Volume 13, (1994) number 2, pp. 97-107

COMPUTER GRAPHICS *forum*

A Theoretical Framework for Physically Based Rendering

Eric P. Lafortune and Yves D. Willems

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

Abstract

In this paper we introduce the concept of the global reflection distribution function which allows concise formulation of the global illumination problem. Unlike previous formulations it is not geared towards any specific algorithm. As an example of its versatility we derive a Monte Carlo rendering algorithm that seamlessly integrates the ideas of shooting and gathering power to create photorealistic images.

Keywords: Rendering and visualisation; Global illumination; Photorealistic rendering

1. Introduction

The global illumination problem consists of determining the lighting functions for a given scene by simulating the transport of light throughout the scene. The starting point for finding a physically correct solution consists in identifying the quantities and properties that are relevant to the problem. Using these notions a mathematical model can be constructed, which formally expresses the physical laws of illumination, reflection, refraction and everything else encompassing the transport of light. The complexity of the problem most often forces simplifying assumptions. The final step is to derive practical algorithms to solve the mathematically formulated problems. A concise mathematical formulation allows standard analytical and numerical techniques to be applied to solve the problem. In computer graphics the solution of the global illumination problem is mostly used to create a realistic rendering of the scene. In this case the solution can be tuned for one or more specific views.

In the following paragraphs we will introduce a new formulation of the global illumination problem. The medium is assumed to be non-participating; only surfaces interact with light, by reflection, refraction and absorption. We will then present a new rendering algorithm and show how it can be derived from the mathematical formulation.

2. Related Work

An important milestone in the development of the global illumination theory for computer graphics was the introduction of the radiosity method by Goral et al.[1]. Originally developed within the field of heat transfer it is based on the energy equilibrium between diffuse emitters and reflectors of radiative energy such as heat or light. Although restricted to the simulation of diffuse lighting effects it presents a sound physical model. Discretisation of the scene in patches or elements results in a mathematical model consisting of a set of linear energy equations. The coefficients are determined by the geometry and the reflective properties of the scene, the right-hand side coefficients are the self-emitted radiosities of the elements and the unknowns are the radiosities of the elements. Various algorithms have been presented to solve this set of equations. The radiosity solution is view-independent. Smits et al.[2] introduced the notion of importance and adapted the radiosity algorithm to tune the solution with respect to the final rendering parameters.

The rendering equation as proposed by Kajiya[3] provides a more general mathematical formulation of the problem that is no longer restricted to diffusely emitting surfaces. It can be expressed as a Fredholm equation of the second kind where the kernel is determined by

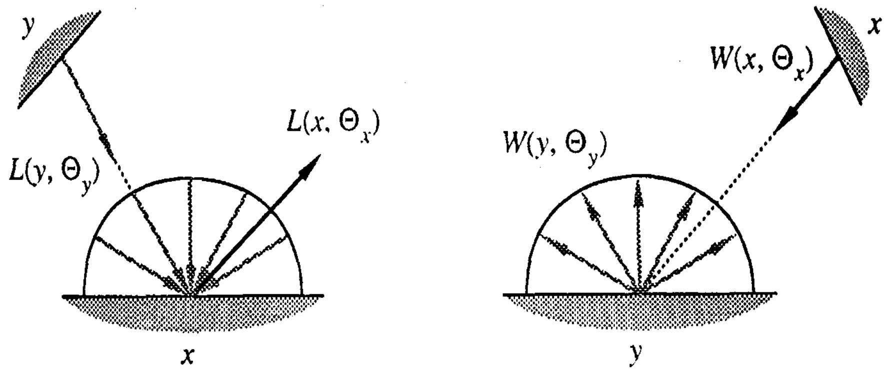

Figure 1: *Explanation of the symbols used in the rendering equation and the potential equation respectively.*

the geometry and the reflective properties of the scene. The unknown function is the radiance across the surfaces of the scene. Path tracing is presented as a Monte Carlo algorithm that solves the rendering equation. The idea is to sample the flux through the pixels, gathering light by following all light paths back to the light sources. As such it is entirely view-dependent. Various other algorithms are based on the same principle[4-9].

Pattanaik[10-12] recently presented the potential equation, which is adjoint to the rendering equation. The notions of radiance and potential capability, and the set of equations form a mathematical framework that explains all common global illumination algorithms. He proposed an accompanying Monte Carlo particle tracing algorithm that shoots light particles from the light sources[13]. The scene is discretised into elements at which received and diffusely emitted power is accumulated during the simulation. In a second view-dependent pass, path tracing or distribution ray tracing is used to render the scene, adding specular effects to the already computed diffuse components.

Most other algorithms that solve the global illumination problem are similarly based on two-pass methods[14-17]. They compute diffuse lighting components in a first extended radiosity pass and specular lighting components in a second view-dependent pass. Some algorithms try to represent the complete radiance function instead of only its diffuse component in order to find a view-independent solution. The main problem with this approach is the huge amount of storage that is required to represent the lighting function. Aupperle and Hanrahan[18] and Christensen et al.[19] tried to meet this problem using the notion of directional importance, explained in a mathematical framework similar to Pattanaik's. Their deterministic algorithms concentrate on important contributions to the solution and thereby strongly reduce the storage requirements and the computational times.

