

# BRDF Measurement Modelling using Wavelets for Efficient Path Tracing

L. Claustres<sup>1</sup>, M. Paulin<sup>1</sup> and Y. Boucher<sup>2</sup>

<sup>1</sup>IRIT, University Paul Sabatier, Toulouse, France

<sup>2</sup>ONERA, Toulouse, France

---

## Abstract

Physically based rendering needs numerical models from real measurements, or analytical models from material definitions, of the Bidirectional Reflectance Distribution Function (BRDF). However, measured BRDF data sets are too large and provide no functionalities to be practically used in Monte Carlo path tracing algorithms. In this paper, we present a wavelet-based generic BRDF model suitable for both physical analysis and path tracing. The model is based on the separation of spectral and geometrical aspect of the BRDF and allows a compact and efficient representation of isotropic, anisotropic and/or spectral BRDFs. After a brief survey of BRDF and wavelet theory, we present our software architecture for generic wavelet transform and how to use it to model BRDFs. Then, modelling results are presented on real and virtual BRDF measurements. Finally, we show how to exploit the multiresolution property of the wavelet encoding to reduce the variance by importance sampling in a path tracing algorithm.

ACM CSS: I.3.7 Computer Graphics—Three-Dimensional Graphics and Realism

---

## 1. Introduction

As global illumination algorithms become more and more efficient and physically realistic, the need for physical data grows. There are lots of different techniques to provide an accurate solution of the general radiative transfer equation [1] used in physics, or the rendering equation [2], used in computer graphics. However, the quality of *global* solutions is highly driven by the quality of *local* solutions. The bidirectional reflectance distribution function (BRDF), defining *local illumination* of the surfaces, has an influence on *global illumination* because of the inter-reflections between the surfaces of the scene. The problem comes from the complexity of the BRDF. The BRDF is theoretically a function of many variables: time, surface position, incoming (or lighting) direction, outgoing (or observer) direction, polarization, incoming and outgoing wavelength. As a consequence, global illumination uses a variety of approximations. First, materials are uniform (suppress dependence on position). Secondly, polarization is ignored (incoherent lighting). Thirdly, time-dependent behaviour is avoided (light has an infinite

speed leading to instantaneous equilibrium). Finally, the energy associated with a given wavelength is independent of the energy associated with a different wavelength (no frequency shift). This does not allow modelling of phosphorescence effects but reduces the wavelength dependence to a single term. There are still five dimensions left : incoming and outgoing directions, and wavelength. Therefore an accurate BRDF measurement generates a huge number of samples. We need a compression scheme which can handle this large amount of data and which is both efficient and convenient.

Fitting an analytical model to the measurement data set is a common way to proceed. This approach requires advanced numerical optimization algorithms. The best parameters of the model are retrieved by minimizing the error between modelled and measured data. Analytical models are created according to a theoretical or empirical approach, but both lack generality. Physical models are very complex and also very specific. Empirical models are only suitable for a particular class of surfaces. Moreover, to the best of our

knowledge, a truly anisotropic and spectral model does not exist. Anisotropic BRDF models, like the Lafortune *et al.* [3] model, can be used with a different set of parameters for each wavelength. But there is no *explicit* modelling of the BRDF wavelength dependence. Finally, in Monte Carlo path tracing simulations we need to generate samples according to the BRDF magnitude (*importance sampling*). This process can dramatically reduce the variance of the algorithm. Unfortunately, the complexity of reflection models does not allow a mathematical inversion. These reasons press us to choose a numerical representation based on wavelets. We see multiple advantages:

- *Generality*: can represent any kind of BRDF;
- *Compression*: well-suited to large data sets;
- *Speed*: reconstruction is done in a logarithmic time according to the number of samples;
- *Multiresolution*: reconstruction at different levels of accuracy is possible;
- *Denoising*: induced by compression.

Wavelets are also well-suited to low-frequency signals with localized high frequencies. They are naturally adapted to glossy surfaces where light reflection is concentrated in a small solid angle around the mirror direction (if the discretization grid is small enough).

The paper is organized as follows: Section 2 introduces the background of BRDF and wavelets. Then, analysis of previous work is done in Section 3, and shows how we were induced to build a new approach. Section 4 explains our choices in detail, while results achieved on real and virtual measurements are presented in Section 5. Section 6 demonstrates that our representation can be used to reduce the variance of a Monte Carlo path tracing application. We conclude and discuss future research in Section 7.

## 2. Definitions

### 2.1. BRDF

The BRDF was first introduced and defined by Nicodemus [4] as a radiometric term describing how a surface reflects light. For a given *illumination* or *incoming* direction  $\omega_i$ , and an *observer* or *outgoing* direction  $\omega_r$ , the BRDF is defined as the ratio of outgoing radiance to incoming irradiance. The BRDF is usually expressed with spherical coordinates in the local coordinate system of the surface point  $x$  (Figure 1).

$$f_r(x, \theta_i, \phi_i, \theta_r, \phi_r, \lambda) = \frac{dL_r(x, \omega_r, \lambda)}{dE_i(x, \omega_i, \lambda)} = \frac{dL_r(\theta_r, \phi_r, \lambda)}{L_i(\theta_i, \phi_i, \lambda) \cos \theta_i d\omega_i} \quad (1)$$

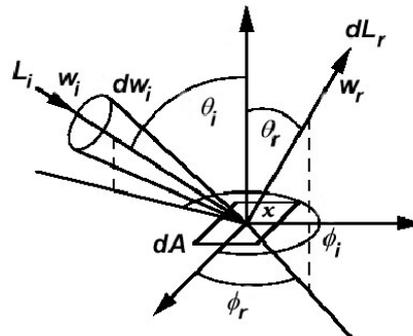


Figure 1: Local surface geometry.

The BRDF has interesting properties:

- (1) *Non-negativity*: the BRDF is a positive definite function

$$f_r(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) > 0 \quad (2)$$

- (2) *Helmholtz reciprocity*: the behaviour of the surface is independent of the luminous flux direction [5], i.e.

$$f_r(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) = f_r(\theta_r, \phi_r, \theta_i, \phi_i, \lambda) \quad (3)$$

- (3) *Energy conservation*: energy reflected by the surface is not greater than incoming energy on the surface, i.e. for each incoming direction  $\omega_i$  (assuming integration over wavelength is implied)

$$\int_{\Omega_r} f_r(\theta_i, \phi_i, \theta_r, \phi_r) L_i(\theta_i, \phi_i) \cos \theta_r d\omega_r < 1 \quad (4)$$

- (4) *Isotropy*: for many surfaces the reflectance does not change with respect to surface orientation (*isotropic BRDF*), i.e.

$$f_r(\theta_i, \phi_i, \theta_r, \phi_r, \lambda) = f_r(\theta_i, \theta_r, \phi = |\phi_r - \phi_i|) \quad (5)$$

otherwise the BRDF is said to be *anisotropic*.

### 2.2. Wavelets

In the last ten years wavelets have become a powerful tool in various numerical applications. They allow compression, fast reconstruction algorithms, and multiresolution. Therefore, it is not surprising to meet them whenever the size of the data is critical: remote sensing or topography [6], surface modelling [7], global illumination algorithms [8,9], and multimedia compression [10,11]. The reader is referred to Mallat [12] and Daubechies [13] work for a detailed and technical description of wavelet theory. Specific applications of wavelets to Computer Graphics are described by Stollnitz [14].



Figure 2: Tetrahedral subdivision at levels 0, 1, 2.

### 3. Previous Work

Representations based on projective methods using bases other than wavelets might be used for BRDF modelling. Cabral [15] gives a solution using *spherical harmonics*. The problem comes from the cost of evaluation, because spherical harmonics have *global support* on the sphere. Moreover, compression is less impressive and more difficult to adjust with the accuracy needed.

