GENERIC BRDF SAMPLING
A Sampling Method for Global Illumination

Rosana Montes, Carlos Ureña, Rubén García and Miguel Lastra
Dpt. Lenguajes y Sistemas Informáticos, E.T.S.I. Informática y de Telecomunicación, University of Granada, Spain
{*rosana,curena,ruben,mlastral*}*@ugr.es*

Keywords: BRDF, Importance Sampling, Monte Carlo Integration, Global Illumination, Rendering, Path Tracing.

Abstract: This paper introduces a new BRDF sampling method with reduced variance, which is based on a hierarchical adaptive parameterless PDF. This PDF is based also on rejection sampling with a bounded average number of trials, even in regions where the BRDF does exhibit high variations. Our algorithm works in an appropiate way with both physical and analytical reflectance models. Reflected directions are sampled by using importance sampling of the BRDF times the cosine term. This fact improves computation of reflected radiance when Monte-Carlo integration is used in Global Illumination. Test images have been obtained by using a Monte-Carlo rendering system, and they show reduced variance as compared with those obtained by other known techniques.

1 INTRODUCTION

In Global Illumination software the Bidirectional Reflectance Distribution Function (BRDF) is used to describe how light is scattered at surfaces, and it determines the appearance of objects. Many reflection models have been proposed which account for real visual effects produced by object-to-object reflections, self-shadowing, retro-reflection, etc. Monte Carlo (MC) algorithms, which rely on BRDF sampling, include distributed ray tracing (Cook et al., 1984), path tracing (Kajiya, 1986), bidirectional path tracing (Lafortune and Willems, 1993), density estimation (Shirley et al., 1995) and photon mapping (Jensen and Christensen, 1995).

Monte Carlo methods usually require a large number of rays to be traced through the scene. The direction of the rays follows a stochastic distribution which depends on light sources, BRDFs and visual importance. A mayor challenge in incorporating complex BRDFs into a Monte-Carlo-based global illumination system is efficiency in sampling, however, complex reflectance models have no corresponding sampling strategies to use with.

In (Lawrence et al., 2004) a Monte-Carlo importance sampling technique was presented for general analytic and measured BRDFs based on its factorization. We have used factorized approximations of those BRDFs in order to compare Lawrence approach with ours.

This document presents a method to improve Monte-Carlo random walks by applying importance sampling of BRDFs to reduce the variance of the estimator. Reflected directions are generated with a probability density function that is exactly proportional to the BRDF times the cosine term. For generality, we have sampled many parametric BRDFs that are well-known in computer graphics: for plastics the Phong model and its variants (Phong, 1975; Blinn, 1977; Lewis, 1993; Lafortune and Willems, 1994) and (Schlick, 1993), for metals the He model (He et al., 1991), Strauss (Strauss, 1990), Minnaert Lunar reflectance (Minnaert, 1941), for rough and polished surfaces based on Torrance's microfacet representation (Maxwell et al., 1973; Cook and Torrance, 1982; Poulin and Fournier, 1990) and (Oren and Nayar, 1994). Anisotropy models (Ward, 1992; Ashikhmin and Shirley, 2000a; Ashikhmin and Shirley, 2000b) are also considered. In fact our representation makes no assumptions on the BRDF model but the need for evaluating the function giving two directions.

The rest of this document is organized considering: Section 2 gives an overview of current techniques for sampling the BRDF and explains how importance sampling works when Monte Carlo integration is used. Section 3 provides details of our algorithm which adaptively samples the BRDF. Results and time-error analysis are given in Section 4. Some discussion and ideas for future work conclude the paper.

2 REFLECTANCE EQUATION AND MONTE-CARLO ESTIMATION

One of the main interests in Global Illumination relies on the evaluation of the reflected radiance, by using the *reflectance equation*:

$$L_r(\mathbf{w}_o) \stackrel{\text{def}}{=} \int_\Omega f_r(\mathbf{w}_o,\mathbf{w}_i)\, L_i(\mathbf{w}_i)\, (\mathbf{w}_i\cdot\mathbf{n})\, d\sigma(\mathbf{w}_i) \quad (1)$$

Here L_i stands for incoming radiance and L_r for reflected radiance. The above equation is usually solved in global illumination by using MC integration, because it is often impossible to obtain analytic expressions for L_r or L_i. Let $\mathbf{w}_o = (u_x, u_y, u_z)$ and $\mathbf{w}_i = (v_x, v_y, v_z)$ be two unit-vectors in Ω, the hemisphere of unit radius with $\mathbf{n} \stackrel{\text{def}}{=} (0,0,1)$.