3. The Rendering Equation and the Potential Equation

We briefly recall the rendering equation and the potential equation which were presented by Pattanaik[10-12] as a set of adjoint equations, both describing the global illumination problem from a different point of view. The rendering equation expresses the emitted radiance at a given point on a surface and in a given direction as the sum of the self-emitted radiance and of the reflected radiance resulting from incoming light from other surfaces (Figure 1):

$$L(x,\Theta_x) = L_e(x,\Theta_x) + \int_{\Omega_x^{-1}} f_r(x,\Theta_y,\Theta_x)L(y,\Theta_y)\,|\Theta_y \cdot N_x|\,d\omega_y$$

where:

- $L(x,\Theta_x)$ = the emitted radiance at point x along direction Θ_x $[W/m^2sr]$,
- $L_e(x,\Theta_x)$ = the self-emitted radiance at point x along direction Θ_x $[W/m^2sr]$,
- $f_r(x,\Theta_y,\Theta_x)$ = the bi-directional reflection distribution function *(brdf)* at point x for light coming in from direction Θ_x and going out along direction Θ_y $[1/sr]$,
- y = the point that 'sees' point x along direction Θ_x,
- $|\Theta_y \cdot N_x|$ = the absolute value of the cosine of the angle between the direction Θ_y and the normal to the surface at point x,
- Ω_x^{-1} = the set of incoming directions around point x (a sphere or a hemisphere),
- $d\omega_y$ = a differential angle around Θ_y $[sr]$.

The flux of light leaving a given set S, consisting of pairs of points on a surface and directions around these points in which one is interested, can then be expressed by:

$$\Phi = \int_A \int_{\Omega_x} W_e(x, \Theta_x) L(x, \Theta_x) |\Theta_x \cdot N_x| \, d\omega_x d\mu_x$$

where:

- $W_e(x, \Theta_x) = 1$ if (x, Θ_x) belongs to the set S, and 0 otherwise,
- A = the set of all surface points of the given scene,
- $d\mu_x$ = a differential surface area element around point **x** on surface A.

Sets S in which one is commonly interested are the points and directions associated with the patches in a scene for radiosity methods, ana me points ana directions seen through the pixels for ray tracing techniques.

The potential capability $W(x, \Theta_x)$ is the dual of the radiance function $L(x, \Theta_x)$. It is defined as the differential flux through a given set S of points and directions, resulting from a single unit of emitted radiance from point x along direction Θ_x. Although not presented as such earlier, this definition can be expressed formally as:

$$\begin{aligned} W(x, \Theta_x) &= \frac{\partial^2 \Phi_{output}}{L_{input}(x, \Theta_x) \partial\omega_x \partial\mu_{x\perp}} \\ &= \frac{\partial^2 \Phi_{output}}{L_{input}(x, \Theta_x) \partial\omega_x \, |\Theta_x \cdot N_x| \, \partial\mu_x} \end{aligned}$$

The potential equation that governs the potential capability expresses that the potential of a point and a direction to light the given set results from a direct contribution if the pair belongs to the set and from an indirect contribution through reflection on the surface to which it points (Figure 1). Formally:

$$\begin{aligned} W(x, \Theta_x) = {} & W_e(x, \Theta_x) \\ & + \int_{\Omega_y} f_r(y, \Theta_x, \Theta_y) W(y, \Theta_y) |\Theta_y \cdot N_y| \, d\omega_y \end{aligned}$$

where:

- y = the point seen by point **x** along direction Θ_x,
- Ω_y = the set of outgoing directions around point y.

The flux can now be expressed in terms of the potential capability:

$$\Phi = \int_A \int_{\Omega_x} L_e(x, \Theta_x) W(x, \Theta_x) |\Theta_x \cdot N_x| \, d\omega_x d\mu_x$$

4. The Global Reflection Distribution Function

From the previous paragraph it is clear that the radiance function $L(x, \Theta_x)$ is defined with respect to a fixed self-emitted radiance function $L_e(x, \Theta_x)$, and that the potential function $W(x, \Theta_x)$ is defined with respect to a fixed function $W_e(x, \Theta_x)$ corresponding to the set S. The relationship between $L(x, \Theta_x)$ and $W(x, \Theta_x)$ is not

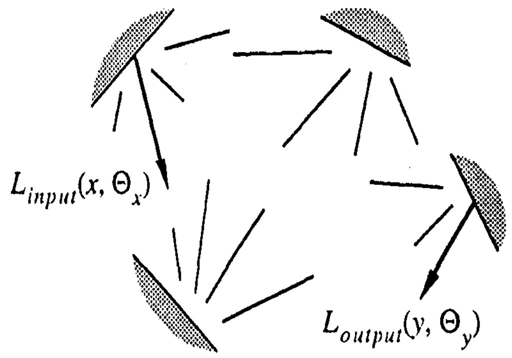

Figure 2: *The global reflection distribution function gives a measure of the fraction of the radiance emitted at point x along direction Θ_x that is eventually output at point y along direction Θ_y through all possible reflections throughout the scene.*

defined by the expressions, making it difficult to combine them into some model or algorithm. We therefore propose the notion of the *global reflectance distribution function (grdf)* which is defined with respect to a given scene but which does not depend on any particular emission function such as $L_e(x, \Theta_x)$ nor on a fixed set such as S.

4.1. Definition

The definition bears some similarity to the approach often taken in system theory, where a system is studied on the basis of its response to some input signal. The grdfexpresses the differential amount of radiance leaving point y along direction Θ_y as result of a single unit of emitted radiance from point **x** on a surface along direction Θ_x (Figure *2*). Formally:

$$\begin{aligned} F_r(x, \Theta_x, y, \Theta_y) &= \frac{\partial^2 L_{output}(y, \Theta_y)}{L_{input}(x, \Theta_x) \partial\omega_x \partial\mu_{x\perp}} \\ &= \frac{\partial^2 L_{output}(y, \Theta_y)}{L_{input}(x, \Theta_x) \partial\omega_x \, |\Theta_x \cdot N_x| \, \partial\mu_x} \end{aligned}$$

The *grdf* may be considered as a generalisation of the *brdf.* While the latter only specifies the local behaviour of light reflecting at a single surface, the former defines the global illumination effects of light reflecting through a complete scene. Note that the function is defined in terms of radiances *leaving* the surfaces, unlike the *brdf* which is commonly expressed in terms of both incoming and outgoing radiances. It is theoretically defined over the entire space of points and directions, but in the following paragraphs we will only use it at the surfaces of the scene themselves. Because we assume that there is

