$$2\pi \cdot L(\vec{y},\omega') \cdot f_r(\omega',\vec{x},\omega) \cdot \cos\theta'_{\vec{x}}.$$

where $\vec{y}$ is the point visible from $\vec{x}$ at direction $-\omega'$ or at ω depending the orientation of the surface.

The radiance transfer needs the identification of those points that are mutually visible in the global direction. In order to solve this global visibility problem, a window is placed perpendicular to the global direction. The window is decomposed into a number of pixels. A pixel is capable to store a list of patch indices and z-values. The lists are sorted according to the z-values. The collection of these pixels are called the *transillumination buffer*[4]. The patches are rendered one after the other into the buffer using a modified z-buffer algorithm which keeps all visible points not just the nearest one. Traversing the generated lists the pairs of mutually visible points can be obtained. For each pair of points, the radiance transfer is computed and the transferred radiance is multiplied by the BRDF, resulting in a random estimate of the reflected radiance.

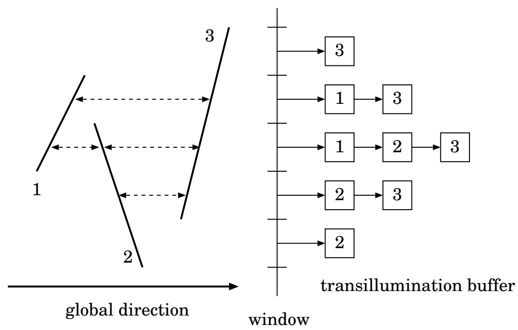

Figure 3: Organization of the transillumination buffer

The random estimate of the contribution in iteration step n is:

$$C_A = \frac{1}{A} \cdot \int_A \mathcal{T}^*_{par} L_{n-1}\, d\vec{x} \approx 2\pi \cdot I_n \cdot f_r(\omega',\vec{x},\omega),$$

where

$$I_n = \frac{\delta P}{A} \cdot \sum_P L^{in}_{n-1}(P)$$

is the irradiance received by this area in iteration step n. In this formula P runs on the pixels covering the projection of area of interest A, $L^{in}_{n-1}(P)$ is the radiance of the surface point visible in pixel P in iteration step $n-1$, $f_r(\omega',\vec{x},\omega)$ is the BRDF of that point which receives this radiance coming through pixel P, and δP is the area of the pixels.

Concerning the eye radiance $2\pi \cdot I_n \cdot f_r(\omega',\vec{x},\omega_{eye})$ the fluctuation can stem from both the fluctuation of the irradiance I_n and from the fluctuation of the BRDF. The latter type of fluctuation can be reduced by weighted importance sampling.

The first variable of weighted importance sampling is the accumulating eye radiance coming from specular reflections, which is incremented by

$$\frac{I(\vec{x},\omega') \cdot f_s(\omega',\vec{x},\omega)}{p(\omega')} = 2\pi \cdot I(\vec{x},\omega') \cdot f_s(\omega',\vec{x},\omega)$$

in each iteration step. The accumulating probability is incremented by

$$\frac{\mathcal{L}(w_s(\omega'_n,\vec{x},\omega_{eye}))}{\mathcal{L}(a_s(\vec{x}_n,\omega_{eye})) \cdot p(\omega'_n)} = \frac{2\pi\mathcal{L}(w)s(\omega'_n,\vec{x},\omega_{eye}))}{\mathcal{L}(a_s(\vec{x}_n,\omega_{eye}))}.$$

The final radiance is obtained as dividing the specular eye radiance by the accumulated probability and then adding the diffuse component.

