Representativity for Robust and Adaptive Multiple Importance Sampling

Anthony Pajot, Loïc Barthe, Mathias Paulin, and Pierre Poulin

Abstract—We present a general method enhancing the robustness of estimators based on multiple importance sampling (MIS) in a numerical integration context. MIS minimizes variance of estimators for a given sampling configuration, but when this configuration is less adapted to the integrand, the resulting estimator suffers from extra variance. We address this issue by introducing the notion of "representativity" of a sampling strategy, and demonstrate how it can be used to increase robustness of estimators, by adapting them to the integrand. We first show how to compute representativities using common rendering informations such as BSDF, photon maps or caches in order to choose the best sampling strategy for MIS. We then give hints to generalise our method to any integration problem and demonstrate that it can be used successfully to enhance robustness in different common rendering algorithms.

Index Terms—[I.6.8.g] Monte-Carlo ; [I.3.7] Three-Dimensional Graphics and Realism

1 Introduction

Global illumination algorithms focus on solving the *light transport equation* (LTE) [1]:

$$L_o(\mathbf{x},\omega_o) = L_e(\mathbf{x},\omega_o) + \int_{\mathcal{S}^2} f_s\left(\mathbf{x},\omega_i \leftrightarrow \omega_o\right) L_i(\mathbf{x},\omega_i)\left|(\mathbf{N_x}.\omega_i)\right| d\sigma\left(\omega_i\right), \quad (1)$$

where $L_o(\mathbf{x},\omega_o)$ is the outgoing radiance at point $\mathbf{x}$ along direction ω_o, $L_e(\mathbf{x},\omega_o)$ is the self-emitted radiance, $L_i(\mathbf{x},\omega_i)$ is the incoming radiance, $f_s\left(\mathbf{x},\omega_i \leftrightarrow \omega_o\right)$ is the bidirectional scattering distribution function (BSDF), and $\mathbf{N_x}$ is the normal at point $\mathbf{x}$.

In the general case, no analytical solutions to this equation are known, resorting to the use of numerical integration methods. The Monte-Carlo method is widely used because it does not need any analytical property in order to converge to the correct result. For a general integrand $f(x)$ defined over a space Ω, Monte-Carlo defines for $I = \int_\Omega f(x)dx$ the following estimator:

$$I \approx F_N = \frac{1}{N}\sum_{i=1}^{N}\frac{f(X_i)}{p(X_i)}, \quad (2)$$

where N is a fixed number of samples used to compute the integral and the X_is are random variables defined over Ω and sampled according to the probability distribution function (pdf) p. F_N is unbiased if, for each value x where $f(x) \neq 0$, $p(x) > 0$.

- *A. Pajot is Ph.D. student at IRIT-CNRS, Université de Toulouse. E-mail: anthony.pajot@irit.fr*
- *L. Barthe is associate professor at IRIT-CNRS, Université de Toulouse. E-mail: lbarthe@irit.fr*
- *M. Paulin is full professor at IRIT-CNRS, Université de Toulouse. E-mail: mathias.paulin@irit.fr*
- *P. Poulin is full professor at dept. I.R.O., Université de Montréal. E-mail: poulin@iro.umontreal.ca*

In its basic form and for a fixed pdf, the estimator F_N has a standard deviation in $O(N^{-1/2})$. When attempting at reducing the variance while sticking to the basic estimator, most of the work is done on the pdf. Importance sampling builds up on this: a pdf that better matches the integrand lowers the variance of its estimator. Each pdf specifically defined to focus on one part of the integrand leads to a *sampling strategy*.

In rendering, it is common to have one strategy to sample the light sources, which correctly matches the direct lighting part of the integrand, and one strategy to sample the BSDF, which correctly matches the glossy parts of the integrand. However building one strategy that takes into account both parts of the integrand at the same time is a difficult task, especially when no assumptions can be made on these two parts. Multiple importance sampling (MIS) [2] tries to lower the impact of this problem by combining several estimators, each using one strategy that correctly matches one part of the integrand.

If S strategies are available, each represented by a pdf p_i, the MIS framework defines two estimators, depending on whether one or several samples are drawn to evaluate the integral. The *one-sample* estimator:

$$F_{os} = w_i(X)\frac{f(X)}{c_i p_i(X)} \quad (3)$$

consists in first choosing a strategy p_i with a probability c_i, and then sampling it to evaluate the integral. w_i is a weighting function. The second estimator is the *multi-sample* estimator:

$$F_{ms} = \sum_{i=1}^{S}\left[\frac{1}{n_i}\sum_{j=1}^{n_i} w_i(X_{i,j})\frac{f(X_{i,j})}{p_i(X_{i,j})}\right], \quad (4)$$

where n_i is the number of samples generated using p_i. These estimators are unbiased as long as some

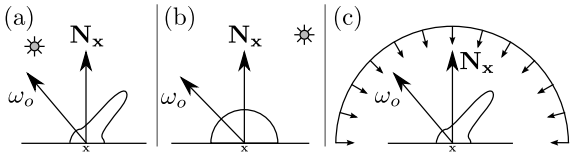


Fig. 1. (a) and (b) Cases where sampling using the BSDF leads to high variance when estimating direct lighting; most directions generated from ω_o do not reach the light source. (c) A case where sampling from light sources fails: most points generated on large light sources such as environment lighting are not in the specular lobe associated to ω_o.

constraints on w_i are satisfied, and any x for which $f(x) \neq 0$ can be generated by at least one pdf. Veach and Guibas [2] derive optimal weighting functions with respect to variance, for a *given* set of c_i or n_i values. This set of *a priori* fixed values is called *sampling configuration* throughout the rest of this paper. These optimal weighting functions are known as the balance heuristic. In the case of multi-sample estimator, the balance heuristic is only near-optimal, and Veach and Guibas provide other heuristics that may behave better in some cases, such as the power heuristic or the maximum heuristic.

MIS does not give any hint about which sampling configuration would lead to the lowest variance for a given integrand, and thus which strategy should be preferably used. As illustrated in Figure 1 for rendering, different integrands require different sampling configurations in order to get an optimal estimator with respect to variance. When the same configuration is used for all these cases, the variance of the estimators would greatly vary from one case to another, meaning they are not *robust*. Finding optimal sampling configurations is a challenging problem, as shown in Appendix A, and would require a huge amount of processing power to be solved with usual means.

The main contribution of this paper is an approach that allows us to compute adequate sampling configurations at a negligible cost, leading to more robust MIS estimators, without introducing any bias. We develop the notion of *representativity*, which is an empirical measure of the match of a strategy with an integrand. As shown in Figure 2, we derive from these representativities both the probability assigned to each strategy when using a one-sample estimator, and the number of samples that should be taken from each strategy when using a multi-sample estimator.

As presented in Section 2, two different approaches have been used to obtain variance reduction. The first approach is to create very specific methods, where the form of the integrands and the strategies are known in advance. The second is to define general methods, that do not benefit from the optimality results provided by the MIS framework. As defined in Section 3, representativity is general and can be applied to any integrand and strategies. Moreover, it is designed to be used from within the MIS framework, and can be used with methods that do not change the MIS framework, such as Quasi-Monte-Carlo ones. More specifically, representativities are the results of the evaluation of a so-called *representativity function*. This representativity function has to be crafted for each strategy used by a MIS estimator. Once this is done, our method automatically computes the sampling configuration that is used by the MIS estimator. In order to apply this method to rendering, we design such functions for a strategy sampling from Ashikhmin-Shirley BSDF's, and a strategy sampling from photon maps. As our method is based on empirical models, we assess its validity in Section 4 by performing numerical analyses on various cases where classic MIS estimators lack robustness. Once its validity is assessed, we show in Section 5 that representativity-based sampling can be used in any context where several importance sampling strategies can be pertinent depending on the integrand, focusing on its potential uses in rendering.

2 Previous Work

2.1 Importance Sampling Strategies for Rendering

Extensive research has involved designing efficient sampling strategies for common BRDFs [3], [4], and designing BRDFs (Bidirectional Reflectance Distribution Functions) and BTDFs (Bidirectional Transmission Distribution Functions) that are well suited for importance sampling, while still providing high-quality results [5], [6]. These strategies generate well-distributed samples where BSDF (Bidirectional Scattering Distribution Function, *i.e.*, BRDF + BTDF) values are larger. Unfortunately, sampling using only the BSDF fails in situations where lighting comes from within a small solid angle, as illustrated in Figure 1.a and 1.b.

Importance sampling of environment maps has also been thoroughly investigated [7], [8]. Similarly, sampling a point on area light sources is a straightforward strategy for computing direct lighting from local light sources [9]. However, using only this strategy to estimate direct lighting contribution at a surface point fails when the BSDF is highly glossy and light sources are large (which is the case of environment maps), since the solid angle within which energy is scattered to the outgoing direction is very small (Figure 1.c).

Sampling from directional maps to better capture indirect lighting effects has been investigated by Jensen [10], [11], and a robust method to sample directions based on particle footprints has been introduced by Hey and Purgathofer [12]. Pharr [13] uses photon maps to guide the final gathering of its improved photon mapping algorithm. In each case, they introduce a user-defined parameter that gives the probability to sample the BSDF instead of the map, and they do not deal with multiple maps.