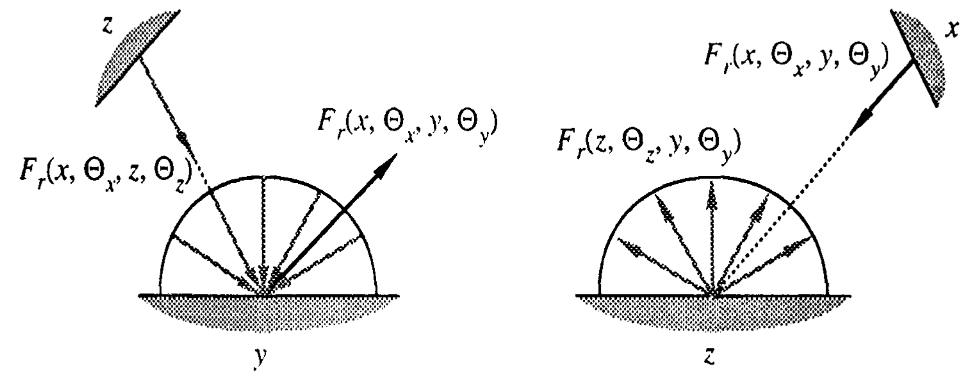

Figure 3: *Explanation of the symbols used in the two adjoint equations that define the global reflection distribution function.*

no participating medium the function is constant when moving x along direction Θ_x and y along direction Θ_y between two surfaces, by virtue of the invariance of the *input* and *output* radiances along these directions.

4.2. Adjoint Equations Defining the *Grdf*

The function is specified by a set of adjoint integral equations. Firstly, one can look at the behaviour of the function for a fixed (x, Θ_x). Similarly to the derivation of the rendering equation, the radiance at (y, Θ_y) is the result of two contributions. If both pairs of points and directions are equal the *input* radiance contributes airectly to the *output* radiance. In this case the *output* radiance is not differential but finite, so that the fraction is a Dirac impulse. The second contribution results from the reflection of incoming radiance at point y which arrives from points z, possibly through multiple reflections throughout the scene (Figure 3). This can be expressed formally as:

$$F_r(x, \Theta_x, y, \Theta_y) = \delta(x, \Theta_x, y, \Theta_y) \quad (1)$$
$$+ \int_{\Omega_y^{-1}} f_r(y, \Theta_z, \Theta_y) F_r(x, \Theta_x, z, \Theta_z) \, |\Theta_z \cdot N_y| \, d\omega_z$$

where $\delta(x, \Theta_x, y, \Theta_y)$ is the Dirac impulse which is 0 if $(x, \Theta_x) \neq (y, \Theta_y)$, but which integrates to 1 over the domain of (x, Θ_x) for a given (y, Θ_y) and vice versa:

$$\int_A \int_{\Omega_x} \delta(x, \Theta_x, y, \Theta_y) \, |\Theta_x \cdot N_x| \, d\omega_x d\mu_x = 1$$
$$\int_A \int_{\Omega_y} \delta(x, \Theta_x, y, \Theta_y) \, |\Theta_y \cdot N_y| \, d\omega_y d\mu_y = 1$$

Alternatively, one can look at the behaviour of the function for a fixed (y, Θ_y). This viewpoint resembles the approach of the potential equation. The question here is how radiance at (x, Θ_x) can contribute to the *output* radiance. Once again, if both pairs of points and directions are equal the *input* radiance contributes directly to the *output* radiance, which is expressed by the Dirac impulse. The *input* radiance can also contribute indirectly, through reflection in all directions around point z which is seen by point x in the direction Θ_x (Figure 3). This results in another recursive equation:

$$F_r(x, \Theta_x, y, \Theta_y) = \delta(x, \Theta_x, y, \Theta_y) \quad (2)$$
$$+ \int_{\Omega_z} f_r(z, \Theta_x, \Theta_z) F_r(z, \Theta_z, y, \Theta_y) \, |\Theta_z \cdot N_z| \, d\omega_z$$

It is extremely interesting to see that these equations are fully equivalent, by virtue of the bi-directional nature of light. The fraction of the radiance emitted from one point that is received by another point equals the fraction that would be received if the roles of the emitter and the receiver exchanged were, thus reversing the paths the light follows. This property can be expressed elegantly in terms of the *grdf:*

$$F_r(x, \Theta_x, y, \Theta_y) = F_r(\tilde{y}, \Theta_y^{-1}, \tilde{x}, \Theta_x^{-1})$$

where $\tilde{x}$ is the point seen by point x along direction Θ_x and $\tilde{y}$ the point seen by point y along direction Θ_y as shown in Figure 4. This expression is equivalent to the well-known reciprocal property of the *brdf;*

$$f_r(x, \Theta_y, \Theta_x) = f_r(x, \Theta_x^{-1}, \Theta_y^{-1})$$

Substitution of these properties in both sides of the second adjoint equation **(2)** yields:

$$F_r(\tilde{y}, \Theta_y^{-1}, \tilde{x}, \Theta_x^{-1}) = \delta(x, \Theta_x, y, \Theta_y)$$
$$+ \int_{\Omega_z} f_r(z, \Theta_z^{-1}, \Theta_x^{-1}) F_r(\tilde{y}, \Theta_y^{-1}, \tilde{z}, \Theta_z^{-1}) \, |\Theta_z \cdot N_z| \, d\omega_z$$

Renaming the variables appropriately results in the first adjoint equation (1). It implies that any of the two

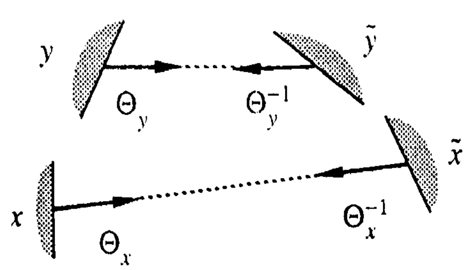

Figure 4: *Convention for naming the opposite points in the reciprocal property of the* grdf.

adjoint equations in combination with the reciprocal property is sufficient to define the *grdf* unambiguously.