#### 3.1. Spherical Wavelets

The first attempt to model BRDF using wavelets was mentioned by Schröder and Sweldens [6]. Schröder extends one-dimensional wavelets to spherical wavelets in order to efficiently represent functions on the sphere  $\mathbf{S}^2$  or the hemisphere  $\mathbf{H}^2$ . Spherical wavelets are defined in  $L_2 = L_2(\mathbf{S}^2, d\omega)$ , the space of finite energy functions defined over  $\mathbf{S}^2$ . This is the case of the BRDF owing to energy conservation.  $\mathbf{S}^2$  is discretized using a geodesic construction. Starting from an octahedron (or a tetrahedron for  $\mathbf{H}^2$ ), a recursive process subdivides each triangle into four children until a given *accuracy level* (Figure 2). At each step, triangles are projected onto the sphere in order to reduce the deformations induced by the curvature. This construction is *parameterization independent* and almost regular in terms of solid angle. But generating an equitable distribution of points on the sphere is still an open mathematical problem. Nevertheless, no direction is privileged in comparison with a standard discretization of spherical coordinates  $(\theta, \phi)$ , which would result in giving more importance to the poles. A function defined over  $\mathbf{S}^2$  is approximated by a piecewise constant function defined over the triangles at the finest level. The value for a triangle is equal to the one for the center of the triangle. Schröder defines a multiresolution analysis in the function space  $L_2(\mathbf{S}^2, d\omega)$  and constructs the *Bio-Haar* wavelet basis, which is an extension of the well-known Haar basis [16,17], thanks to the *Lifting Scheme* [18]. Scaling and wavelet functions at level  $j + 1$  are locally defined over the space of the father triangle (level  $j$ ). They make a biorthogonal basis as they do in the one-dimensional case.

The major drawback of this work is that it can only be used for a fixed wavelength and fixed incoming direction. Indeed it can only handle functions defined over  $\mathbf{S}^2$ . But compressing only BRDF values for fixed incoming

directions is a poor idea as it fails to use the incoming direction coherence.

#### 3.2. Multidimensional Wavelets

The representation presented by Lalonde and Fournier [19] is based on projecting the BRDF onto four-dimensional wavelet bases. Multidimensional wavelet bases are constructed from one-dimensional wavelet bases using a product of basis functions (*non-standard* approach [13]). The BRDF, which is naturally defined over  $\mathbf{H}^2 \times \mathbf{H}^2$ , needs a new parameterization to map the 4D space. Lalonde recommends the use of *Nusselt* embedding of  $\mathbf{S}^2 \times \mathbf{S}^2$  into  $\mathbf{R}^4$  to compromise on the matter of redundancy at the poles [20]. This transformation maps a couple  $(\theta, \phi) \in \mathbf{S}^2$  to a couple  $(\kappa, \lambda) \in \mathbf{R}^2$ . The projection of the BRDF onto the 4D space is written as follows:

$$f_r(\kappa_i, \lambda_i, \kappa_r, \lambda_r) = \sum_j f_j F_j(\kappa_i, \lambda_i, \kappa_r, \lambda_r) \quad (6)$$

$$\text{with } f_j = \langle f_r(\kappa_i, \lambda_i, \kappa_r, \lambda_r) | \tilde{F}_j(\kappa_i, \lambda_i, \kappa_r, \lambda_r) \rangle$$

This work should be easily extended to handle spectral BRDF by adding one dimension. More, the model is not restricted to one particular wavelet basis, in opposition to the previous one that uses a Haar-like basis. Lalonde used particularly Daubechies and spline wavelets. But multidimensional wavelets defined over  $\mathbf{R}^2$  are not well-suited to represent spherical data sets because of the topological distortion of  $\mathbf{S}^2$ . BRDF space cannot be directly mapped to representation space and modelling is *parameterization dependent*.

#### 4. Generic Wavelet Transform

Each of the modelling methods described above has both advantages and drawbacks. Thinking in term of *generic wavelet transform* (wavelet transform of any kind of objects), allows us to keep the advantages while removing most of the drawbacks. The central idea is based on the change usual in functional analysis: every  $f : \mathbf{A} \times \mathbf{B} \mapsto \mathbf{C}$  is equivalent to an  $\tilde{f} : \mathbf{A} \mapsto (\tilde{g} : \mathbf{B} \mapsto \mathbf{C})$ , based on the correspondence between  $((a, b), c)$  and  $(a, (b, c))$ .

##### 4.1. BRDF Representation

The BRDF (and also any radiometric term) is a combination of a *geometrical component* (directional aspect defined over  $\mathbf{S}^2$ ) and a *spectral component* (wavelength dependency defined over  $\mathbf{R}$ ). By defining a generic spherical and one-dimensional wavelet transform, we show that we are able to represent any type of BRDF (isotropic, monochromatic, anisotropic, spectral), with a combination of one or more transforms.

The discrete wavelet transform is basically a filtering operation, so only basic algebraic operators need to be defined (scalar multiplication/division and addition/subtraction). An inner product is also necessary in order to evaluate coefficient magnitude. Consider a function  $f : \mathbf{S}^2 \times \mathbf{R} \mapsto \mathbf{R}$  (for instance a spectral emittance distribution function). It can also be viewed as a function  $\tilde{f} : \mathbf{S}^2 \mapsto (\tilde{g} : \mathbf{R} \mapsto \mathbf{R})$ .  $\tilde{f}$  is a spherical function, whose result is a one-dimensional real function  $\tilde{g}$ . A discretized version will result in a spherical signal, whose elements are one-dimensional vectors. Each geodesic triangle on the sphere stores a vector. Because we can define algebraic operators, and also an inner product, vectors can be used as basic elements of a discrete wavelet transform. Instead of using a multidimensional transform (with multidimensional wavelet and scaling functions), we can apply *multiple atomic transforms* (one per space). The wavelet transform of the spherical signal is computed and results in wavelet and scaling coefficients that are vectors. Then, a standard one-dimensional wavelet transform is applied on these resulting coefficients. To perform signal synthesis, inverse transforms are applied in the reverse order.

In fact, this method is the *standard* approach [13] that uses the product of decompositions to transform multidimensional signals in contrast with the non-standard approach that uses the product of basis functions. The major difference is that we compress in a generic way too. Instead of removing all final real coefficients under a given threshold (which is usually done), we have different levels of compression. First a high level compression, that handles coherence in the function domain of  $\tilde{f}$ , which is  $\mathbf{S}^2$ . Then a basic compression level, that handles coherence in the hidden dimension-connections of  $\mathbf{S}^2$  and  $\mathbf{R}^2$ . Indeed, weak spherical wavelet coefficients (vectors) are removed according to their importance. We exploit coherence in the spherical domain. Then, weak elements of the remaining spherical coefficients are removed. Here we use coherence of the dimension-connections because the real coefficients inside each vector have undergone both transforms (spherical and one-dimensional transform). The compression is not globally controlled, but for each space. It is possible to compress the function more in the one-dimensional domain than in the spherical domain.

With generic transform there is unicity of source code because transforms are written identically on real values or other kind of objects in a programming language that handles genericity and algebraic operator definition (like C++). Once the generic wavelet transform is written, only algebraic operators and inner product have to be defined for objects we want to transform. Generic transforms keep the advantages of spherical and one-dimensional transforms. There is no need for space mapping. And even if we are restricted to the lifted Bio-Haar basis on the sphere, the wide number of one-dimensional wavelet bases is usable for the spectral component. Moreover the Bio-Haar basis

is well-suited to signals with discontinuities as shown by Schröder and Sweldens [6] (this is the case of shiny surface measurement data sets because of narrow peaks of the BRDF); and allows fast numerical inversion of the BRDF (this will be shown in Section 6).

## 4.2. Software Architecture

The root object of our architecture is the *Signal*. A Signal is simply a discretized analytical function  $f : \mathcal{A} \mapsto \mathcal{B}$ . Every measurement data set is a Signal (a sampled function). We represent a Signal as a generic object with two generic parameters: the type of the objects in domains  $\mathcal{A}$  and  $\mathcal{B}$ . Then we define a *Wavelet Transform* to be a signal that can be transformed (analysis), compressed, and inverted (synthesis). This conception introduces no restriction on object types and allows the creation of multiple wavelet transforms.

We define a *Reflectance* as a real spherical wavelet transform by letting  $\mathcal{A}$  be  $\mathbf{T}$  (the set of geodesic triangles) and  $\mathcal{B}$  be  $\mathbf{R}$ . This object can be used to represent either a monochromatic directional hemispherical reflectance, a directional emissivity, or any function  $f : \mathbf{S}^2 \mapsto \mathbf{R}$ . Now we choose to define a monochromatic BRDF as a spherical wavelet transform of Reflectance objects. It can be used to represent any function  $f : \mathbf{S}^2 \times \mathbf{S}^2 \mapsto \mathbf{R}$ , and therefore monochromatic BRDFs. Each incoming triangle stores a Reflectance object encoding BRDF values on the outgoing hemisphere. The BRDF wavelet transform is performed in two steps: first a spherical wavelet transform is done on incoming directions or Reflectance objects, then another transform is computed on outgoing directions or resulting coefficients (which are Reflectance objects themselves). Following the same idea, compression is done, first on incoming directions and then on outgoing directions. This process must be inverted for synthesis. The inverse transform is first performed on outgoing directions and then on incoming directions.