2.1 MC Numerical Estimation of L_r

Integration over the hemisphere Ω can be done by using three related measures defined in that domain: (1) the solid angle measure (which we note as σ), (2) the projected solid angle measure (σ_p) and (3) an area measure A.

$$(\mathbf{w}\cdot\mathbf{n})\, d\sigma(\mathbf{w}) \;=\; d\sigma_p(\mathbf{w}) \;=\; dA(h(\mathbf{w})) \quad (2)$$

Let D denote the unit radius disc in $\mathbb{R}^2$. By using equation (2), the reflectance equation (1) can be alternatively expressed as:

$$L_r(\mathbf{w}_o) = \int_D f_r(\mathbf{w}_o,\mathbf{w}_{xy})\, L_i(\mathbf{w}_{xy})\, dA(x,y) \quad (3)$$

where $\mathbf{w}_{xy} \in D$ is the projection of $\mathbf{w}_i$ onto D.

When numerical integration of an arbitrary integrable (w.r.t. a measure μ) function $g \in S \to \mathbb{R}$ is done by using MC techniques, random samples in S must be generated from a random variable with probability measure P —which obeys $P(S) = 1$ and it is absolutely continuous w.r.t μ—. The function $p \stackrel{\text{def}}{=} dP/d\mu$ is frequently called the probability density function (PDF) of those samples. From n such random samples (namely $\{x_1,\ldots,x_n\}$) we can build a new random variable (r.v.) X_n whose mean value is the integral I we want to compute. This is done by generating samples sets whose PDF is p, and evaluating X_n on them. The variance of X_n is a value which determines the efficiency of the method.

Designing efficient MC sampling methods usually means designing *good* PDFs by using all available information about g. The closer p to g/I the less variance we obtain (ideally $p = g/I$). Consider now integrals like equation (1) and assume we have no knowledge about irradiance or other terms of the integrand, but with a known BRDF. In these circumstances, the best option is to use a PDF which is as proportional as possible to the BRDF times the cosine term.

To compute an estimator of $L_r(\mathbf{w}_o)$, as defined in equation (1), for a given $\mathbf{w}_o \in \Omega$, we must use a set of samples $(\mathbf{s}_1,\ldots,\mathbf{s}_n)$, which are n identically distributed random vectors defined in Ω, with probability measure $P_{\mathbf{w}_o}$ (the probability measure depends on $\mathbf{w}_o$). With this sample set, the estimator of the outgoing radiance can be obtained as:

$$L_r(\mathbf{w}_o) \;\approx\; \frac{1}{n}\sum_{k=1}^{n} \frac{f_r(\mathbf{w}_o,\mathbf{s}_k)\,(\mathbf{s}_k\cdot\mathbf{n})}{q_{\mathbf{w}_o}(\mathbf{s}_k)}\, L_i(\mathbf{s}_k) \quad (4)$$

where $q_{\mathbf{w}_o} = dP_{\mathbf{w}_o}/d\sigma$ is the PDF associated to $P_{\mathbf{w}_o}$.

An alternative expression can be given by using equation (3) instead of (1) and it's used in our algorithm. In this case, the set of samples $((x_1,y_1),\ldots,(x_n,y_n))$ contains random vectors in D instead of in Ω, and the estimator becomes:

$$L_r(\mathbf{w}_o) \;\approx\; \frac{1}{n}\sum_{k=1}^{n} \frac{f_r(\mathbf{w}_o,\mathbf{s}_k)}{p_{\mathbf{w}_o}(x_k,y_k)}\, L_i(\mathbf{s}_k) \quad (5)$$

where $\mathbf{s}_k$ is the projection of (x_k,y_k) onto Ω.

In this case, the PDF $p_{\mathbf{w}_o} = dP_{\mathbf{w}_o}/d\sigma_p = dP_{\mathbf{w}_o}/dA$ is defined w.r.t. area measure A, and its domain is D. Finally, from equations (4) and (5) we conclude that the PDF must be evaluated, and thus we should be capable to do this in a short time.

2.2 Sampling the BRDF

2.2.1 Lobe Distribution Sampling

A well known class of BRDF models are based on cosine-lobes, which have an associated algorithm for sampling. Within this category are Phong, Blinn and their respective normalized versions delivered by Lewis, Lafortune and Ward. The single-lobe BRDF is defined as:

$$f_r(\mathbf{w}_o,\mathbf{w}_i) \;=\; C(n)\,(\mathbf{w}_i\cdot\mathbf{w}_{or})^n$$

where $n \geq 0$ is a parameter, and $C(n)$ is a normalization factor which normally depends on n and ensures these BRDFs obey conservation of energy.