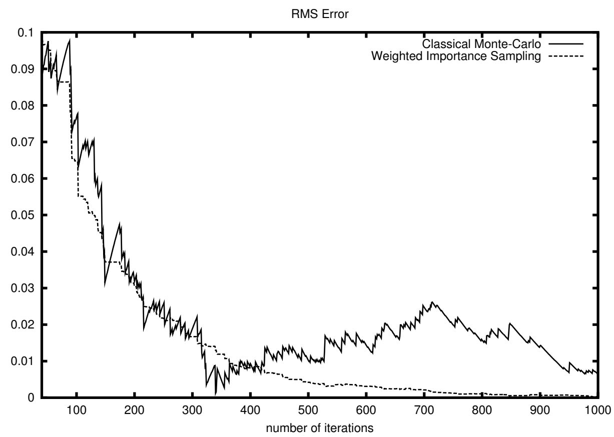

Figure 4: Comparison of the classical Monte-Carlo and the Weighted Importance Sampling methods with parallel ray bundles

Figure 4 compares the error curves of classical Monte-Carlo method with the proposed weighted importance sampling. Note that the error of weighted importance sampling is not only smaller, but the large error fluctuation is eliminated (figure 5). This means that even at early stages, when the result is not accurate, the disturbing large variations will not be present in the image.

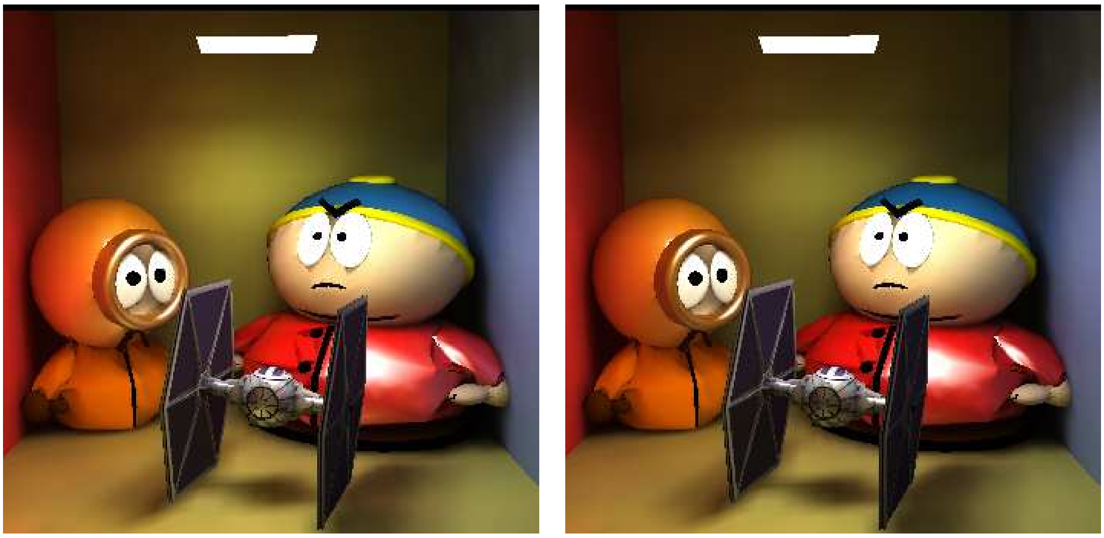

Figure 6: The same scene rendered without (left) and with weighted importance sampling (right). The images have been rendered on 800×800 resolution in 28 seconds, including an initial first shot that replaced the light sources by their first reflection.