4.3. Flux in Terms of the *Grdf*

Now that the *grdf* has been defined we can use it to determine the flux through the given set of points and directions S (defined by $W_e(x,\Theta_x)$) and with respect to the given self-emitted radiance function $L_e(x,\Theta_x)$. From the definition of the *grdf* and the expression of the flux in terms of radiances we can derive:

$$\Phi = \int_A \int_{\Omega_x} \int_A \int_{\Omega_y} L_e(x,\Theta_x) W_e(y,\Theta_y) \quad (3)$$
$$F_r(x,\Theta_x,y,\Theta_y)\,|\Theta_x \cdot N_x|\,|\Theta_y \cdot N_y|\,d\omega_y d\mu_y d\omega_x d\mu_x$$

Note that it is only at this time that the functions $L_e(x,\Theta_x)$ and $W_e(x,\Theta_x)$ come into play; the *grdf* itself is totally independent of them. If required, it remains possible to express the radiance function with respect to a given $L_e(x,\Theta_x)$ and the potential function with respect to a given $W_e(x,\Theta_x)$ in terms of the *grdf:*

$$L(y,\Theta_y) = \int_A \int_{\Omega_x} L_e(x,\Theta_x) F_r(x,\Theta_x,y,\Theta_y)\,|\Theta_x \cdot N_x|\,d\omega_x d\mu_x$$

$$W(x,\Theta_x) = \int_A \int_{\Omega_y} W_e(y,\Theta_y) F_r(x,\Theta_x,y,\Theta_y)\,|\Theta_y \cdot N_y|\,d\omega_y d\mu_y$$

The occurrence of a Dirac impulse in both of the adjoint equations may seem a bit awkward for practical use in algorithms. Luckily one will always want to integrate the *grdf* in order to obtain some flux. Therefore it is an advantage rather than a disadvantage, since integrating a Dirac impulse is extremely easy by virtue of its definition.

5. Bi-Directional Path Tracing

The light sources in a scene and the eye point of the viewer have always been identified as being very important for solving the global illumination problem and creating a realistic rendering. Some algorithms such as ray tracing are entirely built around the importance of the viewing point. Other algorithms such as the progressive radiosity method put a primary emphasis on the contributions of the light sources. Ideally one would want an algorithm which takes into account the importance of both the light sources and the viewing point. In another paper[20] we have presented bi-directional path tracing as a Monte Carlo algorithm which treats light sources and the viewing point on an equal basis. We will now demonstrate how it can be derived mathematically from the theory presented in the previous paragraphs.

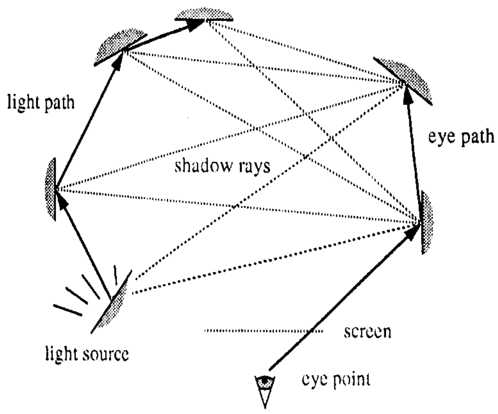

Figure 5: *A schematic representation of the bidirectional path tracing algorithm.*

5.1. Outline of the Algorithm

The algorithm differs from distribution ray tracing or path tracing in its computation of a primary estimator for the flux through each pixel. Each sample consists in tracing a pair of random walks through the scene: a light path starting from a light source and an eye path starting from the eye point. All hit points on the respective particle paths are then connected by means of shadow rays. For each shadow ray a contribution is computed and added to the flux of the pixel in question. As a result the lighting contributions are estimated by casting shadow rays from the eye paths, not only to the primary light sources themselves as in classical path tracing, but implicitly also to important secondary, tertiary,... light sources (Figure 5).

5.2. Mathematical Derivation

The algorithm can be derived from the mathematical model presented above, using classical Monte Carlo methods. The model consists of a set of two adjoint equations (1 and 2), which have been proven to be equivalent, and an expression for the flux (3). Our goal is to combine the ideas of shooting and gathering power.

5.2.1. Combining the Adjoint Equations

For convenience we will first abbreviate the adjoint integral equations (1) and (2) using integral operators T_e and T_l. The respective equations then look as follows:

$$F_r(x,\Theta_x,y,\Theta_y) = \delta(x,\Theta_x,y,\Theta_y) + T_e F_r(x,\Theta_x,y,\Theta_y)$$

$$F_r(x,\Theta_x,y,\Theta_y) = \delta(x,\Theta_x,y,\Theta_y) + T_l F_r(x,\Theta_x,y,\Theta_y)$$

where the integral operators T_e and T_l are defined as:

$$T_e F_r(x,\Theta_x,y,\Theta_y) = \int_{\Omega_y^{-1}} f_r(y,\Theta_z,\Theta_y) F_r(x,\Theta_x,z,\Theta_z)\,|\Theta_z \cdot N_y|\,d\omega_z$$

$$T_l F_r(x,\Theta_x,y,\Theta_y) = \int_{\Omega_z} f_r(z,\Theta_x,\Theta_z) F_r(z,\Theta_z,y,\Theta_y\,|\Theta_z \cdot N_z|\,d\omega_z)$$

For the purpose of integrating the ideas of shooting and gathering power these equations have to be combined. The most straightforward way to do so is to take a linear combination of them. This yields the following starting point:

$$\begin{aligned} F_r(x,\Theta_x,y,\Theta_y) &= \delta(x,\Theta_x,y,\Theta_y) \\ &+ w_e\, T_e F_r(x,\Theta_x,y,\Theta_y) \\ &+ w_l\, T_l F_r(x,\Theta_x,y,\Theta_y) \end{aligned} \tag{4}$$

where the sum of the weights $w_e + w_l = 1$. The solution of this integral equation can be found by recursively filling in the left-hand side term into the two terms with the integral operators on the right-hand side. Noting that one can vary the weights at each recursive step and that both integral operators are interchangeable the solution can eventually be written as a Neumann series of the form:

$$F_r(x,\Theta_x,y,\Theta_y) = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} w_{ij}\, T_l^i T_e^j \delta(x,\Theta_x,y,\Theta_y)$$