For this BRDF, a related and normalized PDF can be defined as:

$$p_{\mathbf{w}_o}(\mathbf{w}_i) \;=\; \frac{1}{N_1(\mathbf{w}_{or},n)}\,(\mathbf{w}_i\cdot\mathbf{w}_{or})^n$$

where N_1 ensures normalization and is defined as:

$$N_1(\mathbf{a},n) \stackrel{\text{def}}{=} \int_\Omega (\mathbf{w}_i \cdot \mathbf{a})^n \, d\sigma(\mathbf{w}_i)$$

N_1 is called a *single axis moment around axis* $\mathbf{a}$ and analytical expressions for it are known (Arvo, 1995).

In order to obtain samples distributed according to this PDF, we obtain a random vector $\mathbf{w}_i$ whose spherical coordinates are:

$$(\theta_{\mathbf{w}i}, \phi_{\mathbf{w}i}) = \left(arccos\left(\xi_1^{\frac{1}{n+1}}\right), 2\pi\xi_2 \right)$$

where ξ_1 and ξ_1 are two independent uniformly distributed random variables with values in $[0,1)$.

A variant of this PDF avoids evaluation of N_1 by using samples on the whole sphere $\mathcal{S}^2$, instead of only the hemisphere Ω. Taking into account the part of the lobe under the surface, it makes $N_1(\mathbf{w}_o,n)$ independent of $\mathbf{w}_o$ and equal to $N_1(\mathbf{n},n) = 2\pi/(n+1)$. This PDF is defined in the sphere, however, when a sample is produced under the surface, the contribution of that sample to the integral is taken as zero. The algorithm is faster and still unbiased, but it has higher variance when $\mathbf{w}_o$ approaches grazing angles.

Cosine-lobe sampling is the most efficient sampling for Phong BRDF and its variations but this scheme is not suitable for non-lobe-based BRDFs.

2.2.2 The Factorized BRDF Representation

Recent work about effective importance sampling strategies for arbitrary BRDFs is Lawrence's factorization of the BRDF (Lawrence et al., 2004). This function is decomposed as the product of two 1D functions, stored compactly in tabular form, and then it is used for sampling.

A first factorization, after a reparametrization based on the half angle, gives a decomposition into 2D factors of the initial data matrix Y containing $N_{\mathbf{w}} \times N_{\mathbf{w}o}$ values along the outgoing elevation angle and the outgoing azimuthal angle. After that, Y is approximated by the product of two matrices of lower dimension: G is $N_{\mathbf{w}} \times J$ and F is an $J \times N_{\mathbf{w}o}$ matrix. Both matrices are always positive by using the non-negative matrix factorization (NMF) method (Lee and Seung, 2000) [1].

A second factorization of the view independent G matrix leads to the product of two matrices of one dimension, very easy to sample by numerical inversion of the Cumulative Distribution Function after normalization.

[1]Code sample is given by J. Lawrence in <http://www.cs.virginia.edu/ jdl/nmf/>.

$$f_r(\mathbf{w}_o,\mathbf{w}_i)\cos(\mathbf{w}_i) \approx \sum_j^J F_j(\mathbf{w}_o) \sum_k^K u_{jk}(\theta_{\mathbf{w}}) v_{jk}(\phi_{\mathbf{w}}). \quad (6)$$

Each $L = J \times K$ factor is intuitively the approximation of a specific lobe of the original BRDF. When the factorization is used in generating random directions two steps are necessary. First sampling according to F selects one of the L lobes that contributes more energy for the current view. The CDF for this step is recomputed when the outgoing direction changes. Next the hemisphere is sampled according to selected lobe l by sequential generation of elevation and azimuthal angles using pre-computed CDF for factors u_l and v_l respectively.

3 OUR ALGORITHM

We consider the reflectance equation given in (3), and the estimator in (5). The proposed sampling scheme yields more samples in areas where the BRDF times the cosine term has higher values, thus achieving importance sampling. The usage of area measure A on D is better than σ on Ω because this makes it unnecessary to include the cosine term in the formulation or the computation, making the first simpler and the second faster and more reliable. Also, the algorithm is independent of the BRDF and avoids user guidance.

Our method is based on rejection sampling (Gentle, 2003). This is a very simple and well known technique that yields a PDF proportional to any function $g \in G \rightarrow \mathbb{R}$. It only requires that g can be evaluated, and its maximum value m in G to be known. However, it runs a loop which in fact can be executed a unbounded number of times, thus it potentially yields large computing times even in the cases when g can be quickly evaluated.