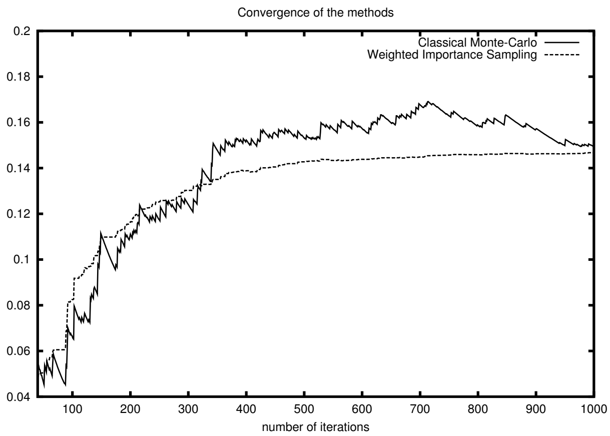

Figure 5: Convergence of a single area of interest

Conclusions

This paper proposed the application of a known variance reduction technique in non-diffuse global illumination problems. In our approach the weighting emphasizes the points and directions from which significant eye contribution is possible. In this way, we could incorporate a probability density used in gathering algorithms into shooting methods. Thus, while preserving the advantages of shooting techniques (i.e. they can efficiently distribute the illumination of the light sources), we could reduce its drawbacks (i.e. measuring specular surfaces). The algorithm is cheap computationally and requires just the separation of diffuse and specular radiance and one additional scalar value to store the accumulated radiance per patch or per pixel. The proposed idea was implemented in ray-shooting and ray-bundle shooting algorithms. In ray-shooting we can expect that weighted importance sampling reduces both the variance stemming from not following the BRDFs at the target and from the random fluctuation of the number of hits per target. In the method applying bundles, the random fluctuation of the number of hits has already been eliminated, thus we could observe less improvement.

References

[1] Ph. Bekaert. *Hierarchical and stochastic algorithms for radiosity*. PhD thesis, University of Leuven, 1999. http://www.cs.leuven.ac.be/ cwis/research/graphics/ CGRG.PUBLICATIONS/PHBPPHD.

[2] Philippe Bekaert, Mateu Sbert, and Yves D. Willems. Weighted importance sampling techniques for monte carlo radiosity. In B. Peroche and H. Rushmeier, editors, *Rendering Techniques 2000 (Proceedings of the Eleventh Eurographics Workshop on Rendering)*, pages 35–46, New York, NY, 2000. Springer Wien.

[3] M. Cohen and D. Greenberg. The hemi-cube, a radiosity solution for complex environments. In *Computer Graphics (SIGGRAPH '85 Proceedings)*, pages 31–40, 1985.

[4] L. Neumann. Monte Carlo radiosity. *Computing*, 55:23–42, 1995.

[5] S. N. Pattanaik and S. P. Mudur. Adjoint equations and random walks for illumination computation. *ACM Transactions on Graphics*, 14(1):77–102, 1995.

[6] M. Powell and J. Swann. Weighted importance sampling — a monte-carlo techniques for reducing variance. *Inst. Maths. Applics.*, 2:228–236, 1966.

[7] M. Sbert. *The Use of Global Directions to Compute Radiosity*. PhD thesis, Catalan Technical University, Barcelona, 1996.

[8] L. Szirmay-Kalos. Stochastic iteration for non-diffuse global illumination. *Computer Graphics Forum (Eurographics'99)*, 18(3):233–244, 1999.

Weighted Importance Sampling in Shooting Algorithms

László Szirmay-Kalos
Antal György
Benedek Balázs,
Budapest University of Technology

Category: survey

Format: print

Contact:	László Szirmay-Kalos
	Budapest, Magyar Tudósok krt. 2, H-1117, HUNGARY
phone:	(361) 463–2030
fax:	(361) 463–2204
email:	szirmay@iit.bme.hu

Estimated # of pages: 8

Keywords: Monte-Carlo integration, variance reduction, importance sampling.

This paper proposes the application of a variance reduction technique called weighted importance sampling in shooting type global illumination algorithms. The sampling applied by shooting type Monte-Carlo global illumination algorithms can mimic the power transfer, but not the BRDFs at the visible target of the transfer. Consequently, these algorithms are poor in rendering visible specular surfaces. In order to eliminate these drawbacks, the BRDFs at the visible targets are taken into account as an additional weighting of the sampling density. After discussing the basic concepts we demonstrate the proposed idea with two algorithms. The first one uses conventional rays, while the second one ray-bundles to transfer the light in the scene.