We also define a *Spectrum* as a one-dimensional real wavelet transform by letting  $\mathcal{A}$  and  $\mathcal{B}$  be  $\mathbf{R}$ . We can use it to represent any function  $f : \mathbf{R} \mapsto \mathbf{R}$ . In a similar way we define a *Spectral Reflectance* to be a spherical wavelet transform of Spectrum objects and a *Spectral BRDF* to be a spherical wavelet transform of Spectral Reflectance objects. With these two kinds of objects we can handle any function  $f : \mathbf{S}^2 \times \mathbf{R} \mapsto \mathbf{R}$  or  $f : \mathbf{S}^2 \times \mathbf{S}^2 \times \mathbf{R} \mapsto \mathbf{R}$ . The Spectral BRDF wavelet transform is equivalent to three atomic transforms: incoming and outgoing spherical transforms (geometrical components), plus a one-dimensional transform (spectral component).

## 4.3. Compression Structure

Wavelet compression removes wavelet coefficients lower than a given threshold  $\epsilon$ . But if wavelet coefficients are

only set to zero, there is no compression at all because the memory size of a coefficient does not depend on its value. It is necessary to use a *sparse* structure that can really erase zero coefficients from memory and maps the sparse wavelet representation. We choose to create a generic *Sparse Array* object inspired by strip partitioning of sparse trees. Indeed standard wavelet trees [21] require more memory than a flat structure due to the pointer overhead. Moreover a whole subtree or branch can be preserved if there is just one valid coefficient stored at a leaf. We will see it is not the case with our structure. Because our goal is to reduce the memory required, elegant and general structures such as hash-coding structures are not relevant. Indeed, the individual cost for each element is high (requiring an additional key).

Sparse array objects are used as standard one-dimensional arrays with, in addition, the ability of compression. An element of the array can be *invalidated*, i.e. considered as a zero value and erased from memory. A separation must be made between *localization* and *values* (or data). A *strip* consists of:

- a bit index : a bit array that indicates which element is valid or invalid (localization);
- a data array : an array that stores the valid elements (values).

The Sparse Array internal structure is a double-linked list of strips (Figure 3). The strip size  $\mathcal{B}_s$  (number of data elements per strip) is a construction parameter. From a standard array index  $i$  the strip number and the strip index are computed respectively as  $i/\mathcal{B}_s$  and  $i \bmod \mathcal{B}_s$ . When compression occurs, all elements of a strip can be invalidated, and the strip is entirely eliminated from the structure. So two adjacent strips can have non-consecutive numbers.  $\mathcal{B}_s$  is an important parameter because it regulates the global memory cost of the Sparse Array structure. If  $\mathcal{B}_s$  is very large, only a few strips will be used to represent the signal. So, we reduce the extra-memory cost of the list structure (pointers to the previous and next strip). But at the same time, we increase the memory cost of the bit index, and the bit index is not compressed while the data array is. This can result in a large bit index with only a few bits set and a lot of wasted unset bits. A Sparse Array composed of lots of little strips will be more flexible. In our experiment we found that  $\mathcal{B}_s = 32$  offers a good compromise.

The Sparse Array structure is only adapted to one-dimensional signals. But we can create a bijection between the set of geodesic triangles  $\mathbf{T}$  and array indices. The idea is to give a unique number to each spherical triangle. This number is used as an index in the sparse array structure; representing what we call a *Sparse Sphere*. A 2D topology can certainly not be mapped to a 1D topology without a loss of neighbourhood information. However, generating triangle numbers according to neighbourhood relationships on the

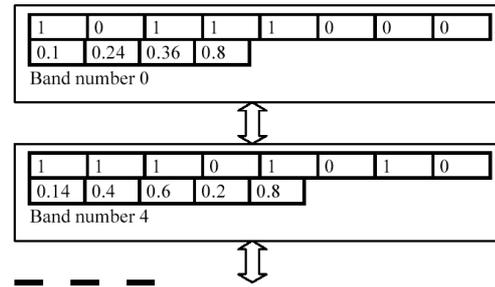


Figure 3: The Sparse Array structure.

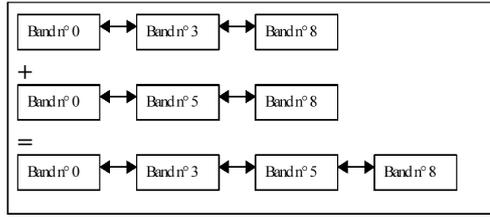
sphere and depth level, asserts a maximal data coherence. This trick allows spherical signals to use the Sparse Array structure. Moreover this method presents a major advantage: only one recursive triangle structure is necessary. For a discretized spherical function, the value of each triangle is stored in a Sparse Array according to its number. This flat structure is less memory intensive than a recursive data structure (due to the pointer overhead) such as the one used by Schröder. Nevertheless it can also be manipulated using simple recursive procedures *driven* by the single spherical topology structure.

#### 4.4. Algebraic Operators Coherence

We must define algebraic operators for our Signal objects, because they are elements of a generic wavelet transform (like the Reflectance objects for a monochromatic BRDF). Therefore we must define algebraic operators for our Sparse Array objects, because every Signal uses a Sparse Array to store data elements.

As first glance, it is very simple: apply each operator between elements at the same index in the two arrays. This works naturally for one-dimensional vectors (Spectrum objects) but also for spherical signals (Reflectance objects), because the same index corresponds to the same geodesic triangle. But an additional dimension must be taken into account: *data coherence*. Indeed, when the signals are compressed, the reconstruction step performs algebraic operations between compressed signals. A direct solution is to uncompress the arrays (all invalidated values are set to zero) before applying the operator to avoid any problem. In fact, operators can be directly defined on compressed arrays, which is more efficient (in terms of both memory and speed).

Multiplication, division by a scalar, or inner product are not problematic, even if the array is compressed. They are just scaling operations, so a zero value (or compressed value) will not affect the result of the operation. Addition (or subtraction) between two Sparse Array objects is more difficult because they usually have different compressed structures. The solution is given by a simple algorithm. If



**Figure 4:** Algebraic operation between two Sparse Array.

a strip is not present in both arrays, it will not be present in the resulting array. If a strip is only present in one array, the strip is also present in the resulting array. If a strip is present in both arrays, addition (or subtraction) is done between the two strips. The same algorithm is applied for the elements of the strips, according to their bit indices. Example of Sparse Array addition operator is presented in Figure 4.

#### 4.5. Interpolation

One drawback of our approach, which uses measurement data sets, is that we have a discrete reconstruction. Interpolation is done naturally during rendering, with the path tracing algorithm, by Monte Carlo integration. But a local continuous reconstruction is necessary for other applications (ray tracing for instance). We have chosen to separate the wavelet representation, which reconstructs the discrete data set, and the interpolation scheme, which reconstructs the continuous function. There are two reasons for this choice. First, because we want to use measurement data sets directly, without the need of any transformation. Secondly, because we use a Haar-derived wavelet on the sphere, which is piecewise constant.

We use a cubic B-Spline interpolation scheme for the spectral component [22], which is a powerful one-dimensional method. Spherical interpolation is more problematic because of the topology of  $S^2$ . We have implemented different techniques to reconstruct a continuous signal on a specific triangle. Some are based on topology, using neighbour triangles or shared vertices (for instance a barycentric interpolation), and some are topology independent (for instance a Monte Carlo integration). A cubic Clough–Tocher interpolant [23] gives good results. However, a detailed discussion of the interpolation problem is beyond the scope of this paper and will not be addressed here. The principal conclusion is that a good interpolation scheme allows a good compression ratio with regard to a given error.

## 5. Results

### 5.1. Measurements

In this section, we present results achieved on BRDFs measured with the *goniometer* developed and set up by the

ONERA [24]. We have selected a diverse range of materials. Artificial targets: spectralon (quasi-perfect diffuse surface used for calibration panels), plastic (ivory PVC), mélaminé (wood covered by a glossy white surface used in desk construction) and red cloth. But also natural targets: grass (green and dry) and sand. The total number of samples for a measurement is 485376 (474 directions and 1024 wavelength in the range 420–950 nm). We are restricted to the isotropic case for real data, due to our goniometer limitations. But anisotropic measurement data sets can be simulated using a virtual goniometer [25,26] or an analytical model.