The probability for a sample to be accepted is e/m, where $e > 0$ is the average value of g in the domain G. The number of times the main loop is executed (until a valid sample is obtained) is a geometric distribution with success probability e/m, and thus the average number of trials is m/e, which can be quite large for $e \ll m$.

The core of our approach is an hierarchical quadtree structure which can be used to efficiently obtain samples with a PDF exactly proportional to the target function. The adaptive approach checks whether a region can be safely used for raw rejection sampling. This check consists on evaluating, for that region, the average number n_t of trials with rejection sampling in that region. This can be known provided we know both e and m for the region. If n_t is above a

threshold number n_{max}, then the region is subdivided in four, and the criterion is applied to these four subregions. Otherwise, the region is not subdivided. If we apply this recursive process starting from D (the unit radius disc centered at the origin), we obtain a quadtree which can be used to efficiently sample the BRDF. In the next section, further details are given about this process.

3.1 Building the Adaptive Structures

As the sampling process requires a PDF proportional to $f_r(\mathbf{w}_o,\cdot)$ for arbitrary values of $\mathbf{w}_o$ and for a finite collection of BRDFs in a scene, it is necessary to create a *quadtree* structure that subdivides the unit disc domain for each $(f_r,\mathbf{w}_o)$ pair. In the case of $\mathbf{w}_o$, a finite set of vectors $S=\{\mathbf{w}_1,\ldots,\mathbf{w}_n\}$ can be used. When an arbitrary $\mathbf{w}_o$ is given, it is necessary to select the nearest $\mathbf{w}_j$ to $\mathbf{w}_o$ and use the corresponding structure. The error induced by using $\mathbf{w}_j$ instead of $\mathbf{w}_o$ can be reduced by using a large n and uniformly distributing vectors $\mathbf{w}_j$. Note that, since we assume the BRDF to be isotropic, it is enough for S to include vectors in the plane XZ, thus a rotation must be applied to $\mathbf{w}_o$ before finding the nearest $\mathbf{w}_j$. The inverse rotation must be applied to resulting samples.

For a given quadtree in this structure, each node i has an associated region $R_i \subseteq D$, which it is a square area defined by:

$$R_i \;=\; [u_i,u_i+s_i)\times[v_i,v_i+s_i)$$

where (u_i,v_i) is the lower left vertex of the region boundary and s_i is the edge length. The region associated to the root node is the full domain $[0,1)^2$.

The algorithm creates the root node and it checks the criteria for subdivision. If the split is necessary, four new child nodes are created, each one with an associated region with a edge length size half of that of the parent. Then, process is recursively applied to these new four nodes. The recursive algorithm ends in case no split is necessary or a predefined maximal depth is reached.

In order to check the subdivision criteria for node i these values must be computed:

$$\begin{aligned} M_i &= \max\{\, f_r(\mathbf{w}_o,\mathbf{w}_{i_{xy}}) \mid (x,y)\in R_i \,\} \\ I_i &= \int_{R_i} f_r(\mathbf{w}_o,\mathbf{w}_{i_{xy}})\,dA(x,y) \\ V_i &= s_i^2 M_i \end{aligned}$$

M_i is the smaller upper bound for values of f_r in the i-th region, I_i is the integral of the BRDF in the region and V_i is the volume of the space where rejection sampling is done. Both M_i and I_i can be computed by evaluating f_r on a very dense grid of points in R_i creating the quadtree, or alternatively a bottom-up approach could be used which starts by obtaining these values at the maximum depth possible (with a high resolution grid) and then it stores them so the data can be used during tree construction. Therefore, the algorithm only requires to be able to evaluate the BRDF. In any case, it holds that the sum of the I_i values for the four children of a parent node must be equal to that value on the parent.

The subdivision criteria used must ensure that rejection sampling on leaf nodes can be done with an *a priori* bounded number of average trials n_{max}. This can be easily ensuring that:

$$n_{max}\,\frac{I_i}{V_i} \;\geq\; 1 \qquad (7)$$

where the probability for accepting a sample is I_i/V_i.

When this inequality does not holds, the node must be split. In our implementation, we have used $n_{max}=2$. The larger n_{max} the less memory that is needed (because the quadtree has smaller depth) and the less time is used for quadtree traversal, but more time is needed for rejection sampling on leaf nodes.