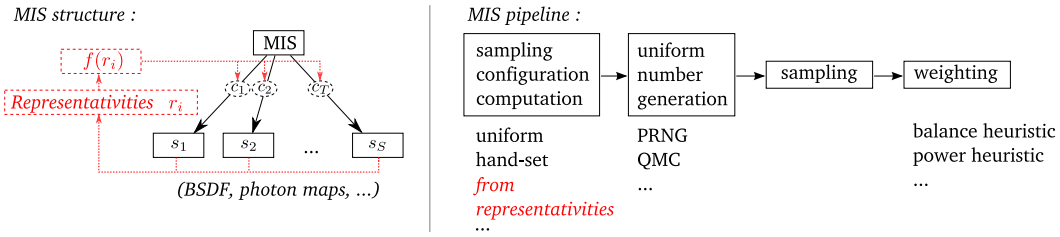


Fig. 2. Our contribution (in dashed-red and italics) relative to MIS framework. Representativities allow us to automatically derive empirically good probabilities for each of the available strategies. The examples of importance sampling strategies are taken from the rendering domain. Our method is orthogonal to the uniform number generation methods, as well as the weighting heuristic that is used to compute the final estimate. Therefore, it provides another way to improve the robustness of the MIS estimators, while benefiting from better uniform number generation methods or better heuristics.

2.2 General Variance Reduction

We are not aware of any published work focusing on the problem of automatically finding good sampling configurations. To our knowledge, only hand-set constants or uniformity are used.

Several methods other than MIS have been developed to create low-variance estimators, even for very complex integrands, such as Metropolis sampling or sampling importance resampling. All these methods rely on pseudo-random uniform number sequences. An extensive work has been done to create uniform number sequences with very good discrepancy properties, leading to the Quasi-Monte Carlo (QMC) methods.

Metropolis sampling [14] aims to sample from any target distribution. It uses mutations to transform a base sample and probabilistically accepting it as a new base sample, or keeping the previous one if the mutated sample is rejected. Using adequate acceptance tests, the base samples obtained by this process are distributed according to the target distribution. Besides the correlation between the samples, Metropolis sampling requires an initialization phase to create the first base sample, and performs several steps to converge to the target distribution. These two tasks are prohibitively expensive when generating a small number of samples from a target distribution. Moreover, mutations must be carefully designed to get a faster convergence to the target distribution, which is a difficult task. More generally, Markov-Chain Monte-Carlo methods, to which Metropolis sampling belongs, are not adapted when taking small numbers of samples from a target distribution, because of their complexity and initialization costs.

Sampling importance resampling [15] is another method to sample complex distributions by using simpler importance sampling strategies. It generates N samples from a single strategy s, and then filters the generated samples to create a distribution close to the function f we would like to sample from. If the pdf associated to s is quite different from f, the number of samples to generate using s can be large before reaching enough samples. This leads to useless computations, or increased variance when the number of samples to generate from s is fixed and too small.

In the domain of rendering, several methods have emerged, taking into account the form of the integrand given by the rendering equation [1]. Bidirectional importance sampling methods [16], [17], [18] have been mainly developed for direct lighting, where lighting is represented by environment maps. Recently, a similar method has been developed for indirect lighting represented by virtual point light sources [19]. These methods perform well for the cases they are designed for, but they are limited to the combination of only two importance sampling strategies (whether these strategies take several factors into account or not), whereas more sampling strategies could provide better results. Rousselle *et al.* [20] developed a method for sampling products of functions, but an intermediate hierarchical representation for each integrand must be built before sampling. Although handling an arbitrary number of functions in the product, the memory cost of their method is linear with the number of functions, and the efficiency of the sampling depends on the shapes of the functions.

(Randomised-)Quasi-Monte Carlo methods [21] aim at replacing the pseudo-random number generators by deterministic sequences that have very good discrepancy properties, leading to a better exploration of the sample space. These methods have been used in computer graphics for a while [22], with highly convincing results. Similarly to the other methods presented above, our method considers that the uniform numbers used for sampling are given by a black-box system. Therefore, it is very easy to benefit from the QMC pattern's low discrepancy properties to further lower the variance of the estimators, at the cost of introducing bias if a non-randomized QMC sequence is used.

3 Representativity-based Sampling

To render an image with photon-map-based importance sampling, we first trace photons from the light sources to build these maps. Then, we estimate the value of each pixel by tracing a number of camera paths. At each bounce of a camera path, we perform an estimation

of the rendering equation using MIS. When using the one-sample estimator (Equation (3)), we first choose the sampling strategy according to the probability c_i of choosing each of the strategies. When using the multi-sample estimator (Equation (4)), the number of directions to generate using each strategy is given by the n_i values. We generate each direction using the associated strategy, and recursively estimate the radiance arriving from this direction. The final estimation is the sum of the weighted contribution of each direction.

While weighting functions such as the balance heuristic or power heuristic are at the very end of MIS, representativities come up as the first step of the estimation, as it helps choosing amongst the available strategies (Figure 2). Therefore, the representativity of a strategy has to measure the appropriateness of a sampling strategy to evaluate an integrand: the more a sampling strategy reduces variance for the current integrand, the higher its representativity value is.

The sampling configuration should thus reflect the estimation of relevance by assigning probabilities that are function of the representativity of each sampling strategy for a given integrand. This implies that representativities should be unitless comparable values, expressed here between 0 and 1. We design such functions in the context of global illumination. We want to use MIS-based estimators, with $n+1$ possible strategies: either sampling from the BSDF, or sampling from n different photon maps, where n can be arbitrarily large.

All the representativity functions that we now derive have two implicit parameters, which completely describe the integrand when the scene is fixed: the estimation point $\mathbf{x}$ and the outgoing direction ω_o.

3.1 BSDF-based Strategy Representativity

Importance sampling from BSDF is one of the most widely used strategy when simulating global illumination. For this reason, we derive a representativity function for strategies based on a BSDF.

Our representativity function for BSDFs is built upon the directionality of the BSDF, for the given outgoing direction. Indeed, a diffuse BSDF has a low directionality, all directions having the same scattering behavior. Conversely, an almost mirror-like glossy BSDF has a high directionality, since light is scattered only within a tiny cone of directions in the outgoing direction.

In our rendering engine, the BSDF model combines an Ashikhmin-Shirley anisotropic BRDF [5] $AS(\omega_i \rightarrow \omega_o)$ with parameters k_d and k_s, and a specular BTDF $ST(\omega_i \rightarrow \omega_o)$ with parameter k_t, with a Fresnel term $F(\omega_o)$ to weight between BRDF and BTDF:

$$\begin{aligned} f_s(\omega_i \rightarrow \omega_o) \quad = \quad & F(\omega_o)\, AS(\omega_i \rightarrow \omega_o) + \\ & (1 - F(\omega_o))\, ST(\omega_i \rightarrow \omega_o). \end{aligned} \tag{5}$$

This BSDF has three components: the diffuse part does not guide more in any particular direction, the glossy part of the BRDF guides in function of the roughness terms (n_u, n_v) that are similar to Phong exponents [5], and the specular BTDF part guides completely in the unique contributing direction. Thus, the final representativity function is a composition of the directionality of each component.

The directionality of the diffuse part is set to a minimum value, corresponding to the uniform probability to sample any direction: $d_d = 1/(2\pi)$.

The directionality of the glossy part can be estimated by the aperture angle of the cone containing a proportion of the directions generated by importance sampling. This angle can be computed considering that the importance sampling procedure creates directions whose angle to the perfect mirror reflection direction is decreasing as the random number u used to sample this direction increases. We define our directionality by considering the angle θ_h^n obtained for $u = 0.5$ for a given Phong exponent n, meaning that the cone contains half of the generated directions. This does not affect the sampling procedure in itself, which still considers all the contributing directions. We only use this angle for the directionality estimation. θ_h^n is obtained from the importance sampling formula:

$$\theta_h^n = \cos^{-1}\left(0.5^{\frac{1}{n+1}}\right). \tag{6}$$

The anisotropy is handled by taking $n = \max(n_u, n_v)$, because it is the most directional (*i.e.*, narrower), and so, it makes the BRDF more representative.

As a small angle implies a high directionality, we can not directly use the computed angle, but we need to revert it, using the maximal angle that can be obtained ($\cos^{-1}(0.5) = \frac{\pi}{3}$, given by $n = 0$). We also need to ensure that the directionality is at maximum 1, with a minimal value corresponding to diffuse scattering. Combining all the terms, the directionality for the glossy part of roughness n corresponds to:

$$d_s = \frac{1}{2\pi} + \left(1 - \frac{1}{2\pi}\right)\left(\frac{\pi}{3} - \cos^{-1}\left(0.5^{\frac{1}{n+1}}\right)\right)\frac{3}{\pi}. \tag{7}$$

The specular BTDF is the most directional scattering behavior we can have, thus its directionality is maximal: $d_t = 1$.