The modelling error presented in the next three sections is given in terms of relative (normalized)  $L_1$  and  $L_2$  errors. Let  $f_i$  be the values of the original function with  $n$  samples, and  $f'_i$  the reconstructed values from the compressed one, then:

$$\mathbf{E}_{L_1} = \frac{1}{n} \sum_i \frac{|f'_i - f_i|}{f_i} \quad (7)$$

$$\mathbf{E}_{L_2} = \sqrt{\frac{1}{n} \sum_i \frac{(f'_i - f_i)^2}{f_i^2}} \quad (8)$$

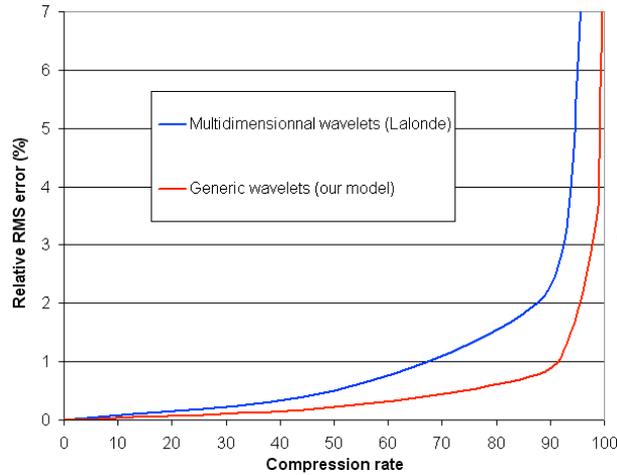
All BRDF measurements are projected on a geodesic sphere at level 4 ( $2^{10}$  triangles in the hemisphere). With this accuracy each triangle covers a solid angle of about 0.006 steradian, which is fine enough for our data sets. The compression *rate*  $t_c$  is the percentage of initial data removed by compression. The compression *ratio*  $r_c$  is the ratio between the initial number of data and the final number of data after compression. They are related by a simple equation:

$$t_c = 100 * \left(1 - \frac{1}{r_c}\right)$$

Usually  $r_c$  is written in the form  $x : 1$ , which means that only one coefficient for  $x$  initial coefficients is kept after compression. The results given in the next three sections show, for each BRDF, compression ratio and corresponding number of coefficients versus total percent error. The error is estimated between the compressed and the initial measurement data set. The initial errors are given for a ratio of 1 : 1 (no compression). They are induced by the projection of the measurement data set on the discrete sphere. For analytical models these errors do not exist because the virtual measurement data set generation is done according to the spherical subdivision.

### 5.2. Isotropic Results

Isotropic BRDFs are simpler than anisotropic ones because they depend only on three angles: both zenith angles and the relative azimuth angle. In fact, only the reflectance of the surface for incoming directions in the plane  $\phi = 0$  is needed. Any BRDF value can be recovered from this information



**Figure 5:** Comparison of our method to the previous work of Lalonde, Phong BRDF with exponent of 30.

**Table 1:** Modelling results for real data sets at fixed wavelength

$r_c$	# coefficient	Spectralon		Cloth		Plastic		Green grass		Dry grass		Sand	
		$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$
1:1	$2^{14}$	0.07	0.33	0.26	1	0.19	1	0.36	1.1	0.45	1.6	0.3	1
2:1	$2^{13}$	0.12	0.33	0.62	1.1	0.26	1	0.5	1.1	0.64	1.6	0.64	1.1
8:1	$2^{11}$	0.61	0.76	1.8	2.3	1	1.5	1.5	2	2.5	3.4	1.3	1.6
16:1	$2^{10}$	1.2	1.4	2.6	3.2	8.1	11	2.9	3.6	4.5	5.5	2.2	2.7
64:1	$2^8$	2.1	2.5	4.2	5.4	36	80	4.4	5.7	7.4	9.2	3.2	3.9
128:1	$2^7$	2.8	3.3	4.5	5.9	45	100	5.3	7.1	8.9	11	3.7	4.6
256:1	$2^6$	2.9	3.5	4.9	6.8	48	112	8.2	12	11	14	6.9	9.6

by a rotation of  $\phi$  around the local surface normal. Only incoming triangles that intersect the plane are valid. With a subdivision level of 4, 16 incoming triangles cut the plane  $\phi = 0$ , which leaves (with the  $2^{10}$  triangles of the outgoing hemisphere) a total of  $16 \times 2^{10} = 2^{14}$  triangles. In this case the spherical wavelet transform on incoming directions is not relevant. Using multiple transforms, partial reconstruction is possible. In the case of isotropic BRDFs, we only apply the transform on outgoing directions and not on incoming ones.

### 5.2.1. Comparison with Previous Work

In order to compare our model performances with the results achieved by Lalonde, we use a virtual measurement data set generated using the Phong BRDF model with an exponent of 30. Lalonde used  $2^5$  samples for each of  $\kappa_i, \lambda_i, \kappa_r, \lambda_r$  ( $2^5 \times 2^5 = 2^{10}$  samples for each hemisphere), which is equivalent to our resolution of 4 that generates  $2^{10}$  triangles for the incoming hemisphere and the outgoing hemisphere. Figure 5 shows the compression rate versus the normalized root mean square error (relative error in  $L_2$ ). As Lalonde

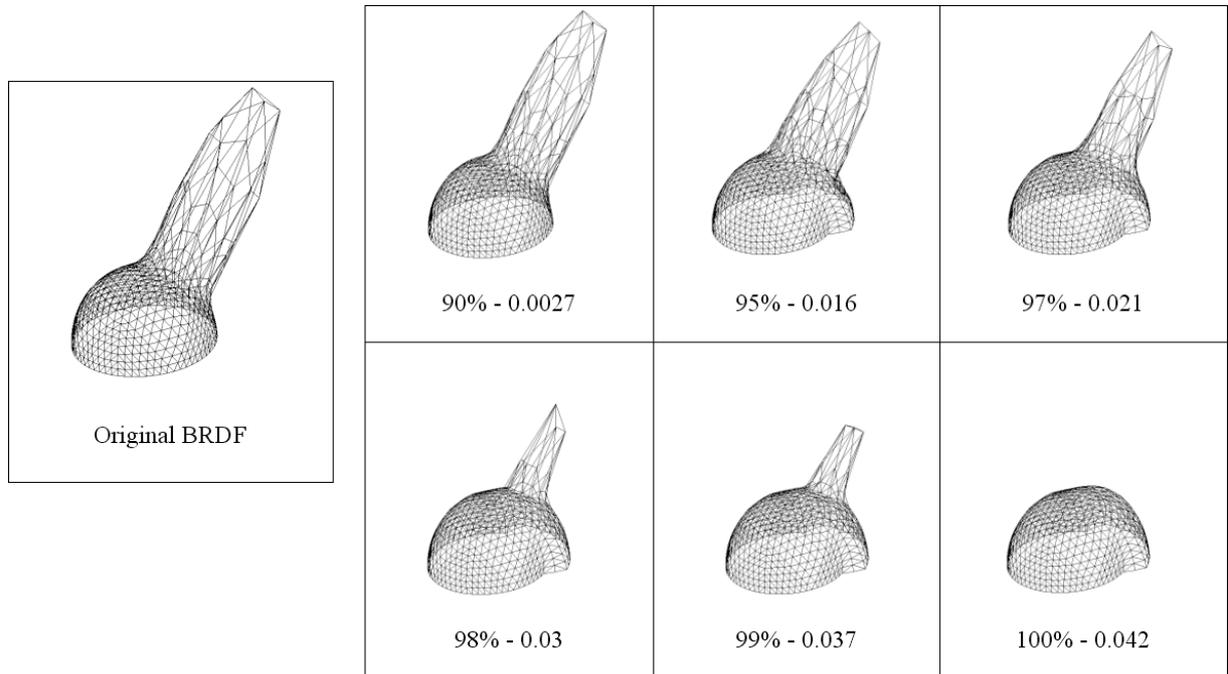
cannot use partial reconstruction, the BRDF is sampled for each incoming and outgoing direction, which gives a total of  $2^{20}$  samples. Thus we use a combination of two transforms (incoming and outgoing hemisphere). In this case, our generic compression is more efficient because a high level compression on incoming directions is efficient. Indeed the data set is isotropic, and as a consequence the BRDF coherence in the incoming hemisphere is very important: the responses for two incoming directions are similar with regard to a rotation of the relative azimuth angle between the directions. The model of Lalonde fails to achieve comparable results because coherence of the dimension-connections is less important here.