3.2 Obtaining Sample Directions

Generating a random direction involves selecting a leaf node and then doing rejection sampling on that node. If the i-th node is a leaf node, then the probability for selecting it must be proportional to I_i (more exactly it is I_i/I_0, if we assume the root node has index 0). A leaf node is selected following a path from the root to the leaf. On each step, starting from the root, the integrals I_i of the descendants nodes are used for randomly choosing one child to continue the path down.

To do this, we can store in each node i four values $F_{i0},\ldots,F_{i3}$, defined as:

$$F_{ik} \;=\; \frac{\sum_{j=0}^{k} I_{C_{ij}}}{\sum_{j=0}^{3} I_{C_{ij}}}$$

where C_{ij} is the index of i-th node j-th child node (note that $F_{i3}=1$). Leaf selection is then simply a loop:

Algorithm 1

LeafNodeSelection( )

```
i := 0 (index of root node)
while i-th node is not a leaf do begin
   r := uniform random value in [0,1)
   j := min. natural such that r < F_ij
   i := j
end
return i
```

Rejection sampling on the resulting i-th node is carried out. This consists in selecting a random vector $(x,y,z) \subseteq \mathbb{R}^3$ with uniform distribution in the prism $R_i \times [0, M_i]$. Value $z = \sqrt{x^2+y^2}$ is then obtained and the condition $f_r(\mathbf{w}_o, \mathbf{w}_{xy}) < z$ is checked. If it holds,$\mathbf{w}_{xy}$ is returned as the resulting sample, otherwise a new sample must be generated and checked. A sample is valid with probability I_i/V_i, which is necessarily greater than $1/n_{max}$, because of inequality (7).

With our method samples on the disc will follow a distribution where more samples are placed in parts of the domain where the function has higher values. In fact, it is exactly proportional to the BRDF.

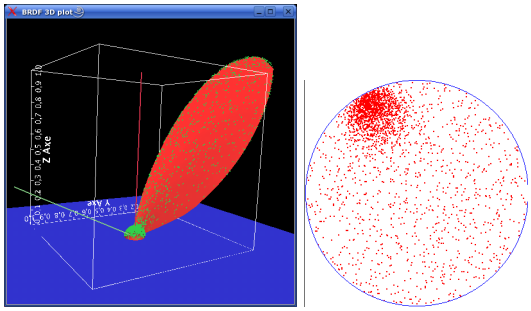

Figure 1: Both images show a distribution of 2500 samples obtained with our disc method. The left one shows how the samples match the BRDF function (in red). The image on the right is the projection on disc of those directions.

3.3 Quadtree Traversing for Optimal Sampling

Some considerations should be taken in order to increase the time performance. For example rather than asking for a single sample $\mathbf{s}_i$ we can implement a single recursive traversal algorithm which yields a set of n samples. Each node is visited once at most, instead of visiting it n times as it would be the case when using the basic approach we introduced.

First the algorithm starts by requesting n samples in the root node region and proceeding recursively. Whenever a node with index i is visited, the program must produce t random samples in R_i. If the i-th node is a leaf, those t samples are obtained by rejection sampling. When i-th node is an inner node, a partition of t is done, selecting four random integer values $m_{i,0}, \ldots, m_{i,3}$, which hold $m_{i,0} + m_{i,1} + m_{i,2} + m_{i,3} = t$ and in such a way that the average value of $m_{i,j}$ is $n I_{C(i,j)} / I_i$. Then the algorithm is recursively called for each j-th child $C(i,j)$ of i-th node (this is not done if $m_{i,j} = 0$), and as a result we obtain four sets with t samples in total. These four sets can be joined in one, which is the resulting set of t samples. Each leaf node j contains nI_i/I_0 samples on the average, as required by importance sampling.

4 RESULTS

In this section we provide results for our adaptive sampling method for various reflectance models, and we compare the computing time and average relative error we obtain for several images under different sampling strategies (PDFs).

We have analyzed different PDF functions (including the one we present) and we have measured their performance for various BRDFs models when high variation occurs, for instance, at a specular peak. Images have been obtained with a varying number of samples per pixel ranging from 1 sample, 5^2, 10^2, 15^2, 20^2, 30^2, 40^2, 50^2 and 1000^2 samples. The maximum quality (1000^2 samples) has been used to produce a reference image. We assign to each image a relative error value, computed with respect to this reference image. We average relative error for all pixels with non-null radiance in the reference image and report it as a percentage.

The full set of images takes each BRDF function from a list of the most common theoretical and empirical models (table 2) and samples them with the following five PDFs: (1) uniform sampling technique, (2) cosine lobe sampling on $\mathcal{S}^2$ and Ω, (3) Lawrence's factorization (Lawrence et al., 2004) and (4) the proposed adaptive method. All the images were rendered using a naive path tracing algorithm in a Linux machine with an AMD64 processor and 2GB of RAM.