From a physical point of view $i+j$ expresses the number of reflections the light undergoes when travelling from (x,Θ_x) to (y,Θ_y). The weights w_{ij} are subject to the following constraints:

$$\sum_{i=0}^{\infty} w_{i,N-i} = 1 \text{ for } N = 0,1,\ldots$$

On the basis of equation (3) the flux can then be written as:

$$\Phi = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} w_{ij} C_{ij} \tag{5}$$

where:

$$\begin{aligned} C_{ij} = & \int_A \int_{\Omega_x} \int_A \int_{\Omega_y} L_e(x,\Theta_x) W_e(y,\Theta_y) \\ & w_{ij} T_l^i T_e^j \delta(x,\Theta_x,y,\Theta_y) \\ & |\Theta_x \cdot N_x|\,|\Theta_y \cdot N_y|\, d\omega_y d\mu_y d\omega_x d\mu_x \end{aligned}$$

This expression will be used later to effectively evaluate an estimate for the flux.

5.2.2. Estimating the Flux

We will now show how to find an estimate of the flux for each pixel in turn. The starting point is the original equation (3) expressing the flux. The selected pixel determines the function $W_e(x,\Theta_x)$ The function is 1 for points on the surfaces and respective directions which contribute directly to its flux, and 0 otherwise. The given lighting conditions determine the function $L_e(x,\Theta_x)$. The integrand contains Dirac impulses which have to be evaluated separately; we will take this into account later.

In order to solve this integral using Monte Carlo integration, the variables of the integrands x, Θ_x, y and Θ_y have to be sampled over their respective domains. The choice of a probability density function *(pdf)* is free, but the principle of importance sampling states that the variance of the stochastic process will be lower when the pdffollows the function to be integrated more closely. On the basis of the expression for the flux (3) we therefore select samples x_0 and Θ_{x_0} according to the following *pdf* (Figure 6):

$$pdf(x,\Theta_x) = \frac{L_e(x,\Theta_x)\,|\Theta_x \cdot N_x|}{L}$$

where L is the normalisation factor:

$$L = \int_A \int_{\Omega_x} L_e(x,\Theta_x)\,|\Theta_x \cdot N_x|\,d\omega_x d\mu_x$$

From a physical point of view this pdfensures that more light particles are shot from bright emitters and in bright directions, rather than distributing them uniformly and weighting their contributions to the flux afterwards. In the same vein we select samples y_0 and Θ_{y_0} according to the following *pdf* (Figure 7):

$$pdf(y,\Theta_y) = \frac{W_e(y,\Theta_y)\,|\Theta_y \cdot N_y|}{G}$$

where G is the normalisation factor:

$$G = \int_A \int_{\Omega_y} W_e(y,\Theta_y)\,|\Theta_y \cdot N_y|\,d\omega_y d\mu_y$$

The appropriate weights for the samples can be derived on the basis of these *pdfs* using the template for importance sampling:

$$I = \int f(x)dx = \int pdf(x) \times \text{ weight}(x)dx.$$

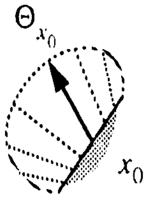

Figure 6: *Sampling x_0 and Θ_{x_0} on the basis of the self-emitted radiance of light sources.*

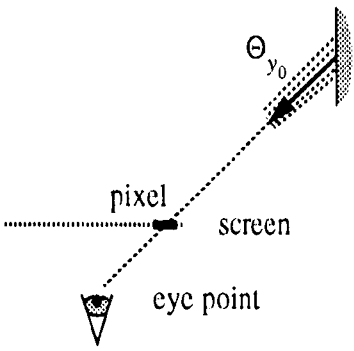

Figure 7: *Sampling y_0 and Θ_{y_0} with respect to the pixel under consideration.*

Without taking into account the fact that the *grdf* contains Dirac impulses the resulting primary estimator for the flux looks like:

$$\langle \Phi \rangle = L \times G \times F_r(x_0, \Theta_{x_0}, y_0, \Theta_{y_0})$$

5.2.3. Estimating the *Grdf*

The estimator of the flux contains the *grdf* $F_r(x, \Theta_x, y, \Theta_y)$ evaluated for the specific points x_0 and y_0 and directions Θ_{x_0} and Θ_{y_0}. Neither the function nor this particular value are known, so the value has to be estimated in its turn. Integral equation (4) for $F_r(x, \Theta_x, y, \Theta_y)$ which has been derived earlier forms the basis for another Monte Carlo process. According to standard Monte Carlo theory the particular value can be estimated by performing a random walk. Because the equation contains two integrals there are actually many recursive random walks, intertwined with each other. By re-using the samples this can be reduced to

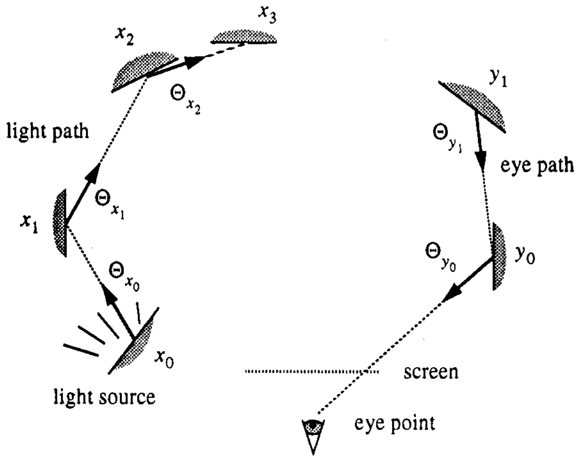

Figure 8: *Naming conventions for the points and directions $x_0, \Theta_{x_0}, \ldots, \Theta_{x_{N_l-1}}$ and $y_0, \Theta_{y_0}, \ldots, \Theta_{y_{N_e-1}}$ to specify the light path and the eye path respectively. In this example N_l equals 3 and N_e equals 2.*

two stochastically determined paths. These are written as:

- $x_0, x_1, x_2, \ldots, x_{N_l}$ for the light path, where x_{i+1} is the point seen by point x_i along direction Θ_{x_i}, and
- $y_0, y_1, y_2, \ldots, y_{N_e-1}$ for the eye path, where y_{j+1} is the point that sees point y_j along direction $\Theta_{y_{j+1}}$.