For more complex BTDFs, such as the microfacet-based model of Walter *et al.* [6], similar derivations as the one used for d_s can be applied.

The final representativity function is obtained by weighting the three directionalities, according to the Fresnel term $F(\omega_o)$ and the normalised version of each component, obtained from the k_d, k_s, and k_t parameters as $\overline{k_d} = k_d/(k_d+k_s+k_t)$, respectively for $\overline{k_s}$ and $\overline{k_t}$. Taking the normalised version of each component ensures that the global albedo of the BSDF is not taken into account, as it affects only the final value of the integral but not its shape. This leads to:

$$R(\text{BSDF}) = F(\omega_o)\,(\overline{k_d}\, d_d + \overline{k_s}\, d_s) + (1 - F(\omega_o))\,\overline{k_t}\, d_t. \tag{8}$$

Note that representativities obtained from this function can be null only if there are no contributing direc-

tions, ensuring that no bias is added. This final representativity function meets all the requirements presented above, as representativities computed from it are always between 0 and 1, and are unitless. Such representativity functions are called *single*.

3.2 Photon-Map-based Strategy Representativity

When computing global illumination, strategies that match the incident radiance part of the integrand can greatly help reducing variance when incident light is highly non-uniform. This is most visible when caustics are present, as in this case the incident radiance term is the most important of the integrand. In our application, we choose to use photon maps to sample incident directions. More specifically, we have several photon maps, each considering a different part of the radiance field (caustics, diffuse indirect lighting, *etc.*). Each of these maps can be sampled, leading to one strategy per map. Consequently, we derive a representativity function that can be used for all these map-based strategies.

Photons that are stored in the sampled maps provide a flux estimation whose value is not limited to the range $[0, 1]$. Moreover, it is not a good absolute measure of interest, as the flux depends on the light sources intensities. However, these flux values can be compared between photon maps, to help choosing amongst maps. We thus introduce a two-level representativity function for each photon map: the first-level representativity function helps choosing between sampling the BSDF term or sampling the incident radiance term. The second-level representativity function helps choosing one particular photon map amongst the available photon maps, and can therefore use all the available physical data.

Representativity function construction:
The representativities obtained from the first-level representativity function have to be unitless and contained between 0 and 1 in order to be comparable with the BSDF representativity. Once computed for each photon map, we combine the first-level representativities to obtain the representativity of all the photon maps at once, gathered in a group. We call it *group representativity*.

The particular first-level representativity function that we now derive has the advantage of being fully and efficiently precomputable. For each map strategy, its first-level representativity is computed from the photons densities. We build an SAH-based kd-tree [9] from the photons in the associated map, with a given maximum number of photons N_{p_max} per leaf. For each leaf l, we estimate its density by computing the ratio $d(l) = n_p(l)/SA(l)$, where $n_p(l)$ is the number of photons in the leaf, and $SA(l)$ is the surface area of the bounding box of the leaf. We use the surface area because photons in the map are distributed on surfaces, and thus we want to keep comparable units (number of points over area). This per-leaf density is then converted to a representativity $r(l)$ by switching to a global probability model based on a gaussian distribution. We avoid the use of $r(l) = d(l)/\max_{l'}(d(l'))$ in order to be robust to very high densities caused by one very small leaf containing photons.

The final leaf representativity is the value of the cumulative distribution function of the global gaussian distribution:

$$r(l) = \mathrm{P}(X \leq d(l)), \;\; X \sim \mathcal{N}(\mu, \sigma). \tag{9}$$

The average μ of the global gaussian distribution is taken as the average density of the non-empty leaves (denoted as μ_d). Its standard deviation σ is computed from the standard deviation of the non-empty leaves densities (denoted as σ_d) and μ_d:

$$\sigma = \min\left(\sigma_d, \frac{\mu_d}{2}\right). \tag{10}$$

This clamping of the standard deviation eliminates non-negligible representativities for large leaves (in terms of surface area) with very few photons in it, whereas they would not be representative at all. Note that this first-level representativity function can be used for any map- or cache-based strategy, by replacing photons by the adequate term in the description above.

The group representativity of g is then defined as the maximum of each first-level representativity in the group:

$$GR(g) = \max_{s \in g}(r(s)) \tag{11}$$

where $r(s)$ is the representativity of the leaf containing the estimation point $\mathbf{x}$ in map $m(s)$ associated to strategy s, or 0 if $\mathbf{x}$ is not contained in $m(s)$. The average or any other combination of the first-level representativities could also be used to compute the group representativity. We choose the maximum to be conservative and to avoid missing a probabilistically very good strategy even though the others in the group are not adapted at all, and thus have very low first-level representativities.

Choosing amongst several photon maps is done thanks to the second-level representativity function. Representativities obtained from this function are called *local representativity*, as opposed to the group representativity. We now derive such a local representativity function for a photon map m. We use the photons in leaf l containing $\mathbf{x}$, and define the local representativity as the average of the potential contribution of each photon p:

$$LR_g(m) = \frac{1}{n_r(l)} \sum_{\mathrm{p} \in l} \left[f_s(\mathbf{x}, \omega_i(\mathrm{p}) \rightarrow \omega_o)\, w(\mathrm{p})\, k(\mathbf{x}, \mathrm{pos}(\mathrm{p})) \right] \tag{12}$$

where $\omega_i(\mathrm{p})$ is the photon's incident direction, $w(\mathrm{p})$ is the photon's weight, and $k(\mathbf{x}, \mathrm{pos}(\mathrm{p}))$ is a kernel value based on the distance between the photon's position and the estimation point $\mathbf{x}$.

Representativity function usage:
When estimating an integral at $\mathbf{x}$ in a scene, we find the leaf containing $\mathbf{x}$ in the map associated to the strategy. This corresponds to descend in the kd-tree. The first-level representativity is the leaf's representativity, as defined by Equation (9). If point $\mathbf{x}$ is outside the kd-tree's

global bounding box, the first-level representativity of the strategy using this map is set to 0. This computation does not add noticeable overhead compared to user-defined sampling configurations. The final group representativity is the combination of each strategy's first-level representativity, using Equation (11). The local representativity is obtained for each map by using Equation (12).

At this moment, one could argue that we have one parameter in the first-level representativity function we have designed: the maximum number of records per leaf N_{p_max}. However there are major differences between a user-defined sampling configuration and this parameter. First, N_{p_max} does not vary within a scene, and in practice, it does not vary between scenes either, but it is affected by the number of photons in a map. N_{p_max} is a tradeoff between the resolution of the representativities over the scene on one hand, and the accuracy of density estimation on the other hand. A larger N_{p_max} value leads to larger leaves, thus less-varying representativities. The more points there are, the more accurate density estimation is for uniform zones, but it does not adapt well to rapid density variations, typical of caustic effects for instance. In all our tests, N_{p_max} has been taken as the minimum of $n_p(m)/10000$ and 100 ($n_p(m)$ being the number of photons in map m), without any special tuning, the range of values producing good results being quite large in practice.

3.3 Sampling Configurations from Representativities

Grouping the photon maps can be generalised: all strategies which rely on absolute values should be clustered together, first- and second-level representativity functions being created for them. This leads to situation similar to the one depicted in Figure 3. Different sets of strategies can be created. The first one, $\mathcal{S}$, contains all the strategies whose representativity function is single, as the BSDF sampling strategy. The set $\mathcal{G}$ contains all the groups created for two-level representativity functions. There is one such group photon maps, and radiance cache would add another group.

We now consider a fixed integrand. Once we have computed the representativities $R(s)$ for the strategies in set $\mathcal{S}$, the group representativities $GR(g)$ for the set $\mathcal{G}$ of groups g, and the local representativities for the strategies in the groups $LR_g(s)$, we can compute the strategy sampling probability $p(s)$ for all these strategies.

Letting

$$\begin{aligned} \text{norm} &= \sum_{g\in\mathcal{G}} GR(g) + \sum_{s\in\mathcal{S}} R(s) \\ \overline{GR}(g) &= GR(g)/\text{norm} \\ \overline{LR_g}(s) &= \frac{LR_g(s)}{\sum_{s'\in g} LR_g(s')} \end{aligned}$$

the final probability $p(s)$ of strategy s belonging to group g is finally given by:

$$R(s) = \begin{cases} R(s)/\text{norm} & \text{if } s \in \mathcal{S} \\ \overline{GR}(g) \times \overline{LR_g}(s) & \text{otherwise.} \end{cases} \quad (13)$$

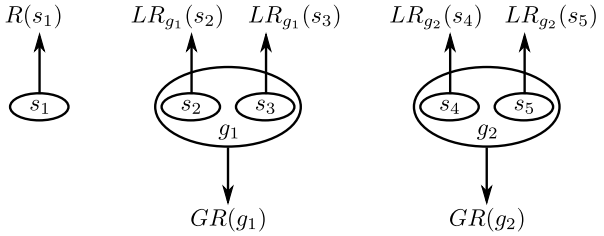


Fig. 3. Example of a situation where a total of five sampling strategies are available, with one strategy whose representativity function is single (s_1), and two groups, each containing two strategies with two-levels representativity functions (s_2 and s_3 in g_1, s_4 and s_5 in g_2).