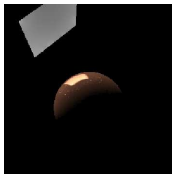

Figure 2: Rendering image example of the test scene.

4.1 Glossy Sphere

Considering a sphere object lit by a single area light, as shown in Figure 2. We focused our measures on the portion of the image containing the highlight on the sphere, because in that portion is where efficiency of different sampling approaches differs the most.

Each PDF model, with exception of uniform sampling and our method, is assigned a set of manually adjusted parameters in order to match the target

BRDF. For example, a cosine-lobe based PDF uses an exponent parameter n. This value could be taken from the corresponding exponent in the BRDF in use, however, there is no information to set the PDFs exponent if we sample a BRDF model which does not depend on that parameter, thus a constant must be used. To make comparisons fairer we have manually found the exponent that yields the best match between the lobe-based PDF and each BRDF function. Even for Phong-based BRDFs, the best n for the PDF can be different to the BRDFs exponent. This is because both the PDF and the BRDF include the term $(\mathbf{w}_o \cdot \mathbf{w}_i)^n$, however the BRDF also includes the cosine term $(\mathbf{w}_i \cdot \mathbf{n})$ whereas the PDF does not.

For the Factored PDF we have found the best factorization. It is necessary to find seven values for each BRDF. Parameters are: $N_{\theta_{\mathbf{w}o}} \times N_{\phi_{\mathbf{w}o}}$ and $N_{\theta_{\mathbf{p}}} \times N_{\phi_{\mathbf{p}}}$ for matrix size, $J \times K$ for the numbers of lobes that approximates the BRDF and whether or not to use the half-angle reparametrization. Best values are found comparing the average original matrix value with the average from the product of factors.

To compare the various PDF functions we plot the sampling time obtained vs. non-null pixel averaged relative error. By considering this, we can select the best method as the one that gives less error for a given time. The results are plotted in Figure 3. Numerical data is given in Table 1.

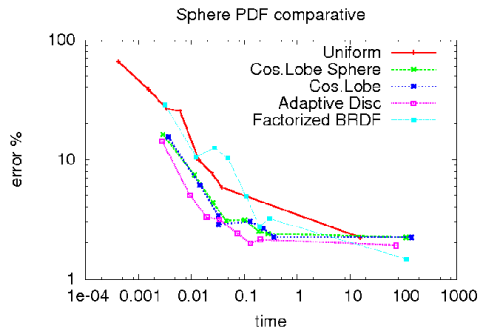

Figure 3: PDF comparative for Sphere scene. Manual selection of the cosine lobe exponent is needed, as well as the best factorization has to be found.

As the graph shows, the plot of our method is in most cases below the others. This means that, with same time our sampling performed best and also with same error our method needs less time. The adaptive method not only can be used with any isotropic BRDF but it also does not need manual selection of parameters, and it requires no knowledge of the BRDF. It just requires the ability to evaluate the BRDF.

Table 1: Relative error and average sampling time in seconds for each PDF and test scene when 50^2 samples are taken compared to the 1000^2 sample reference image.

	error	time
Uniform	5.82%	0.03792
C.Lobe $\mathcal{S}^2$	2.39%	0.28328
C.Lobe Ω	2.24%	0.35734
Adaptive	2.13%	0.20032
Factored	3.19%	0.2986

4.2 Sampling many BRDFs

In this point we treat on the Dragon model from the Stanford University [2]. The reflectance function used in this scene corresponds to Oren's (Oren and Nayar, 1994) with a rough value of 0.83, and a Strauss instance (Strauss, 1990) mostly smooth for floor and wall respectively. The dragon itself has a Lafortune BRDF (Lafortune and Willems, 1994) with exponent $n = 20$. With this mixture of BRDFs, visually we can compare our sampling method with uniform sampling, cosine lobe in Ω, and the Factored representation of Lawrence, with manually adjusted parametrization to fit the shape of each BRDF instance. With only 100 samples, our algorithm gives results with less noise than the others. You can see in Figure 4.