### 5.2.2. Monochromatic Results

Compression rates and corresponding model errors for different real surfaces are presented in Table 1. Spectralon, green grass and sand are measured at 800 nm, plastic and cloth at 700 nm, and dry grass at 600 nm. All available measurement data sets are modelled with a global error

**Table 2:** Modelling results for synthetic data sets. Two isotropic Phong BRDFs and a Ward anisotropic BRDF

$r_c$	# coefficient	Phong BRDF (1)		Phong BRDF (2)		# coefficient	Ward BRDF	
		$L_1$	$L_2$	$L_1$	$L_2$		$L_1$	$L_2$
1:1	$2^{14}$	0	0	0	0	$2^{20}$	0	0
2:1	$2^{13}$	0.039	0.089	0.036	0.1	$2^{19}$	0.32	0.076
8:1	$2^{11}$	2	3.1	3.9	7.4	$2^{17}$	3.1	5.4
16:1	$2^{10}$	4	6.2	8.6	15.6	$2^{16}$	10	22
64:1	$2^8$	15	20	51	69	$2^{14}$	16	28
128:1	$2^7$	16	21	49	65	$2^{13}$	16	30
256:1	$2^6$	17	23	50	65	$2^{12}$	17	31

**Figure 6:** 3D view of a Phong BRDF for an incident angle of  $50^\circ$ . Compression rates and corresponding RMS errors are given.

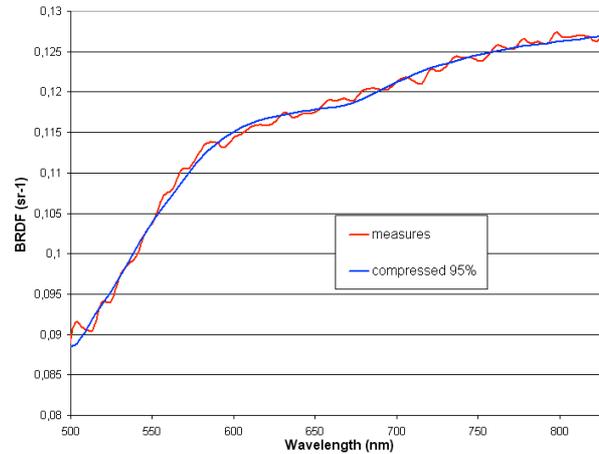
$< 5\%$  and a compression rate better than 90% (10:1). The plastic BRDF is the only one to present large errors for high compression rates. This BRDF is very specular: diffuse and specular parts differ by a factor 100, with a localized narrow peak creating a big discontinuity. With a 'medium' compression ratio ( $\sim 30:1$ ) the specular peak is not affected by compression, while the diffuse part is. But at high compression rates, the specular peak is compressed too, resulting in a smoothing and a lifting of the values around the peak support (because the wavelet transform is basically a convolution). The results presented in Table 2 (case 1 and 2) show the differences between a glossy BRDF and a very specular BRDF. The virtual measurement data sets were generated using the physical Phong reflection

model [27]. Parameters are respectively  $k_d=0.75$ ,  $k_s=0.25$ ,  $n=20$ , and  $k_d=0.5$ ,  $k_s=0.5$ ,  $n=50$ . The effect of 'lifting' can be viewed on the 3D view of the compression process of the first specular BRDF in Figure 6. Increasing specularity implies less effective compression with a given resolution. But the efficiency can be recovered by increasing the spherical subdivision. Indeed the specular peak is then more detailed, and discontinuities are less important. We are certainly limited by our goniometer sampling precision for real measurement data sets, this is not the case with virtual measurement data sets.

After the plastic BRDF, the worst results are obtained for the grass, which is the measurement data set with the lowest

**Table 3:** Modelling results for real spectral data sets

$r_c$	# coefficient	Spectralon		Mélaminé		Plastic		Sand		Cloth	
		$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$	$L_1$	$L_2$
1:1	1,228,800	0.078	0.36	0.26	1.2	0.42	2.9	0.25	1	0.53	2.8
2:1	614,400	0.19	0.4	0.39	1.3	0.7	2.9	0.47	1	1.3	3.2
8:1	153,600	0.29	0.47	0.61	1.3	1.8	3.6	0.79	1.2	1.9	4.2
16:1	76,800	0.36	0.56	0.89	1.5	2	3.7	1	1.6	6.6	11
64:1	19,200	1	1.2	4.6	5.9	4.8	9	1.3	1.8	6.2	11
128:1	9,600	1.1	1.4	7.5	9.8	20	37	2.1	2.7	8.1	14
256:1	4,800	1.1	1.4	9.9	13	21	41	3.3	4.2	9.5	16

**Figure 7:** A close-up view of the spectral sand BRDF,  $\theta_i = \theta_r = \pi/3$ ,  $\phi = \pi$ . Wavelet compression removes instrumental noise.

BRDF magnitude ( $\sim 0.03 \text{ sr}^{-1}$ ). The measurement quality is certainly poorer and introduces more noise in this case. As wavelet compression is also a well-known denoising technique, an hypothesis is that the noise was removed by modelling. So a direct comparison to the measured BRDF shows bad results because of the noise. But data smoothing is not necessarily a bad thing.

### 5.2.3. Spectral Results

As explained in Section 4, anisotropic spectral BRDF combines three atomic transforms. In the case of isotropic data, two transforms remain (outgoing and spectral transforms), each with its own compression threshold. A good compromise must be found between geometrical and spectral compression accuracy. Two different sets of thresholds could induce the same global error, but one is more accurate geometrically and the other one is more accurate spectrally. This is the critical point of our model, there is not a unique compression solution for a given global error. We have implemented 52 different wavelet bases for the spectral compo-

nent in our software. Most of them belong to the same family (Daubechies, Symmlet, Coiflet, etc.). The best basis in term of error in  $L_2$ , according to a given threshold, is chosen at compression time.

The results of Table 3 were achieved for the spectral range 430–830 nm sampled every 5 nm. The spectral data sets were constructed by a decimation of the original data sets from our goniometer. Indeed its spectral accuracy is too high for the usual spectral consideration in rendering (according to CIE specifications). Results are better than in the monochromatic case because our data are smoother spectrally than geometrically. Indeed, natural and artificial surfaces do not have a complex spectral behaviour in the visible spectrum. Modelling is good (global errors often  $< 5\%$ ) for all surfaces, with a compression rate  $> 95\%$  (20:1). Wavelet denoising is a very interesting property because our measurement data sets are noisy. Even if the modelling error grows with compression rate, we can suppose that we find the best BRDF profile by removing noise (Figure 7).

**Table 4:** The real memory cost of our model for a compression ratio of 256 : 1

Case BRDF State	Monochromatic isotropic Spectralon			Monochromatic anisotropic Ward			Spectral isotropic Spectralon		
	Data	Strips	Total	Data	Strips	Total	Data	Strips	Total
Initial									
	16,384	2088		1,048,576	133,250		3,281,920	411,688	
(kB)	128	84	237	8192	5215	15,136	25,040	16,100	46,109
Compressed									
	64	23		4444	2225		13,966	8855	
(kB)	0.5	0.9	3	35	89	163	109	355	617

### 5.3. Anisotropic Results

In order to test anisotropic BRDFs, we have generated a virtual measurement data set using the Ward anisotropic BRDF model [28]. For results presented in Table 2, model parameters are: X-roughness=0.35, Y-roughness=0.05, kd=0.75, ks=0.25. The remark raised in Section 5.2.2 about specular surfaces is also relevant here. But we can see that the same order of error than in the Phong model case is reached according to compression ratio. So, introducing anisotropy does not reduce our model accuracy.

### 5.4. Memory Considerations

As yet we have only presented compression rates or ratios in terms of measurement points. But as explained before, data are only one component of the model. The other one is the localization of the data. In Table 4 we present the real memory requirement of our model for three different cases with a compression ratio equal to 256 : 1. The first column of the table gives the number of measurement data points, while the second gives the number of strips needed to store them in a Sparse Array structure. The total cost of the model is computed by adding data, strips, and some extra memory cost due to our actual implementation (which should be removed in a production system). The first row is the initial memory state, and the second after compression. In the example, the total memory use is 46 MB for the initial spectral spectralon BRDF, reduced to 617 kB after compression. In a similar way, Ward BRDF uses 15 MB before compression, and only 163 kB afterwards. Before compression, data use more memory than localization, but after compression, localization uses more memory than data. This is why the localization structure must be very flexible as discussed in Section 4.3.