As an example, consider the situation depicted in Figure 3. For this situation, letting $\text{norm} = GR(g_1) + GR(g_2) + R(s_1)$, probabilities are

$$p(s_1) = \frac{R(s_1)}{\text{norm}} \quad (14)$$

$$p(s_2) = \underbrace{\frac{GR(g_1)}{\text{norm}}}_{\overline{GR}(g_1)} \times \underbrace{\frac{LR_{g_1}(s_2)}{LR_{g_1}(s_2) + LR_{g_1}(s_3)}}_{\overline{LR_{g_1}}(s_2)} \quad (15)$$

and $p(s_3)$, $p(s_4)$, and $p(s_5)$ have a similar expression as $p(s_2)$.

These probabilities can be directly used with the one-sample estimator, keeping it unbiased as long as the representativity of a strategy is not 0 if this strategy can generate at least one contributing sample. In the case of a multi-sample estimator, a sufficient but not required way to ensure unbiasedness is to have a special strategy s_c (c for *complete*) that can generate any such sample, and ensure that the number of samples n_c assigned to s_c is at least one. In the context of rendering, the BSDF sampling strategy is a very good candidate for being a complete strategy. Nevertheless, we must also ensure that the distribution of n_i still follows as much as possible the probability distribution given by all p_i when assigning systematically at least one sample to s_c. This implies changing the probabilities of strategy s_c (originally given by p_c) and of all other strategies p_i, giving new probabilities p_i^t, (t for *temporary*) to maintain the expected value $E[n_i] = p_i \times N$ for each strategy:

$$p_c^t = \max\left(\frac{(p_c \times N) - 1}{N - 1}, 0\right) \quad (16)$$

$$p_i^t = \frac{p_i \times N}{N - 1} \quad \text{if } s_i \neq s_c. \quad (17)$$

As is, $p_c^t + \sum_i p_i^t$ can be larger than 1.0 if $((p_c \times N) - 1)/(N - 1)) < 0$. We thus normalize the probabilities, leading to the final probabilities actually used to compute the number of samples assigned to each strategy:

$$p_c^f = \frac{p_c^t}{p_c^t + \sum_i p_i^t} \quad \text{and} \quad p_i^f = \frac{p_i^t}{p_c^t + \sum_i p_i^t}. \tag{18}$$

To obtain each n_i (including n_c) while ensuring unbiasedness, we start by setting $n_c = 1$ and all other n_i to 0. We then sample $N - 1$ times the probability distribution defined by all p_i^f (including p_c^f), selecting each time a strategy s_i. Each time a strategy s_i is chosen, its associated n_i is increased by one.

3.4 General Hints for Defining Representativities

Similarly to Metropolis mutations [14], a representativity function is an observational model, whose quality affects the rate of convergence of estimators. Ideal representativity functions should have the two following properties:

- be proportional to the relevance of informations locally available for the strategy,
- be computed using only data from the strategy or the group of strategies it represents. The normalisation is the only operation that considers all strategies at once.

For strategies with two-level representativity functions, second-level representativity functions should use as much information as possible to favor strategies that are better than other strategies within the same group.

4 Numerical Analyses

We performed numerical analysis to assess the robustness brought by our method. For a number of very different cases, we compare the behavior of estimators obtained with our method to static sampling configurations. These cases are specifically designed to cover a wide range of common situations in rendering.

We used the photon map guided path-tracing system described in Section 5.2 to perform the tests, because its unbiasedness ensures that tests based on reference averages criteria are meaningful, such as the mean square error (MSE). All the estimators use the balance heuristic to obtain the final estimation value for a sample. Three strategies are available in our test implementation: sampling using a BSDF, sampling using a diffuse indirect map, and sampling using a map for specular paths.

Each optical situation leads to a different integrand to evaluate. For each integrand, 11 estimators have been considered: 10 test estimators using an increasing probability ρ_b of sampling a BSDF, and our automatic estimator. The test estimators we have chosen allow us to cover a wide range of possible sampling configurations, including the uniform one recommended by Veach and Guibas [2]. Each estimator can be near-optimal for an integrand, but behave poorly on others. For each integrand, we can then compare the best MIS estimator with our adaptive estimator.

For the test sampling configurations, the probabilities ρ_b to sample according to the BSDF range from 0.1 to 1.0 by steps of 0.1. The map-based strategies probabilities are computed as follows: if both maps are present, it is $(1 - \rho_b)/2$ for each one, otherwise it is $(1 - \rho_b)$ for the available map. If no maps are present at point $\mathbf{x}$, we set $\rho_b = 1$. Note that this is already a sort of adaptation to local estimation, but it is simple enough to be implemented in a basic photon-map-based path-tracer. The uniform sampling configuration is closely approximated by the case $\rho_b = 0.3$ when both maps are present, and is represented exactly by $\rho_b = 0.5$ when only one map is present.

Figures 5 to 10 present the numerical results we use to evaluate the efficiency of our method. They are available in a larger format in the supplemental material. Each curve in these figures corresponds to the MSE of an estimator for an increasing number of samples, from 2 to 1024, used to estimate the final value. Let E_n be an estimator using n samples, $\Theta(E_n)$ the random variable associated to the luminance of an estimation made by E_n, and $\hat{\theta}$ the luminance of the reference value computed with path-tracing. The MSE of E_n is computed as

$$\text{MSE}(E_n) = \text{Var}(\Theta(E_n) - \hat{\theta}). \tag{19}$$

As our test scenes exhibit only one main color, luminance can be safely used. Nevertheless, extending this metric to color samples is straightforward.

In each case, the reference value has been computed as the average of 16 unbiased estimations computed with 2^{16} samples each, using a photon-map guided path-tracer with good sampling probabilities for each specific integrand, hand-tuned to reduce the variance of the final estimate. To avoid numerical instability, we did not directly use a 2^{20} samples estimation. The effective value of $\text{MSE}(E_n)$ has been estimated by running m times E_n and computing the variance using the numerically-stable Knuth's algorithm [23]. To avoid too long computations for no practical gain in precision, m is decreasing as n is increasing, since the variance of the sequence of estimators (E_n) should decrease as n increases.

4.1 Simple Scenes

Figure 4 presents the simple scenes that were used to study the behavior of estimators obtained by using our method. We made this study for different integrands in well-controled conditions and at a location where the estimation of the variance can be computed with high accuracy. To achieve this, we estimate the LTE (Equation (1)) with at maximum one indirect bounce. These scenes have been constructed to have no direct illumination at the estimation point, and analytical solutions for indirect bounces. Even if very simple, these scenes can lead to arbitrary high variance when using an inadequate sampling configuration.

Scene (4, *a*) description: L_1 is a spotlight and L_2 is an area light source. W_1 is a highly glossy wall, and W_2 is a perfectly specular glass wall. Both walls scatter light coming from respectively L_1 and L_2 toward the floor F. The floor's material is described by two parameters

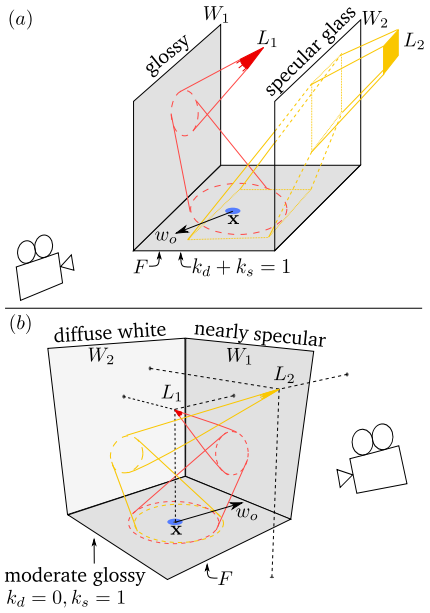


Fig. 4. The configuration of the two simple test scenes. Light sources are in red and yellow, the blue circle at the middle of the floor indicates the estimation point.

(k_d, k_s), with $k_d + k_s = 1$. k_d is the coefficient for the diffuse part, and k_s is the one for the glossy part. In the BRDF used for this scene, the glossy part is nearly perfect specular. The camera's position has been chosen so that the k_s part does not scatter any light directly to the camera through the observed point on floor F, all the energy coming from the k_d part. Increasing k_s leads to higher variance when using the BSDF strategy with a high probability. This variance can be made arbitrarily high for pure path-tracing. Note that in scene (4, *a*), for each configuration each light source power has been adapted to keep final values in the same order of magnitude (between 0.1 and 1.0).