Using this notation the random samples which have to be selected to compute the two integrals of $F_r(x_i, \Theta_{x_i}, y_j, \Theta_{y_j})$ are $\Theta_{x_{i+1}}$ and $\Theta_{y_{j+1}}$ respectively. These variables determine x_{i+2} and y_{j+1} unambiguously (Figure 8). The so-called *subcritical pdf* for $\Theta_{x_{i+1}}$ is again chosen according to the principle of importance sampling with respect to equation (1) (Figure 9):

$$pdf(\Theta) = f_r(x_{i+1}, \Theta_{x_i}, \Theta) \left| \Theta \cdot N_{x_{i+1}} \right|$$

It is subcritical because it does not integrate to 1 over all possible angles Θ, at least for physically valid *brdfs*. The actual value of the integration (the overall reflectivity for this specific incident angle) gives the chance that the random walk is continued, which ensures that the random walk terminates.

The *subcritical pdf* for $\Theta_{y_{j+1}}$ is chosen in the same way, with respect to equation *(2)* (Figure 10):

$$pdf(\Theta) = f_r(y_j, \Theta, \Theta_{y_j}) \left| \Theta \cdot N_{y_j} \right|$$

Note that these *spdfs* can be made identical simply by renaming the variables and using the reciprocal property of the *brdf*. This property implies that both random walks can be performed by a single algorithm.

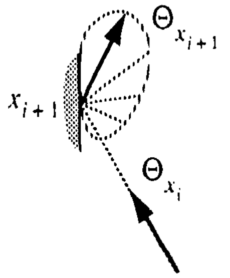

Figure 9: *Sampling* $\Theta_{x_{i+1}}$ *according to the* brdf *of the surface at point* x_{i+1} *for the incoming direction* Θ_{x_i}.

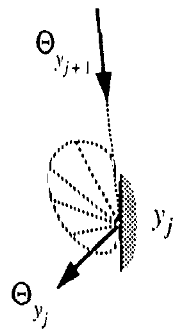

Figure 10: *Sampling* $\Theta_{y_{j+1}}$ *according to the* brdf *of the surface at point* y_j *for the incoming direction* Θ_{x_j}.

On the basis of the selected *spdfs* and without taking into account the special character of the Dirac impulses the primary estimator for $F_r(x_i, \Theta_{x_i}, y_j, \Theta_{y_j})$ would look like:

$$\begin{aligned}\langle F_{r(i,j)}\rangle &= \delta(x_i, \Theta_{x_i}, y_j, \Theta_{y_j}) \\ &+ w_l\, \langle F_{r(i+1,j)}\rangle \text{ if } \Theta_{x_{i+1}} \text{ is found} \\ &+ w_e\, \langle F_{r(i,j+1)}\rangle \text{ if } \Theta_{y_{j+1}} \text{ is found}\end{aligned}$$

5.3. Evaluating the Estimate of the Flux

Now that we have shown how the stochastic variables for the integrals and for the random walks are selected we will actually evaluate the final result. Because of the special nature of the Dirac impulses it is easier to evaluate expression (5) for the flux. The infinite *Neumann* series of the flux becomes a finite sum for the estimate of the flux by virtue of the finite random walks:

$$\langle \Phi \rangle = \sum_{i=0}^{N_l} \sum_{j=0}^{N_e} w_{ij} \langle C_{ij} \rangle$$

The Dirac impulses can be removed from the expressions using their definitions. Eventually three cases have to be distinguished when evaluating the estimates $\langle C_{ij}\rangle$:

- $i = 0, j = 0$. From a physical point of view this term is an estimate for the flux received from a light source that is directly seen through the pixel under consideration. Taking into account the previously selected *pdfs* for y_0 and Θ_{y_0} the estimate becomes:

$$\langle C_{00}\rangle = G \times L_e(y_0, \Theta_{y_0})$$

- $i = 0, j > 0$. This term is an estimate for the flux that reaches the eye from the light source through j reflections on the eye path, as in classical path tracing. Taking into account the previously selected *pdfs* for $x_0, y_0, \Theta_{y_0}, \ldots, \Theta_{y_{j-1}}$ the corresponding estimate becomes:

$$\begin{aligned}\langle C_{0j}\rangle &= L' \times G \\ &\times L_e(x_0, \Theta_{x_0 \to y_{j-1}}) \\ &\times f_r(y_{j-1}, \Theta_{x_0 \to y_{j-1}}, \Theta_{y_{j-1}}) \times v(x_0, y_{j-1}) \\ &\times \frac{\left|\Theta_{x_0 \to y_{j-1}} \cdot N_{x_0}\right| \left|\Theta_{x_0 \to y_{j-1}} \cdot N_{y_{j-1}}\right|}{\|x_0 - y_{j-1}\|^2}\end{aligned}$$

where:

$$L' = \frac{L}{\int_{\Omega_x} L_e(x_0, \Theta_x)\, |\Theta_x \cdot N_{x_0}|\, d\omega_x}$$

and where the visibility function $v(x, y)$ is 1 if point x 'sees' point y (along direction $\Theta_{x \to y}$). The visibility function and the fraction on the last line result from a transformation from spherical coordinates to coordinates on the surfaces in the scene. In practice the visibility function is evaluated by casting a shadow ray between the two points and checking whether it is intersected by an object in the scene.