We have presented here memory costs for a radical compression ratio. But not all BRDFs can be compressed until this point because the modelling error is too large. This is particularly the case of specular BRDFs with non-appropriate subdivision level (see Section 5.2.2). However,

memory saving is very important even if the compression ratio is in the order of 64 : 1 (memory use is about four times greater than the numbers given above).

## 6. Application

### 6.1. Wavelet-based Importance Sampling

Our BRDF representation can be directly used in radiative transfer codes or for physical database compression, but it is not sufficient. Indeed accurate simulation often means *inversion*. In particular physically based rendering solves the rendering equation (Equation 9) [2] using Monte Carlo methods, which can handle all luminous transfers (diffuse reflection, specular reflection, caustics, but also participating media). The reader is referred to [29] for a general dissertation on physically based rendering, Monte Carlo algorithms and their improvement.

$$L_r(x, \omega_r, \lambda) = L_e(x, \omega_r, \lambda) + \int_{\Omega_i} f_r(x, \omega_i, \omega_r, \lambda) L_i(\omega_i, \lambda) \cos \theta_i d\omega_i \quad (9)$$

Basically Monte Carlo methods provide an estimate of the integral by generating a sequence of random samples in the integration domain (here the surrounding hemisphere of the surface point  $x$ ), and computing their average. The major problem is the noise introduced by this stochastic process. Numerous techniques exist to reduce the variance. The most powerful is known as *importance sampling*. The idea is to generate samples according to the magnitude of the function we want to integrate. One can understand that it is better to generate samples where the function is important, because at these places, the integral is also important. In general, mathematical inversion of the function under consideration cannot be done, and other techniques are required. In this section, we present an extension to the work of Lalonde and Fournier [30] that allows fast numerical inversion of the BRDF using our wavelet-based representation.

Although we could generate samples directly from our BRDF representation, we prefer to use a different structure. Two reasons motivate this choice. First, the BRDF must be spectrally integrated because we are looking for directions only important in terms of global energy. It would be very expensive to do spectral integration on the fly. Secondly, we want to use importance sampling when modelling is done through an analytical model as well. So we need a standalone structure that can be used independently of the wavelet BRDF representation. Let  $f_r^\lambda$  be the spectral integration of the BRDF:

$$f_r^\lambda = \int_{\lambda} f_r(\omega_i, \omega_r, \lambda) d\lambda$$

In fact, a monochromatic BRDF structure is used to store  $f_r^\lambda$  because integration reduces the spectral BRDF to a function from  $\mathbf{S}^2 \times \mathbf{S}^2$  to  $\mathbf{R}$ . This structure can naturally be compressed, and certainly much more than a real BRDF because less precision is required (see Section 7). To optimize the sampling, we include the cosine term of the rendering equation [31], and work with projected solid angle.

Consider a strictly positive piecewise constant function  $f : [a, b] \mapsto \mathbf{R}$ . Pieces  $p_i$  are defined over  $[x_i, x_{i+1}]$  and valued  $v_i$ ,  $i \geq 0$ . Suppose we want to generate a random sample  $x$  according to  $f$ . One way to proceed is to choose first a piece according to its importance, and then generate a random sample uniformly in the piece. The probability to select  $p_k$  is:

$$\begin{aligned} P(p_k) &= \frac{v_k(x_{k+1} - x_k)}{\int_a^b f dx} \\ &= \frac{v_k(x_{k+1} - x_k)}{\sum_i v_i(x_{i+1} - x_i)} \end{aligned}$$

The probability to select any  $x$  uniformly in  $p_k$  is:

$$P(x \in p_k) = \frac{1}{(x_{k+1} - x_k)}$$

The final probability for  $x$  is:

$$\begin{aligned} P(p_k) * P(x \in p_k) &= \frac{v_k(x_{k+1} - x_k)}{(x_{k+1} - x_k) \sum_i v_i(x_{i+1} - x_i)} \\ &= \frac{v_k}{\sum_i v_i(x_{i+1} - x_i)} \end{aligned}$$

The practical way to select our  $p_k$  is to choose a random number:

$$y \in [0, \int_a^b f dx]$$

And then looking for the first piece  $p_k$  (or abscissa  $x_{k+1}$ ) for which we have:

$$y < \int_{x_0}^{x_{k+1}} f dx$$

With our representation we have a piecewise constant function representing the BRDF on the sphere. So we can do exactly the same thing and generate a random number:

$$y \in [0, \int_{\Omega_r} f_r^\lambda(\omega_i, \omega_r) d\omega_r]$$

Then we choose a level of accuracy, say  $j$ , and starting from the first triangle  $T_j^0$  at level  $j$ , we accumulate the integral value of the function until we found  $T_j^k$  for which:

$$y < \int_{\cup_{l=0}^{l=k} T_j^l} f_r^\lambda(\omega_i, \omega_r) d\omega_r$$

The algorithm cost is  $O(n)$ , where  $n$  is the number of triangles (or samples) at level  $j$ . Using multiresolution the cost is reduced to  $O(\log_4 n)$ .

The Bio-Haar basis, which could seem inefficient, has a useful property that allows us to save computation time as shown by Lalonde and Fournier [30]. We adapted this method to our spherical representation. Each triangle value at level  $j$  is the average of its children values at level  $j+1$ . As the function is assumed to be constant over the triangles, we have:

$$\int_{T_j^k} f_r^\lambda(\omega_i, \omega_r) d\omega_r = \sum_{l=0}^{l=3} \int_{T_{j+1}^l} f_r^\lambda(\omega_i, \omega_r) d\omega_r$$

where  $T_{j+1}^l$  is a child of  $T_j^k$

This property can increase the search speed of our  $T_j^k$ , and allows a recursive search. Indeed we can start from level 0, and accumulate integral value of  $f_r^\lambda$  over the triangles of the tetrahedron. First we find the triangle  $T_0^u$  for which:

$$y < \int_{\cup_{l=0}^{l=u} T_0^l} f_r^\lambda(\omega_i, \omega_r) d\omega_r$$

Then we recursively examine its children, until the accuracy level  $j$  required, to find  $T_j^k$ . The algorithm presented in Figure 8 performs the recursive search of the reflected direction and compute its final probability value (*pdf* or probability density function value). Each triangle defines an associated solid angle, which is used for integral computations.

## 6.2. Results

We have incorporated the wavelet BRDF model in our global illumination software called *Ray Of Light*. This engine is a high-level platform with abstraction of the surface definitions, physical material properties, and rendering techniques, allowing a simple integration of our model. All images are rendered using the three standard R,G,B

```

function sample(Direction  $\omega_i$ , Integer level) : return Direction  $\omega_r$ , Real pdf
{
  Real reflectance =  $\int_{\Omega_r} f_r(x, \omega_i, \omega_r) \cos \theta_r d\omega_r$ ;
  Real y = random(0, reflectance)
  Real total = 0
  Boolean found = false
  Integer i = 0
  while ( (not found) and (i < 4) )
  {
    recursive_sample( $\omega_i, T_0^i$ , reflectance, y, total, 0, level, found,  $\omega_r$ , pdf)
    i++
  }
}
function recursive_sample(Triangle T, Real reflectance, Real y, Real total,
Integer current_level, Integer end_level, Boolean
found) :
  return Direction  $\omega_r$ , Real pdf
{
  if (found)
    return
  else if (current_level == end_level) then
    if (y < total * T.value * T.d $\omega$ ) then
      for i = 0..4
        recursive_sample(T.child[i], reflectance, y, total,
          current_level + 1, end_level, found)
    else
      total += T.value * T.d $\omega$ 
  else
    {
      total += T.value * T.d $\omega$ 
      if (y < total) then
        {
          found = true
           $\omega_r = T.random\_point()$ 
          pdf =  $\frac{T.value * T.d\omega}{reflectance} * \frac{1}{T.area}$ 
        }
      }
  }
}

```

**Figure 8:** Wavelet importance sampling.

wavelengths as spectral rendering is still in development. We used three compressed monochromatic BRDFs for each surface.