4.3 Quadtree Set Construction Requirements

It was mentioned previously that our algorithm involves some more computations in order to closely represent any BRDF function. Table 2 shows information related to the cost in seconds of the pre-computation for a given number of quadtree structures and varying incident angle directions. Once we have these structures on memory, they are used to estimate radiance. The values that are listed in the table correspond to the pre-computation of 90 quadtrees, which is high enough to ensure a structure is available very close to any incident direction. Average value is 20.71 seconds compared with 51.27, the cost of factorized computation and pre-computation of CDFs for sampling by using Lawrence's technique. Also you may notice the extreme difference in terms of time between experimental and physically based reflection models.

Another issue concerning the requirements of our method is memory consumption. Let us consider

[2]The Stanford 3D Scanning Repository at <http://graphics.stanford.edu/data/3Dscanrep/>

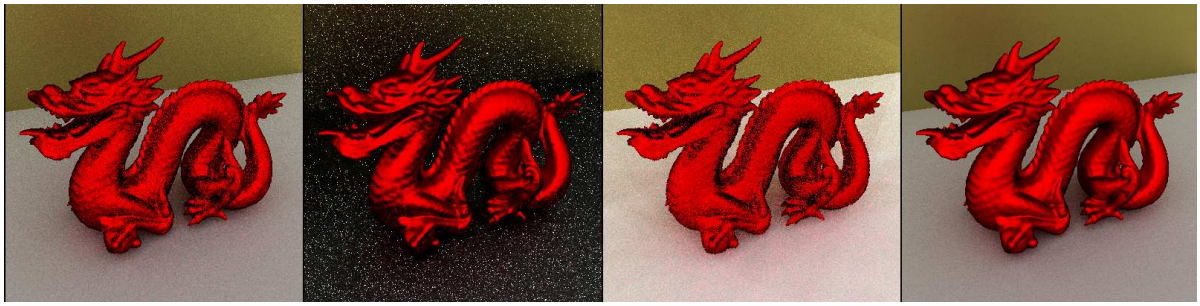

Figure 4: From left to right, images corresponding to uniform PDF, adjusted cosine-lobe strategy in Ω, the Factored representation of the BRDF and finally our algorithm sampling. Adaptive Disc shows less noise using the same number of samples than the others. The resolution is 400 x 400 pixels. Following the same order, sampling time is: 9.232, 114.735, 90.028 and 133.172 seconds respectively.

Table 2: Quadtree creation times for each BRDF model for Adaptive method compare with factorization and precomputation times of Factored PDF. Memory requirements for both methods are also given. Data is relative to the glossy scene.

BRDF	Adaptive		Factorized	
	(sec)	(KB)	(sec)	(KB)
Ashikhmin	51.4	6.25	82.9	1031
BeardMax.	15.5	1713.25	17.3	6454
Blinn	8.7	582.25	83.7	6481
Coupled	22.6	6.25	22.2	1033
He	102.7	2407.25	75.1	1034
Lafortune	6.6	1275.25	53.2	6445
Lewis	6.9	1279.25	119.3	6445
Minnaert	7.3	1461.25	4.9	1031
Oren	10.5	6.25	8.5	1033
Phong	6.9	1279.25	9.2	6445
Poulin	35.5	297.25	5.4	1038
SchlickD	19.1	342.25	77.9	1033
SchlickS	13.2	780.25	26.5	1043
Strauss	10.9	727.25	59.3	1052
Torrance	8.3	631.25	122.0	1029
Ward	20.7	483.25	51.3	1038

firstly our basic algorithm with no optimizations. A single quadtree represents the unit disc domain as node regions given an incident direction. Its depth, and so its memory, depends on the nmax parameter (see equation 7). Table 3 shows the cost of a single quadtree when the nmax param changes. Memory increments when many quadtrees are calculated and stored. Table 2 shows the cost in KB for 90 quadtrees and for each BRDF taking nmax as 2. Avarage value of our method is 0.81 MB compared with 2.67 MB of the factorized BRDF.

Table 3: Memory in KBytes for Adaptive Disc using $\theta_{\mathbf{w}o} = 74^\circ$.

Disc	nmax	1.3	1.6	2	2.3	2.6
	KB	2056.5	543	12.51	4.6	3.33

5 CONCLUSIONS

We have presented a new sampling method based on an adaptive and parameterless algorithm which implements a PDF exactly proportional to generic BRDF. Reflected directions were sampled using importance sampling of the BRDF times the cosine term which is preferable to only sampling the BRDF. The method can be used for numerical Monte-Carlo based integration in global illumination or in other contexts. Its efficiency is similar or even better than standard sampling methods with manually selected optimal parameter values.