- $i > 0, j > 0$. This term is an estimate for the flux that reaches the eye from the light source, through i reflections on the light path and j reflections on the eye path. Again taking into account the previously selected *pdfs* for $x_0, \Theta_{x_0}, \ldots, \Theta_{x_{i-1}}, y_0, \Theta_{y_0}, \ldots, \Theta_{y_{j-1}}$ the corresponding estimate becomes:

$$\begin{aligned}\langle C_{ij}\rangle &= L \times G \\ &\times f_r(x_i, \Theta_{x_{i-1}}, \Theta_{x_i \to y_{j-1}}) \\ &\times f_r(y_{j-1}, \Theta_{x_i \to y_{j-1}}, \Theta_{y_{j-1}}) \times v(x_i, y_{j-1}) \\ &\times \frac{\left|\Theta_{x_i \to y_{j-1}} \cdot N_{x_i}\right| \left|\Theta_{x_i \to y_{j-1}} \cdot N_{y_{j-1}}\right|}{\|x_i - y_{j-1}\|^2}\end{aligned}$$

6. Results

We have implemented the bi-directional path tracing algorithm as described above. The program has been written in the programming language C on an IBM RS/6000 Model 320. It is based on the library routines of the public domain ray tracing program 'Rayshade'.

6.1. Selection of the Weights

The mathematical derivation in the previous paragraphs shows that one is free to choose the various weights for the contributions as long as they comply with the given constraints. The alternatives are obviously endless. We present some notable examples:

- $w_{ij} = 1$ for $i = 0$, and 0 otherwise.
  One can easily verify that this set of weights yields the classical path tracing algorithm. The generic bi-directional path tracing algorithm therefore also covers this technique. This instantiation does not fully exploit the potential of our method however, since much information is left unused.
- $w_{ij} = W^{j-1}$ for $i = 0$, and $W^{j-1}(1 - W)$ otherwise.
  This selection of the weights results in an extended path tracing algorithm. It can be explained as follows. The estimate of the incoming radiance at each point on the eye path is found in two distinct ways: by sampling the incoming angles on the basis of the *brdf* and by sampling the points on the eye path. The former contribution is assigned a weight W, the latter weight $1 - W$. The actual value of the weight may be set to 0.5 for instance; it is always rather arbitrary.
- $w_{ij} = \prod_{k=0}^{j-2} W_k$ for $i = 0$, 0 for $j = 0$ and $(\prod_{k=0}^{j-2} W_k)(1 - W_{j-1})$ otherwise.
  This selection of weights is similar to the previous one; only now the last degree of freedom has been put to good use. The idea is that for specular surfaces one would rather rely on the estimate found by following the eye path. For diffuse surfaces the estimate found through the contributions of the light path is more likely to be accurate. Therefore the weight W_j is chosen proportional to a measure of the degree of specularity of the surface at point y_j on the eye path. For highly specular surfaces it approaches 1, for diffuse surfaces it goes to 0. Tests on practical scenes have shown that this technique greatly improves the quality of the images, especially when rendering scenes containing mirrors.

6.2. Comparison with Classical Path Tracing

We have performed some tests in order to compare our bi-directional path tracing algorithm with classical path tracing. The main amount of work in both algorithms consists in performing ray intersection tests. So as to obtain a fair comparison approximately the same numbers of rays are used by the respective algorithms in each test. Table 1 gives an overview of the results. Both implementations use optimised sampling strategies such as importance sampling, stratified sampling and Russian roulette. Neither of the implementations performs adaptive sampling of the pixels.

The scene consists of coloured diffuse walls, a slightly specular floor and a mixture of opaque and transparent objects. In test A the scene is directly illuminated from above. Both algorithms accurately render typical global illumination effects such as diffuse and glossy reflections, soft shadows and colour bleeding. There is little difference visible between the resulting images. In test B the scene is mostly indirectly illuminated by two light sources against the walls. For this typical interior lighting configuration the bi-directional algorithm produces visibly less noise for the same amount of work. In order to further reduce the noise more samples are required per pixel though.

Test	Algorithm	Number of samples per pixel	Total number of rays	Time (sec)
A	Path Tr.	60	45×10^6	15 897
	Bi-D. Path Tr.	20	30×10^6	11 036
B	Path Tr.	60	35×10^6	18 251
	Bi-D. Path Tr.	20	31×10^6	14 929

Table 1: *Overview of the test results.*

7. Conclusion

In this paper we have introduced the notion of global reflection distribution function and reformulated the global illumination problem in terms of this function. This results in two adjoint integral equations which define the grdf for a given scene. We have proven that these equations are equivalent by virtue of the bi-directionality of light. A simple integral containing the grdf expresses the flux through a given set of points and directions, with respect to a given set of light sources. We believe that this fresh point of view, along with its well-founded theoretical basis, may lead to new and better insights in the theory of global illumination

The practical use of the formulation has been demonstrated by deriving a new Monte Carlo algorithm from it using standard mathematical techniques. This bi-directional path tracing treats the viewpoint and the light sources on an equal basis. The technique is very general. Although convergence still is rather slow, practical results show that it outperforms standard path tracing for scenes where indirect lighting effects predominate. Classical optimised sampling techniques such as importance sampling, stratified sampling and Russian roulette further improve the convergence of the algorithm.

Acknowledgements

The first author would like to acknowledge the financial support by a grant from the Belgian "Instituut tot Aanmoediging van Wetenschappelijk Onderzoek in Nijverheid en Landbouw" (I.W.O.N.L. #910184). Implementation was done on equipment kindly made available by IBM Belgium in the framework of an Academic Joint Study Programme. Many thanks also go to the B.O.L.T.S. group and especially to Philip Dutré for the fruitful discussions.