To perform a qualitative comparison we generate images of Figure 9 using Monte Carlo path tracing with the same number of rays per pixel (1000 in this case). The sphereflake surface is modelled with a metallic Phong BRDF, while the walls are diffuse. From left to right we used uniform sampling, analytical importance sampling, and a wavelet importance sampling. The first method provides a uniformly noisy image. The variance is globally high. Analytical sampling of the Phong BRDF model provides a locally noisy image. The variance is low most of the time but some pixels present high variance. Wavelet importance sampling has neither problem. The variance is stable on the image. The qualitative impression is confirmed by quantitative results presented in Table 5. The problem of analytical importance sampling is induced by the assumption that the BRDF is *separable*, i.e. composed of a diffuse term and a specular term. Indeed, a perfect specular term increases the global variance because of the presence of a high-valued narrow peak. If the stochastic process is stopped

**Table 5:** Quantitative comparison for the Phong metallic BRDF

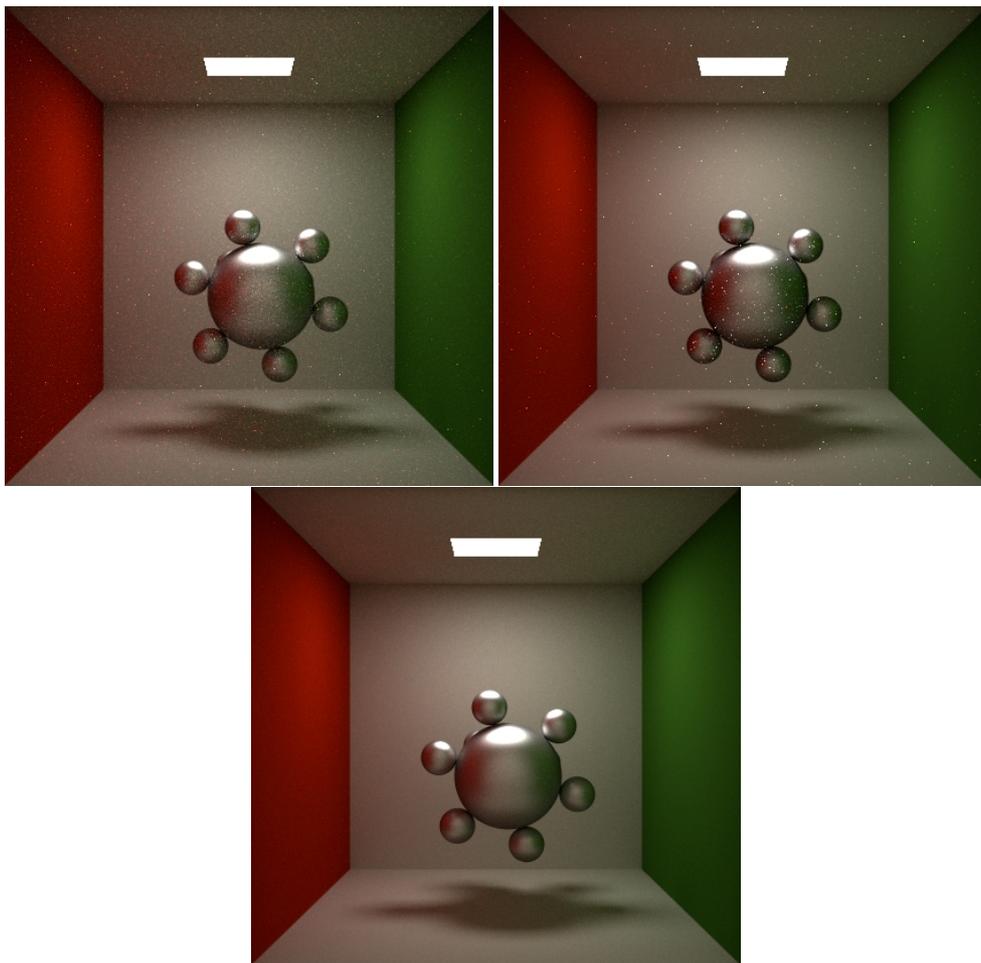
Sampling	Primary rays	Estimated variance
Uniform	1000	0.0094 (reference)
Analytical importance	1000	0.0074 (-21%)
Wavelet importance	1000	0.0062 (-34%)

early (with few samples), some estimators ‘see’ only the specular part because a large number of samples has to be generated before one sample appears in the diffuse part. This is why less precision for our importance sampling structure is actually an advantage. On the one hand, radical wavelet compression smooths narrow peaks. On the other hand, if the accuracy level is not very high, the triangle area is large enough to enclose the narrow specular peak of the BRDF. As the final reflected direction is chosen uniformly in the selected triangle, a smoothing is also performed at this stage. So if the wavelet importance sampling level is adapted to the BRDF type, the sample generation can be optimized in terms of performance (stopped early in the recursive procedure) and quality (variance). The important thing to notice is that even if we use a less precise structure for reflected directions generation, the real BRDF values (from wavelet representation or analytical model) are used for computations to avoid error addition during the process.

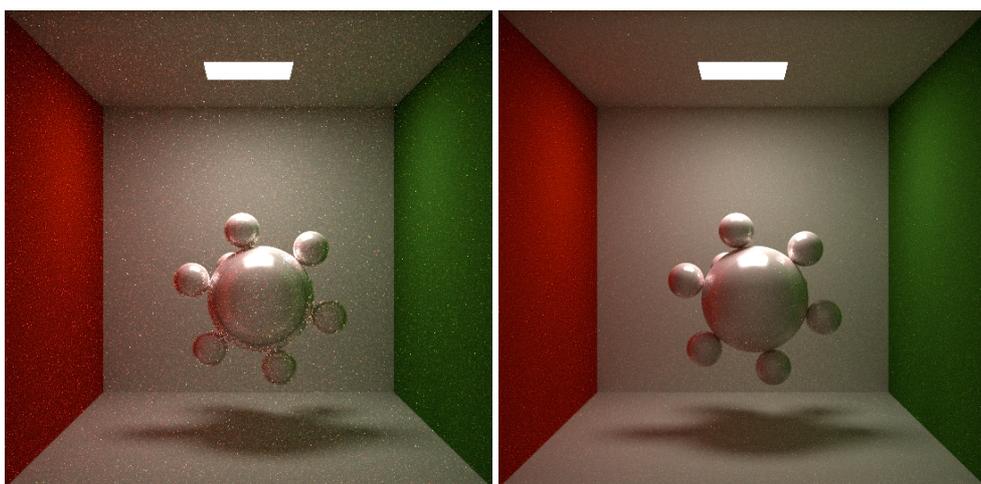
Figure 10 shows again two sphereflakes but using our real plastic BRDF measurement (compression rate is 95%). The first one uses uniform sampling, and the second one wavelet importance sampling at level two. The same number of rays is used: 512 per pixel.

Finally a room where four objects use BRDF measurement data set is presented in Figure 11. The material of the table is mélaminé, that of the textured walls is spectralon, the teapot uses the plastic BRDF and the sofa the red cloth BRDF. There are 128 rays per pixel. All BRDFs are also compressed with a rate of 95%. A caustic is induced by the teapot on the table. A more accurate rendering (512 rays per pixel) of the room is shown in Figure 12. A second caustic appears on the table because of the glass sphere (perfect specular analytical BTDF). Persistent localized noise is also introduced by the glass sphere (perfect specular analytical sampling).

To perform a more significant comparison, we present a quantitative comparison in the case of the room scene. This is a particular example but a representative one composed of 6000 polygons and four different physical BRDFs. Table 6 shows, for each sampling method, the measured variance on the image and corresponding rendering time according to the number of primary rays. Our method consists of the estimation of the local variance for each pixel using the eight nearest neighbours, then in computing the image variance



**Figure 9:** *Metallic BRDF: uniform sampling, analytical importance sampling and wavelet importance sampling.*



**Figure 10:** *Plastic BRDF: uniform sampling and wavelet importance sampling.*



**Figure 11:** Room scene: uniform sampling and wavelet importance sampling, 128 rays per pixel.

**Table 6:** Quantitative comparison for the room scene

Sampling	Primary rays	Estimated variance	Rendering time
Uniform	128	0.043 (reference)	reference
Wavelet importance	128	0.024 (−45%)	+27%
Wavelet importance	32	0.039 (−10%)	−69%

as the mean of local variances. For a given image quality (or variance), wavelet importance sampling reduces rendering time in comparison with uniform sampling. Indeed, the rendering time reduction induced by ray saving is more important than the cost added by sample generation. Moreover in the room scene only the plastic BRDF and the mélaminé BRDF are specular. Other BRDFs are almost diffuse. Variance reduction should be more important in the case of many specular BRDFs.