We also tested our adaptive sampling method with tabulated BRDF representations (Matusik et al., 2003) since they can be evaluated. We further plan to develop, as future work, a method to acquire BRDF data from an inexpensive 3D scanner, as a way to deal with real world materials. We also plan to further reduce both sampling time and quadtree construction time by using more optimized sequential programs, SIMD instructions sets or parallel graphics hardware.

ACKNOWLEDGEMENTS

Special thanks to Francisco Javier Melero for his contribution on this work. This work has been supported by a grant coded as TIN2004-07672-C03-02 of the Spanish Ministry of Education and Science.

REFERENCES

Arvo, J. (1995). Applications of irradiance tensors to the simulation of non-lambertian phenomena. In *SIGGRAPH 1995 Proceedings*, pages 335–342. ACM Press.

Ashikhmin, M. and Shirley, P. (2000a). An anisotropic phong brdf model. *J. Graph. Tools*, 5(2):25–32.

Ashikhmin, M. and Shirley, P. (2000b). A microfacet-based brdf generator. In *SIGGRAPH 2000 Proceedings*.

Blinn, J. F. (1977). Models of light reflection for computer synthesized pictures. In *SIGGRAPH 1977 Proceedings*, pages 192–198. ACM Press.

Cook, R. and Torrance, K. (1982). A reflectance model for computer graphics. In *SIGGRAPH'81 Proceedings*, volume 1, pages 7–24. ACM Press.

Cook, R. L., Porter, T., and Carpenter, L. (1984). Distributed ray tracing. In *SIGGRAPH'84 Proceedings*, pages 137–145, New York, NY, USA. ACM Press.

Gentle, J. E. (2003). *Random number generation and Monte Carlo methods. (2nd ed.)*. Springer.

He, X., Torrance, K., Sillion, F., and Greenberg, D. (1991). A comprehensive physical model for light reflection. In *SIGGRAPH 1991 Proceedings*, pages 175–186. ACM Press.

Jensen, H. W. and Christensen, N. (1995). Photon maps in bidirectional monte carlo ray tracing for complex objects. In *Computer & Graphics*, number 2, pages 215–224.

Kajiya, J. T. (1986). The rendering equation. In *SIGGRAPH '86 Proceedings*, pages 143–150. ACM Press.

Lafortune, E. P. and Willems, Y. D. (1993). Bi-directional path tracing. In *Proceedings of CompuGraphics, Alvor, Portugal*, pages 145–153.

Lafortune, E. P. and Willems, Y. D. (1994). Using the modified phong reflectance model for physically based rendering. Technical Report Report CW197, Department of Computer Science, K.U.Leuven.

Lawrence, J., Rusinkiewicz, S., and Ramamoorthi, R. (2004). Efficient brdf important sampling using a factored representation. In *ACM Transaction of Graphics. Siggraph 2004*, number 3, pages 496–505.

Lee, D. and Seung, H. (2000). Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562.

Lewis, R. R. (1993). Making shaders more physically plausible. In *Eurographics Workshop on Rendering*, pages 47–62, Vancouver, BC, Canada.

Matusik, W., Pfister, H., Brand, M., and McMillan, L. (2003). A data-driven reflectance model. *ACM Trans. Graph.*, 22(3):759–769.

Maxwell, J. R., Beard, J., Weiner, S., and Ladd, D. (1973). Bidirectional reflectance model validation and utilization. Technical report, AFAL–TR–73–303. ERIM.

Minnaert, M. (1941). The reciprocity principle in lunar photometry. *Astrophysical Journal*, pages 403–410.

Oren, M. and Nayar, S. (1994). Generalization of lambert's reflectance model. In *SIGGRAPH 1994 Proceedings*, pages 239–246, New York, NY, USA. ACM Press.

Phong, B.-T. (1975). Illumination for computer generated pictures. In *ACM Siggraph'75 Conference Proceedings*, number 6, pages 311–317, New York, NY, USA. ACM Press.

Poulin, P. and Fournier, A. (1990). A model for anisotropic reflection. In *SIGGRAPH 1990 Proceedings*, number 4, pages 273–282, New York, NY, USA. ACM Press.

Schlick, C. (1993). A customizable reflectance model for everyday rendering. In *Eurographics Workshop on Rendering*, pages 73–84.

Shirley, P., Bretton, W., and Greenberg, D. (1995). Global illumination via density-estimation radiosity. In *Eurographics Workshop on Rendering*.

Strauss, P. S. (1990). A realistic lighting model for computer animators. *IEEE Comput. Graph. Appl.*, 10(6):56–64.

Ward, G. J. (1992). Measuring and modelling anisotropic reflection. In *ACM Siggraph'92 Conference Proceedings*, number 4, pages 265–272.