Scene (4, *b*) description: It is composed of two spotlights L_1 and L_2, two walls and a floor. W_1 is a nearly perfect specular reflective wall, W_2 is made of a diffuse white material, and F is a moderately glossy floor (no diffuse component, *i.e.*, $k_d = 0$ and $k_s = 1$). F is not highly specular so that directions generated using photons coming from W_2 can have a non-null contribution. In this scene, all the light coming from L_1 does not contribute to the final value, since F's material does not scatter energy to the camera's direction. Thus, only a part of the light coming from L_2 and scattered by W_2 contributes to the final value. Here, the map-based strategy can lead to a high variance if the scattered energy coming from L_1 is much larger than the one coming from L_2. The higher the difference, the higher the variance, until reaching an upper bound given by the probability c_m to use the diffuse indirect map strategy. As a matter of fact, using pure BSDF-based path-tracing on this scene results in low variance. Provided there is a large difference of scattered energy in favor of L_1, the variance of the final estimation can be made arbitrarily high by increasing c_m. Thus, poor sampling configurations probabilities can lead to arbitrary high variance.

Tests setup: Each of the two scenes has parameters that can be set to make some estimators exhibit arbitrary high variance. The MSE (Equation (19)) of the LTE estimation has been computed for different values of these parameters. Figure 5 presents the results obtained for one-sample estimators, and Figure 6 the results obtained for multi-sample estimators. For scene (4, *a*), the k_s coefficient of the floor has been changed ($k_s = 0.0, 0.4, 0.8$) leading to a rapid increase in variance for "BSDF-oriented" strategies. For scene (4, *b*), the emitted intensity of L_2 is increased ($L_2 = 1, 100, 10000$, for each component of the spectrum). Note that in scene (4, *b*), the emitted spectrum of L_1 is 100 in all cases. For each scene variation, multi-sample estimators are obtained by the method described in Section 3.3, with N (the total number of samples) set to 16.

Discussion: We explicitly compute only the lighting caused by one indirect bounce, and the LTE restricted to direct lighting can be solved analytically thanks to the presence of Dirac functions, brought either by the spotlight, or by the specular transmission. This allows us to compute a reliable estimation of each estimator's variance, as there is only one LTE solution to compute using an MC estimator, at the point seen by the camera. This is this estimator's variance that we estimate here.

These simple scenes lead to very high variance when a non-adapted sampling configuration is used, and in each case the optimal sampling configuration is different. Figures 5 and 6 show that estimators with test sampling configurations have important variations in variance, meaning that they are not robust. Meanwhile, the sampling configurations obtained by using our representativity method leads to estimators with a low variance in every situation. The results confirm what was speculated above, and assess that estimators obtained using our method are robust. As a matter of fact, even if not the best, there are no estimators that behave consistently better than our estimator in all the cases.

When using one of these test configurations for a whole scene, there would be pixels with very low variance, but also pixels with very high variance. Using our method would lead to homogeneous results, with a rather low variance each time. This automatic robustness is one of the key advantage brought by our method. To further test the robustness of our method in very difficult cases, Figure 7 shows the MSE of the estimators obtained when increasing c_m in scene (4, *b*), with L_2's emitted intensity set to 10000. We can see that our method automatically leads to estimators whose variance does not vary much over the scenes, additionally keeping it low.

4.2 Chains of Estimators

For a more complete study, path-tracing-like estimations have been performed on specific pixels of one scene,

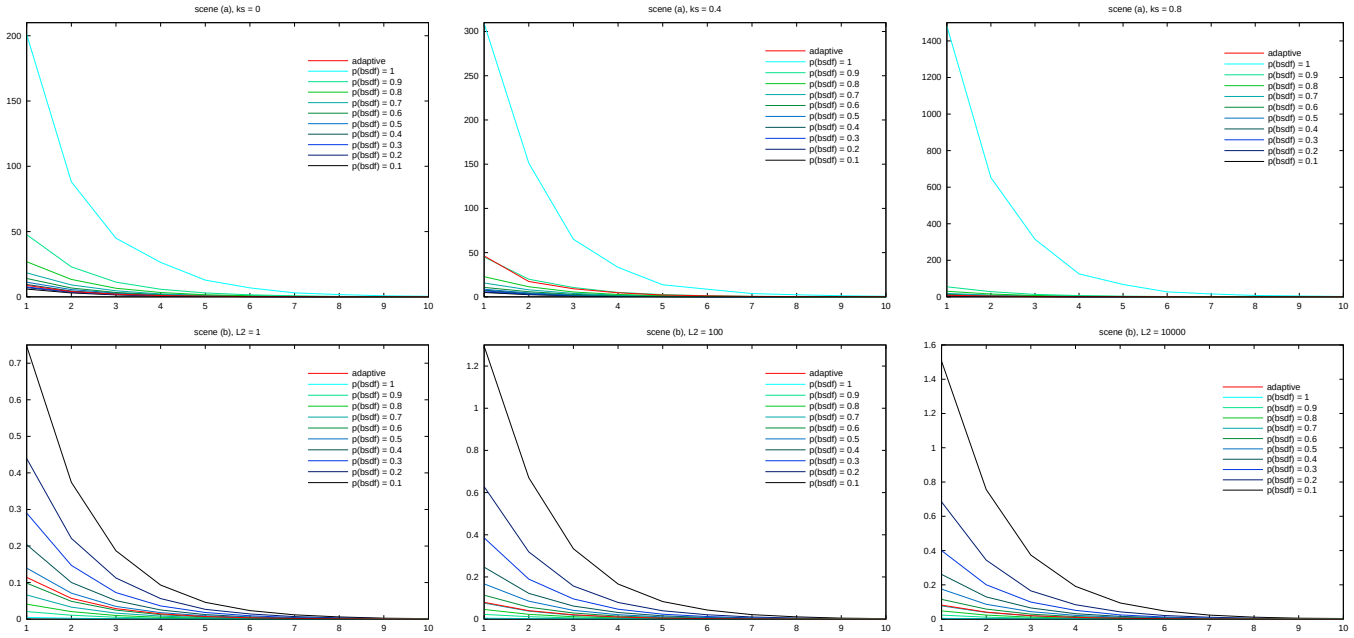


Fig. 5. MSE of one-sample estimators, with a number of test configurations estimators, and the estimator obtained by using our representativity method in bold red. Results for scenes (4, *a*) and (4, *b*) are presented respectively in the top and bottom row. The x axis corresponds to the number of samples generated to perform one estimation of the LTE, from 2^1 to 2^{10}. The important thing to note is that no sampling configuration has consistently a better variance than ours. There are better configurations for each case, but these more adapted configurations change for every test.

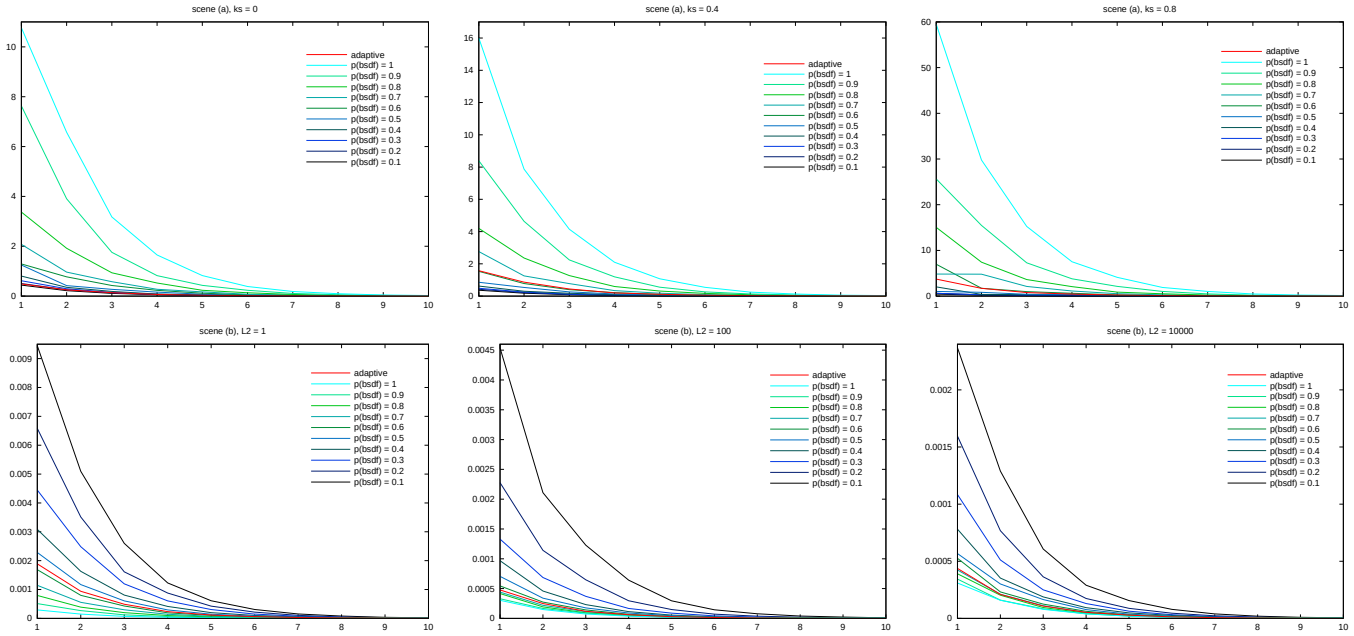


Fig. 6. MSE of multi-sample estimators, with the same conditions and caption as in Figure 5, page 9.