## 7. Conclusions and Future Work

We have presented a novel BRDF numerical model based on wavelet theory, which is extensible and general. In fact, the model can represent any function defined on a Cartesian product of  $\mathbf{S}^2$  and  $\mathbf{R}$ , and therefore isotropic, anisotropic and/or spectral BRDF. The model compression is not globally controlled, but for each space, allowing a good control of the space/accuracy trade-off. Error is always reasonable, even for radical compression. We showed that partial reconstruction is useful in the case of isotropic BRDFs. This property should also be very useful to speed

up calculations in a path tracing algorithm with anisotropic BRDFs. Indeed the same incoming direction is used to generate samples in the reflected hemisphere. If we use a compressed BRDF, the reconstruction of the signal for the given direction results in a compressed Reflectance object, encoding the value of the BRDF for each outgoing direction. Therefore, only fast spherical inverse transform on outgoing directions is computed to obtain a BRDF value. With a non-standard approach, the complete inverse transform is performed for each BRDF evaluation even if the incoming direction does not change. Because there is only one transform using multidimensional basis functions. So, even if the non-standard decomposition is usually more efficient than the standard one [13] as noticed by Lalonde, it is not the case when we can exploit partial reconstruction. Finally, an importance sampling method to generate reflected ray directions according to the BRDF has been explained and optimized to save computation time using multiresolution. This technique reduces the variance of Monte Carlo path tracing algorithms, and therefore rendering time, and it is usable with our BRDF representation and with analytical models as well.

We leave as future work the extension of our simulation tool to represent spectral radiance and spectral emission functions using wavelets. Rendering time will be largely reduced by operating on compressed spectra because accurate spectral representation computations are very expensive. Future research will also test our model with spectral anisotropic BRDF measurement data sets. Finally, we plan to integrate multiscale representation of the BRDF, i.e. BRDF measured at different spatial scales (for example with a goniometer, a plane and a satellite). This is useful for remote sensing databases or applications.



**Figure 12:** Room scene: wavelet importance sampling, 512 rays per pixel

### Acknowledgements

We specially thank Dr N.A. Dodgson for his contribution to the final version of this paper.

### References

1. S. Chandrasekhar. *Radiative Transfer*. Clarendon Press, Oxford, 1950.
2. James T. Kajiya. The rendering equation. In *Computer Graphics (ACM SIGGRAPH '86 Proceedings)*, vol. 20(4), pp. 143–150. August, 1986.
3. Eric P. Lafortune, Sing-Choong Foo, Kenneth E. Torrance and Donald P. Greenberg. Non-linear approximation of reflectance functions. In *Computer Graphics (ACM SIGGRAPH '97 Proceedings)*, vol. 31(3), pp. 117–126. 1997.
4. F. E. Nicodemus, J. C. Richmond, J. J. Hsia, I. W. Ginsberg and T. Limperis. Geometric considerations and nomenclature for reflectance. Monograph 161. National Bureau of Standards (US), October, 1977.
5. M. Minnaert. The principle of reciprocity in lunar photometry. *Astrophysical Journal*, 93:403–410, 1941.
6. Peter Schröder and Wim Sweldens. Spherical wavelets: efficiently representing functions on the sphere. In *Computer Graphics Proceedings, Annual Conference Series, 1995 (ACM SIGGRAPH '95 Proceedings)*, pp. 161–172. 1995.
7. Michael Lounsbery, Tony D. DeRose and Joe Warren. Multiresolution analysis for surfaces of arbitrary topological type. *ACM Transactions on Graphics*, 16(1):34–73, January, 1997.
8. Steven J. Gortler, Peter Schroder, Michael F. Cohen and Pat Hanrahan. Wavelet radiosity. In *Computer Graphics Proceedings, Annual Conference Series, 1993 (ACM SIGGRAPH '93 Proceedings)*, pp. 221–230. 1993.
9. Per H. Christensen, Eric J. Stollnitz, David H. Salesin and Tony D. DeRose. Global illumination of glossy environments using wavelets and importance. *ACM Transactions on Graphics*, 15(1):37–71, January, 1996.
10. R. DeVore, B. Jawerth and B. Lucier. Image compression through wavelet transform coding. *IEEE Transactions on Information Theory*, 38(2):719–746, 1992.
11. G. Beylkin, R. Coifman and V. Rokhlin. Fast wavelet transforms and numerical algorithms I. *Communications on Pure Applied Mathematics*, 44(2):141–183, 1991.

12. S. Mallat. In *A Wavelet Tour of Signal Processing*. Academic Press, San Diego, 1999.
13. I. Daubechies. Ten lectures on wavelets. Philadelphia: Society for Industrial and Applied Mathematics, 1992.
14. Eric J. Stollnitz, Tony D. DeRose and David H. Salesin. Wavelets for computer graphics: theory and applications. Morgan Kaufmann, San Francisco, CA, 1996.
15. B. Cabral, N. Max and R. Springmeyer. Bidirectional reflection functions from surface bump maps. In *Computer Graphics (ACM SIGGRAPH '87 Proceedings)*, vol. 21, pp. 273–281. 1987.
16. M. Girardi and W. Sweldens. A new class of unbalanced haar wavelets that form an unconditional basis for  $L_p$  on general measure spaces. 1995:2, Departement of Mathematics, University of South Carolina, Carolina, May, 1995.
17. M. Mitrea. Singular integrals, hardy spaces and clifford wavelets. *Communications of the ACM*, 32(6):343–349, 1980.
18. Wim Sweldens. The lifting scheme: a construction of second generation wavelets. *SIAM Journal on Mathematical Analysis*, 29(2):511–546, 1998.
19. P. Lalonde and A. Fournier. A wavelet representation of reflectance functions. *IEEE Transactions on Visualization and Computer Graphics*, 3(4):329–336, August, 1997.
20. Robert R. Lewis and Alain Fournier. Light-driven global illumination with a wavelet representation. In *Rendering Techniques '96 (Proceedings of the Seventh Eurographics Workshop on Rendering)*, Springer, New York, pp. 11–20. 1996.
21. J. M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. *IEEE Trans. SP*, 41:3445–3462, 1993.
22. W. H. Press, S. A. Teukolsky, W. T. Vetterling and B. P. Flannery. Numerical recipes in C, the art of scientific computing, 2nd edition. Cambridge University Press, Cambridge, 1992.
23. David Salesin, Daniel Lischinski and Tony DeRose. Reconstructing illumination functions with selected discontinuities. In *Third Eurographics Workshop on Rendering*, Bristol, pp. 99–112. May, 1992.
24. G. Serrot, M. Bodilis, X. Briottet and H. Cosnefroy. Presentation of a new BRDF measurement device. *Proceeding SPIE*, 3494:34–40, 1998.
25. M. Robart, M. Paulin and R. Caubet. A realistic material model for reflectance simulation. In *International Conference on Information Visualisation IV Proceedings*, pp. 205–210. 1998.
26. Stephen H. Westin, James R. Arvo and Kenneth E. Torrance. Predicting reflectance functions from complex surfaces. In *Computer Graphics (ACM SIGGRAPH '92 Proceedings)*, vol. 26(4), pp. 255–264. July, 1992.
27. Eric P. Lafortune and Yves D. Willems. Using the modified phong BRDF for physically based rendering. Technical Report CW197, Department of Computer Science, Katholieke Universiteit Leuven, Leuven, Belgium, November, 1994.
28. Gregory J. Ward. Measuring and modeling anisotropic reflection. In *Computer Graphics (ACM SIGGRAPH '92 Proceedings)*, vol. 26(4), pp. 265–272. July, 1992.
29. Eric Lafortune. Mathematical Models and Monte Carlo Algorithms for Physically Based Rendering, Ph. D. thesis, Leuven, Belgium, Department of Computer Science, Katholieke Universiteit Leuven, February, 1996.
30. Paul Lalonde and Alain Fournier. Generating reflected directions from BRDF data. *Computer Graphics Forum (Eurographics '97 Proceedings)*, 16(3):C293–C300, 1997.
31. Eric P. Lafortune and Yves D. Willems. A 5D tree to reduce the variance of monte carlo ray tracing. In P. M. Hanrahan and W. Purgathofer (eds), *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, Springer, New York, pp. 11–20. 1995.