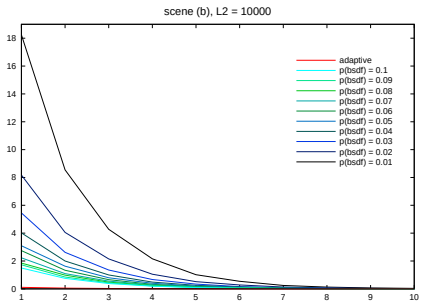


Fig. 7. MSE of one-sample estimators with very low c_m probabilities (in $[0.01, 0.10]$), and MSE of our estimator in bold red. Same details as in the caption of Figure 5.

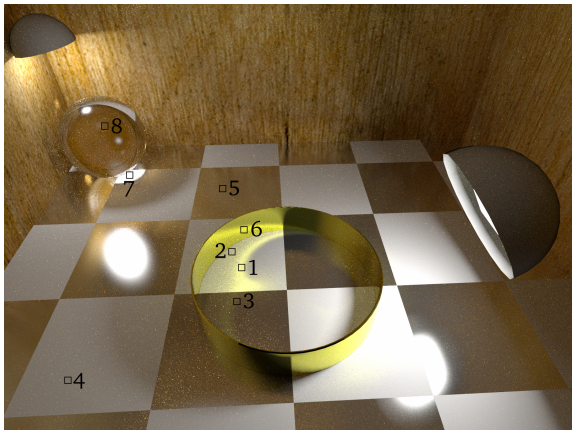


Fig. 8. The test scene for chains of estimators, with labeled test pixels, corresponding to very different optical situations. The floor has a checkerboard pattern composed of diffuse cream tiles and glossy tiles. The ring is glossy with a large Phong-like roughness coefficient, and the sphere is a Fresnel weighted sum of a glossy reflection part and a specular transmission part.

shown in Figure 8. This scene features many different optical configurations, involving specular and glossy caustics, diffuse and specular scatterings, strong indirect illumination, etc. Each test pixel features one (and sometimes more) specific lighting situation, thus leading to very different integrands at each step of the path-tracing algorithm. The goal is to examine the variance of the estimation for these pixels when using chains of estimators derived from our method, compared to chains of test estimators (each c_i being the same for all the estimators in a chain). Even if the number of test pixels can seem low, the results are representative of most lighting situations in any scene, and allow for careful and complete study. An additional series of tests over more than 6000 random pixels confirms the conclusions drawn from our chosen pixels.

Each test pixel has been chosen carefully, in order to control the lighting situations and to get meaningful results:

- Location 1 is a very hard case. At the first bounce (at the point seen by the camera), the diffuse BSDF scatters light coming from all directions, but incident lighting is strongly directional: it comes from the left (ring's caustic). For a direction sampler, it should sample the diffuse indirect map at the first bounce, and then the BSDF at the second bounce, since the ring's BSDF is highly glossy.
- Location 2 is a similar case, but with much lower caustic intensity, thus sampling the BSDF and sampling the indirect diffuse map are both a good choice, each one sampling different effects.
- Location 3 features a highly glossy reflection with a low diffuse part, with strong indirect illumination provided by the ring's caustic. This situation can lead to arbitrary high variance for pure BSDF-based path-tracing when no incident lighting comes from the reflection directions. As a matter of fact, the lower the diffuse part is relatively to the glossy part, the less it is sampled, and thus the less the only contributing directions are generated.
- Location 4 is a case of diffuse surface with low frequency indirect illumination, which is encountered in nearly every scene.
- Location 5 is a similar common case of a glossy surface with low frequency indirect illumination.
- Location 6 is a trickier version of Location 1. It adds one bounce to the "optimal" sampling chain: at the first bounce, the BSDF-based strategy should be used, at the second, the diffuse-indirect-map-based strategy should be used, and at the third, the BSDF-based strategy should again be used.
- Location 7 is a classic case of specular caustic.
- Location 8 is a classic case of specular transmission, where only the BSDF-based strategy can provide contributing directions because the Fresnel term is null, thus there are no glossy reflections.

For representativity-based sampling configurations, the two strategies using the maps are gathered in one group. The BSDF-based strategy is a single strategy using the representativity function defined in Section 3.1. Figure 9 presents the probabilities computed by our method for each of the three strategies for the observed points of the scene. This defines a single estimator, which depends on $\mathbf{x}$ and ω_o.

The goal of these test pixels is to show that for all these situations where test configurations can fail, our adaptive approach performs well. We might not be as good as the best fixed configuration for a given optical situation, but the best fixed configuration is different for each such situation, and therefore an adapted sampling configuration for one illumination situation can give very poor results in another situation, while our approach still gives a good configuration.

Discussion: Figure 10 shows the MSE obtained for each test pixel labeled in Figure 8. Our estimator gives better results than any test estimator in Location 1. The

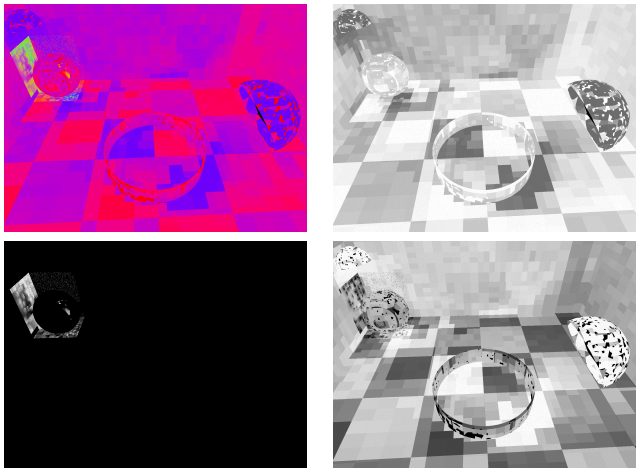


Fig. 9. Probabilities (as pixel intensities) assigned by our method to each strategy for the visible points in the scene. Probabilities of sampling the BSDF are displayed in the top-right image, the ones for sampling the specular map in the bottom-left, and sampling the diffuse indirect map in the bottom-right. The top-left image presents a summary, encoded in the three RGB channels: R for sampling the BSDF, green for sampling the specular map, and B for sampling the diffuse indirect map. Note how sampling the BSDF is favored on the glossy tiles, sampling the diffuse indirect map is favored for the ring caustic on the diffuse tiles, and sampling the specular map is favored under the sphere and on the wall, a specular caustic being created by the right light source. The indirect map contains 1,000,000 photons, and the specular map 100,000 photons, N_{r_max} was set to 100.

glossy caustic is well sampled by the diffuse indirect map, but using this map to sample a direction on the almost-mirror-like glossy ring leads to high variance. Conversely, pure-BSDF path-tracing (cyan curve) is well adapted for the ring, but not for the caustic. Our method favors the strategy using the map for the caustic, and the strategy using the BSDF on the ring. Location 3 is our worst case. The specular map strategy is not available as there are no specular paths in this part of the scene (they are gathered in the sphere's region), thus only the group representativity of the indirect map strategy is taken into account. Both the BSDF and indirect diffuse map strategies have high representativities, but sampling the BSDF would be the most appropriate strategy.

There are several important facts to note about these graphs:

- Our worst case (Location 3) still has a lower MSE than some test sampling configurations, even if the majority of estimators based on test configurations have a better behavior than our estimator.
- The best configurations for Location 1 are the ones with lower ρ_b, configurations with high ρ_b performing very poorly. Conversely, the best configurations for Location 3 are the ones with higher ρ_b, the other ones performing very poorly.
- On the eight test cases, our configuration is good to very good for five of them (Locations 1, 2, 4, 5, and 6), average for one of them (Location 8), and relatively poor for two of them (Locations 3 and 7).
- None of the test configurations perform well for all pixels, and none are consistently better than our representativity-based configuration.

In terms of computation time, there is no noticeable overhead compared to a path-tracer relying on photon maps to guide the sampling, whatever sampling configuration it uses (uniform or the tests ones).

5 Examples of Applications

5.1 Photon Mapping Final Gathering

Using our method to perform final gathering is straightforward. The first-level representativities of all available maps are precomputed (we can think of different kinds of indirect maps, for instance separating glossy and non-glossy scatterings). At each gathering point, local representativities of each strategy are computed, and the base probability p_i for each strategy s_i is obtained using Equation (13).

We then use the multi-sample estimator with N samples, by distributing the N gathering rays over the strategies, as described in Section 3.3, assigning at least one sample to the BSDF sampling strategy to conservatively ensure unbiasedness, as we know that this strategy can always sample any contributing direction.

5.2 Photon-map Guided Path-tracing

Path-tracing with next event estimation can be interpreted as a recursive process rather than an integration method over a path space, where we evaluate at each level of recursion the direct and indirect parts of the rendering equation, using only one sample for the indirect part. In this formulation, the estimation of the indirect part can be computed by the one-sample estimator.

Using our representativity-based model, we compute the probabilities p_i to choose any of the available strategies. As the BSDF representativity, defined in Section 3.1, is non-zero for any non-completely absorbing BSDF, unbiasedness is ensured.

5.3 Direct Lighting in Highly Occluded Environment

Dedicated techniques exist to handle this problem [24]. However, it is possible to use the same representativity functions as the one defined for photon maps in Section 3.2 to construct robust estimators for the direct lighting part of the rendering equation. These representativities would allow us to sample the most appropriate light sources even when there is a large number L of small light sources in a highly occluded environment. These estimators rely on $L+1$ sampling strategies: one strategy per light source – consisting in sampling a point on the light source –, and the strategy that samples

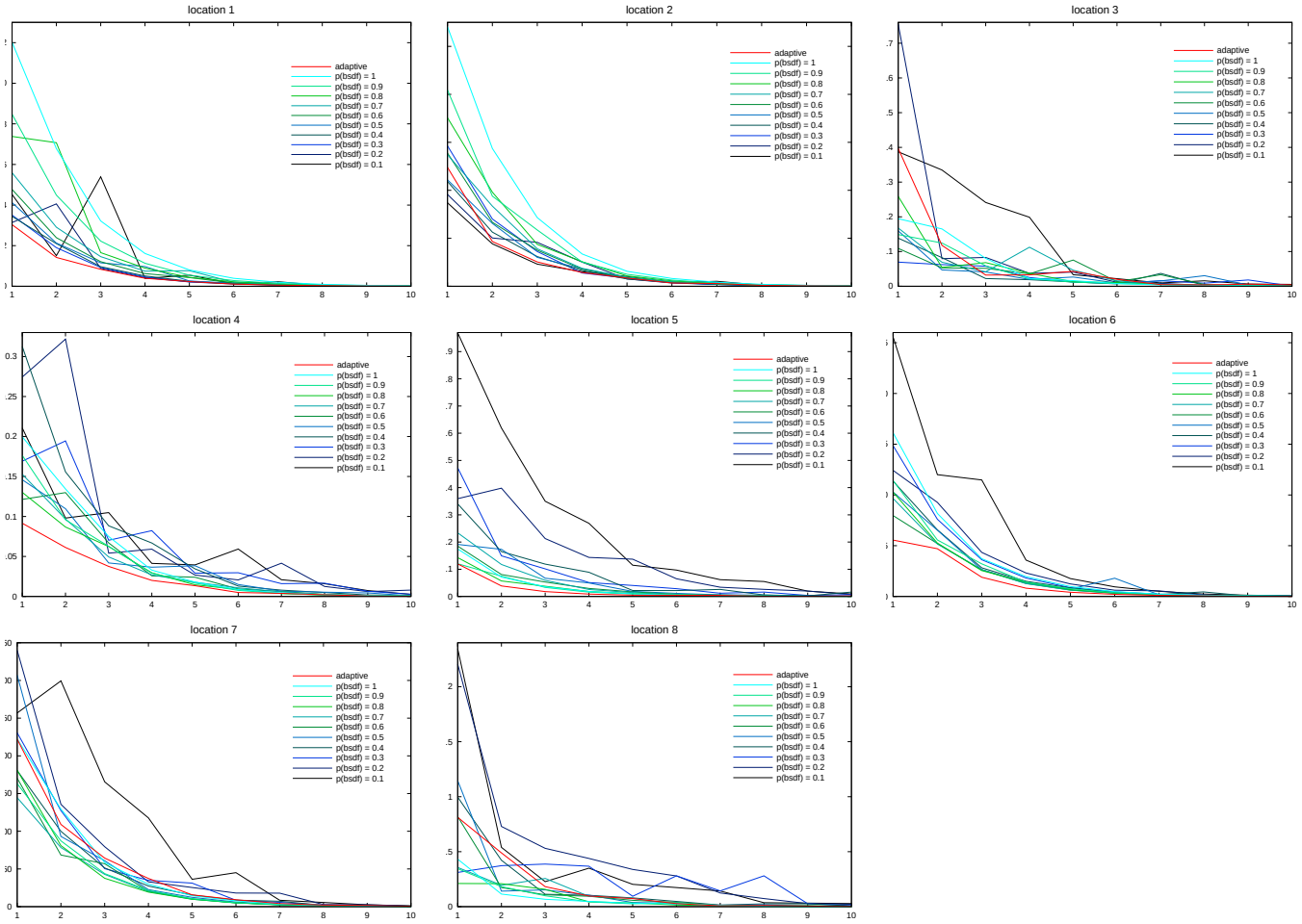


Fig. 10. For each location of the test scene of Figure 8, MSE of our estimator are in red, and MSE of the other estimators are with a color gradient ranging from dark blue to cyan. Dark blue means $\rho_b = 0.1$, cyan means $\rho_b = 1.0$, that is pure BSDF-based path-tracing. The x axis corresponds to the number of samples generated to perform one estimation of the value of a pixel, from 2^1 to 2^{10}. MSE has been computed using a variable number of estimations. For power i, it is given by $\min(10^4, 2^{10-i} \times 10^3)$. The number of estimations is clamped to avoid numerical instability.

the BSDF. All light-based strategies are clustered in one group. By creating one small photon map containing only one-bounce photons per light source, the light strategies mostly have null probability at point $\mathbf{x}$ because of occlusion. The maps containing photons at $\mathbf{x}$ indicate which light sources contribute most to $\mathbf{x}$, accordingly affecting the strategies probability.

5.4 Other Contexts than MIS-based Estimation

Our representativity-based method can also be used in different contexts than MIS-based estimation, by building a pdf taking into account all strategies, making it for instance usable by any algorithm requiring to sample directions, such as bidirectional path-tracing [25], [26], Metropolis light transport [27], etc. This pdf can be used for samples of any type for which several importance sampling strategies are suitable (directions, points, paths, etc.). For sample x (x being a direction, a point on a surface, a path, etc.), this pdf is defined by:

$$\mathrm{PDF}(x) = \sum_{i=1}^{N_s} p(s_i)\,\mathrm{PDF}_i(x) \tag{20}$$

where N_s is the total number of strategies, $p(s_i)$ is the probability of strategy s_i computed from its representativity, and PDF_i is the pdf associated to s_i.

Note that strategies can be ignored when creating an estimator. For instance for bidirectional path-tracing, this ability allows us to only consider strategies using importance-based maps when creating light subpaths, and strategies using energy-based maps when creating camera subpaths.

6 Conclusion

Multiple importance sampling (MIS) is a general and efficient way to perform integral estimation of complex functions using simple importance sampling strategies. Although optimal weighting functions exist, MIS does

not provide any information about how to distribute samples over sampling strategies. To address this issue, we developed an intuitive approach, relying on the notion of *representativity* of a sampling strategy. From a measure of the relevance of a sampling strategy for a given integrand, we derived unbiased estimators for the one-sample and multi-sample models of MIS. These estimators use importance sampling to generate samples, but also importance sampling on the sampling strategies themselves, as we try to favor the strategies that are the most adapted to the given integrand. This corresponds to a form of meta-importance sampling.

We presented rendering-specific representativity functions for BSDFs, maps and caches, with negligible computational overhead. After providing suggestions on how to derive good representativity functions, we then described examples of uses, first in two canonical rendering algorithms, each one using a different estimator from the MIS framework, and for direct lighting in highly occluded environments. We showed that our representativity-based method can be used in any algorithm where several importance sampling strategies are available. Finally, we studied numerical results using a path-tracing-based algorithm in order to validate our approach. We showed that the estimator derived with our method is more robust to a wide range of optical configurations than different estimators, each one based on fixed hand-tuned parameters.

The application we propose of this automatic and robust method to rendering relies on observational representativity functions. An intuitive knowledge of the underlying lighting effects we want to simulate helps defining better representativity functions. Consequently, the robustness of the estimator built upon the defined representativity functions is greatly improved. As representativities should be a measure of the relevance of locally available informations, information theory tools should help to develop more adapted representativity functions, and maybe provide generic tools to automatically define such functions for any kind of strategy.

APPENDIX

Variance-optimal Sampling Configurations:
Our goal is to examine the possibility to use a formal approach in order to find variance-optimal sampling configurations.

We want to solve the following integral, using the Monte-Carlo (MC) method:

$$I = \int_{\mathcal{D}} f(x)dx. \tag{21}$$

The MC method evaluates I using random samples distributed over $\mathcal{D}$ following a probability density function (pdf) p. It is based on the following unbiased estimator of I using N samples:

$$F_= \frac{1}{N}\sum_{i=1}^{N}\frac{f(X_i)}{p(X_i)}, \tag{22}$$

where $\{X_i\}$ are N independent random variables identically distributed according to p. Each tuple (f, p, N) defines an estimator, and setting one or two elements of this tuple defines a family of estimators.

When f is complex, we can use a family of estimator defined in MIS [2] to reduce variance:

$$F_{ms} = \sum_{i=1}^{S}\left[\frac{1}{n_i}\sum_{j=1}^{n_i} w_i(X_{i,j})\frac{f(X_{i,j})}{p_i(X_{i,j})}\right] \tag{23}$$

where S is the number of strategies that can be used, n_i is the number of samples to generate using strategy s_i, w_i is its weighting function , p_i is its associated pdf, and each $X_{i,j}$ is a random variable distributed according to p_i, independent from the other $X_{i,j'}$, $j \neq j'$. Estimators of F_{ms} are called *multi-sample estimators*.

As with an MC estimator, each tuple $(f, \{p_i\}, \{n_i\}, \{w_i\})$, $i \in \{1, \ldots, S\}$ defines an estimator. Optimal weighting functions $\{w_i^*\}$ with respect to the variance $\mathrm{Var}\,(F_{ms}N)$ are solutions of the problem

$$\{w_i^*\} = \min_{\{w_i\}} \mathrm{Var}\,(F_{ms}N)\,. \tag{24}$$

This problem has not been solved exactly, but a very good approximate solution w_i^m is the balance heuristic [2]:

$$w_i^m(x) = \frac{n_i\, p_i(x)}{\sum_{k=1}^{S} n_k\, p_k(x)}. \tag{25}$$

To get still a better variance-wise estimator, it would be necessary to minimize with respect to $\{n_i\}$, *i.e.*, solve

$$\min_{\{n_i\}} \mathrm{Var}\,(E_{bh,ms}[\{n_i\}]) \tag{26}$$

where $E_{bh,ms}[\{n_i\}]$ is the estimator obtained by fixing the $\{n_i\}$ to the values given between brackets. This problem is a constrained discrete optimization problem, which is known to belong to the NP-complete complexity class, and thus cannot be efficiently solved [28].

The family of one-sample estimators present in the MIS framework could help going further:

$$F_{os} = w_i(X)\frac{f(X)}{c_i p_i(X)}. \tag{27}$$

An optimal variance-wise weighting function is a slightly modified version of the balance heuristic w_i^m (Equation (25)):

$$w_i^o(x) = \frac{c_i\, p_i(x)}{\sum_{k=1}^{S} c_k\, p_k(x)}. \tag{28}$$

$\{w_i^o\}$ are the exact solution of the problem $\min_{\{w_i\}} \mathrm{Var}\,(F_{os})$. The family of one-sample estimators using the balance heuristic as weighting functions is denoted as $E_{bh,os}$.

Finding the estimator with minimal variance in $E_{bh,os}$, thus solving the problem

$$\min_{\{c_i\}} \mathrm{Var}\,(E_{bh,os}[\{c_i\}]) \tag{29}$$

is a continuous optimization problem, in theory easier to tackle than the minimization presented in Equation (26). Despite this, directly solving the problem in Equation (29) is not a viable option. First, it requires to compute accurate values of $\mathrm{Var}\left(E_{bh,os}[\{c_i\}]\right)$. This implies a large number of estimations of I using $E_{bh,os}[\{c_i\}_j]$, at each step j of the optimization process, which is very costly. Moreover, the average of all these estimations can already give a very good estimation of I, making the minimization useless. Furthermore, as the computation of the variance is itself an estimation with an inner variance, results obtained for a given sampling configuration could be far from the real value, hence misguiding the optimization process to miss the correct solution.

REFERENCES

[1] J. T. Kajiya, "The rendering equation," in *SIGGRAPH '86*, 1986, pp. 143–150.

[2] E. Veach and L. J. Guibas, "Optimally combining sampling techniques for monte carlo rendering," in *SIGGRAPH '95*, 1995, pp. 419–428.

[3] E. P. Lafortune and Y. D. Willems, "Using the modified phong reflectance model for physically based rendering," Tech. Rep. CW197, 1994.

[4] J. Lawrence, S. Rusinkiewicz, and R. Ramamoorthi, "Efficient BRDF importance sampling using a factored representation," in *SIGGRAPH '04*, 2004, pp. 496–505.

[5] M. Ashikhmin and P. Shirley, "An anisotropic phong light reflection model," *Journal of Graphics Tools*, vol. 5, pp. 25–32, 2000.

[6] B. Walter, S. R. Marschner, H. Li, and K. E. Torrance, "Microfacet models for refraction through rough surfaces," in *EGSR '07*, 2007, pp. 195–206.

[7] S. Agarwal, R. Ramamoorthi, S. Belongie, and H. W. Jensen, "Structured importance sampling of environment maps," in *SIGGRAPH '03*, 2003, pp. 605–612.

[8] V. Ostromoukhov, C. Donohue, and P.-M. Jodoin, "Fast hierarchical importance sampling with blue noise properties," in *SIGGRAPH '04*, 2004, pp. 488–495.

[9] M. Pharr and G. Humphreys, *Physically Based Rendering: From Theory to Implementation*. Morgan Kaufmann Publishers Inc., 2004.

[10] H. Jensen, "Global illumination using photon maps," in *EGWR '96*, 1996, pp. 21–30.

[11] H. W. Jensen, "Importance driven path tracing using the photon map," in *EGWR '05*, 1995, pp. 326–335.

[12] H. Hey and W. Purgathofer, "Importance sampling with hemispherical particle footprints," in *SCCG '02*, 2002, pp. 107–114.

[13] M. Pharr, "Extended photon map implementation," http://www.pbrt.org/plugins/exphotonmap.pdf.

[14] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equations of state calculations by fast computing machines," *Journal of Chemical Physics 21*, 1953.

[15] J. F. Talbot, "Importance resampling for global illumination," Master's thesis, Brigham Young University, 2005.

[16] D. Burke, A. Ghosh, and W. Heidrich, "Bidirectional importance sampling for direct illumination," in *EGSR '05*, 2005, pp. 147–156.

[17] P. Clarberg, W. Jarosz, T. Akenine-Möller, and H. W. Jensen, "Wavelet importance sampling: Efficiently evaluating products of complex functions," in *SIGGRAPH '05*, 2005, pp. 1166 – 1175.

[18] P. Clarberg and T. Akenine-Möller, "Practical Product Importance Sampling for Direct Illumination," in *Eurographics '08*, 2008, pp. 681–690.

[19] R. Wang and O. Akerlund, "Bidirectional importance sampling for unstructured illumination," *Eurographics '09*, pp. 269–278, 2009.

[20] F. Rousselle, P. Clarberg, L. Leblanc, V. Ostromoukhov, and P. Poulin, "Efficient product sampling using hierarchical thresholding," in *CGI '08*, 2008, pp. 465–474.

[21] C. Lemieux, *Monte Carlo and Quasi-Monte Carlo Sampling*, 2009.

[22] A. Keller, "Quasi-monte carlo methods in computer graphics," *Zeitschrift fur Angewandte Mathematik und Mechanik*, vol. 76, no. 3, pp. 109–112, 1996. [Online]. Available: citeseer.ist.psu.edu/article/heinrich94quasimonte.html

[23] D. E. Knuth, *The art of computer programming : semi-numerical algorithms*, 1998, vol. 2, ch. 3, p. 232.

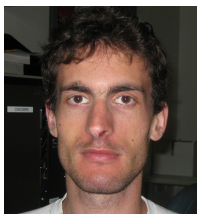
[24] M. Donikian, B. Walter, K. Bala, S. Fernandez, and D. P. Greenberg, "Accurate direct illumination using iterative adaptive sampling," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 3, pp. 353–364, 2006.

[25] E. Veach and L. J. Guibas, "Bidirectional estimators for light transport," in *EGWR '94*, 1994, pp. 147–162.

[26] E. P. Lafortune and Y. D. Willems, "Bi-directional path tracing," in *Compugraphics '93*, 1993, pp. 145–153.

[27] E. Veach and L. J. Guibas, "Metropolis light transport," in *SIGGRAPH '97*, 1997, pp. 65–76.

[28] H. S. Wilf, *Algorithms and Complexity*. AK Peters, 2003.



Anthony Pajot is a Ph.D. student in Computer Science at IRIT, under the supervision of Mathias Paulin and Loïc Barthe. He received his M.Sc. degree in Computer Science from Université Paul Sabatier, Toulouse, France, in 2008. His research interests include physically based rendering, numerical integration using Monte-Carlo methods, and GPU computing.

Loïc Barthe is assistant professor since 2003 in the VORTEX group of the IRIT institute in Toulouse. He received his Ph.D. in 2000 at the University of Toulouse and worked as an research associate at the University of Cambridge and at the RWTH of Aachen. He is interested in expression through graphics, and more specially in shape representation, geometry rendering, and intuitive modelling. He is Vice-President of the Eurographics France executive committee since 2008 and he has been member of 15 international conference program committees.

Mathias Paulin completed his Ph.D. in Computer Science at IRIT in 1995. His career at Université Paul Sabatier progressed from assistant to associate professor in 1996, and then to full professor in 2008. He is leading the Visualisation and Rendering research group at IRIT. His research interests include physically based rendering, high-quality real-time rendering, and advanced architectures for computer graphics.

Pierre Poulin is a full professor in the Computer Science and Operations Research department of the Université de Montréal. He holds a Ph.D. from the University of British Columbia and a M.Sc. from the University of Toronto, both in Computer Science. He has served on program committees of more than 40 international conferences. His research interests cover a wide range of topics, including image synthesis, image-based and procedural modeling, natural phenomena, scientific visualization, and animation.