References

1. C. Goral, K. Torrance, D. Greenberg, and B. Battaile, "Modelling the Interaction of Light Between Diffuse Surfaces", *Computer Graphics,* 18(3), pp. 213–222 (1984).
2. B. Smits, J. Arvo, and D. Salesin, "An Importance-driven Radiosity Algorithm", *Computer Graphics,* 26(2), pp. 273–282 (1992).
3. J. Kajiya, "The Rendering Equation", *Computer Graphics,* 20(4), pp. 143–150 (1986).
4. R. Cook, T. Porter, and L. Carpenter, "Distributed Ray Tracing", *Computer Graphics,* 18(3), pp. 137–145 (1984).
5. M. Lee, R. Redner, and S. Uselton, "Statistically Optimized Sampling for Distributed Ray Tracing", *Computer Graphics,* 19(3), pp. 61–67 (1985).
6. G. Ward, F. Rubinstein, and R. Clear, "A Ray Tracing Solution for Diffuse Interreflection", *Computer Graphics,* 22(4), pp. 85–92 (1988).
7. P. Shirley, *Physically Based Lighting Calculations for Computer Graphics*. PhD thesis, University of Illinois, (Nov. 1990).
8. B. Lange, "The Simulation of Radiant Light Transfer with Stochastic Ray-tracing", in *Proceedings of the Second Eurographics Workshop on Rendering,* (Barcelona, Spain), (May 1991).
9. P. Shirley and C. Wang, "Distribution Ray Tracing: Theory and Practice", in *Proceedings of the Third Eurographics Workshop on Rendering,* (Bristol, UK), pp. 33-43, (May 1992).
10. S. Pattanaik, "The Mathematical Framework of Adjoint Equations for Illumination Computation", in *Proceedings of ICCG,* (Bombay, India), pp. 265–288, (Feb. 1993).
11. S. Pattanaik, *Computational Methods for Global Illumination and Visualisation of Complex 3D Environments*. PhD thesis, Birla Institute of Technology & Science, Pilani, India, (Feb. 1993).
12. S. Pattanaik and S. Mudur, "The potential equation and importance in illumination computations", *Computer Graphics Forum,* 12(2), pp. 131–136 (1993).
13. S. Pattanaik and S. Mudur, "Computation of Global Illumination by Monte Carlo Simulation of the Particle Model of Light", in *Proceedings of the Third Eurographics Workshop on Rendering,* (Bristol, UK), pp. 71–83, (May 1992).
14. J. Wallace, M. Cohen, and D. Greenberg, "A Two-pass Solution to the Rendering Equation: a Synthesis of Ray Tracing and Radiosity Method", *Computer Graphics,* 21(4), pp. 311–320 (1987).
15. F. Sillion and C. Puech, "A General Two-pass Method Integrating Specular and Diffuse Reflection", *Computer Graphics,* 23(4), pp. 335–344 (1989).
16. S. Chen, H. Rushmeier, G. Miller, and D. Turner, "A Progressive Multi-pass Method for Global Illumination", *Computer Graphics,* 25(4), pp. 164–174 (1991).
17. E Sillion, J. Arvo, S. Westin, and D. Greenberg, "A Global Illumination Solution for General Reflectance Distributions", *Computer Graphics,* 25(4), pp. 187–196 (1991).
18. L. Aupperle and P. Hanrahan, "Importance and discrete three point transport", in *Proceedings of the Fourth Eurographics Workshop on Rendering,* (Paris, France), pp. 85–94, (June 1993).
19. P. Christensen, D. Salesin, and T. DeRose, "A continuous adjoint formulation for radiance transport", in *Proceedings of the Fourth Eurographics Workshop on Rendering,* (Paris, France), pp. 95–104, (June 1993).
20. E. Lafortune and Y. Willems, "Bi-directional Path Tracing", in *Proceedings of CompuGraphics,* (Alvor, Portugal), pp. 145–153, (Dec. 1993).

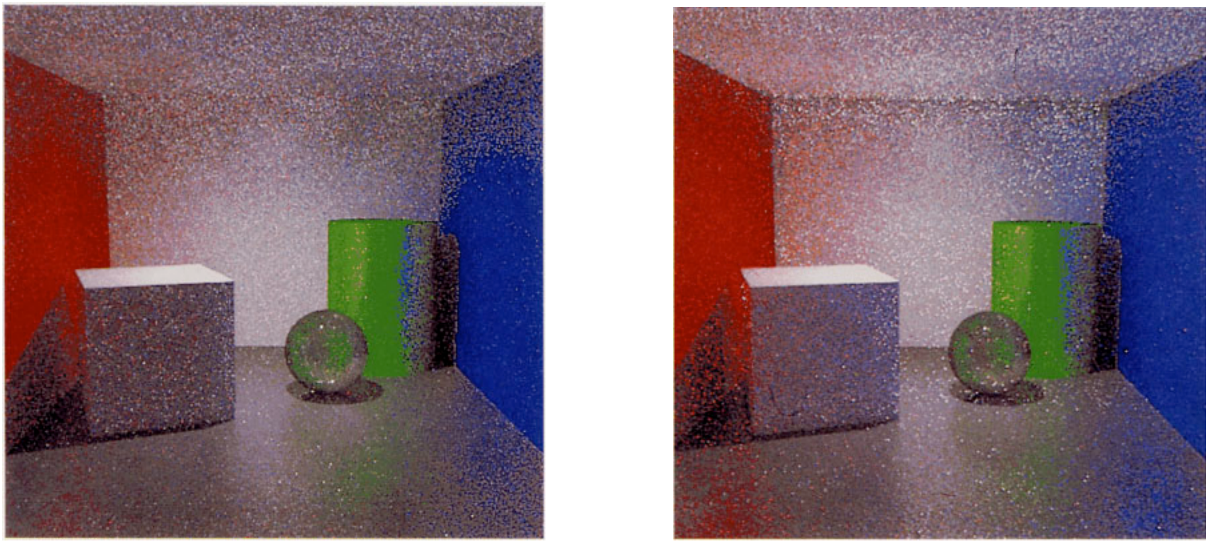

Figure A: *A directly illuminated scene rendered using path tracing and bidirectional path tracing respectively. Little difference between the two images is visible.*

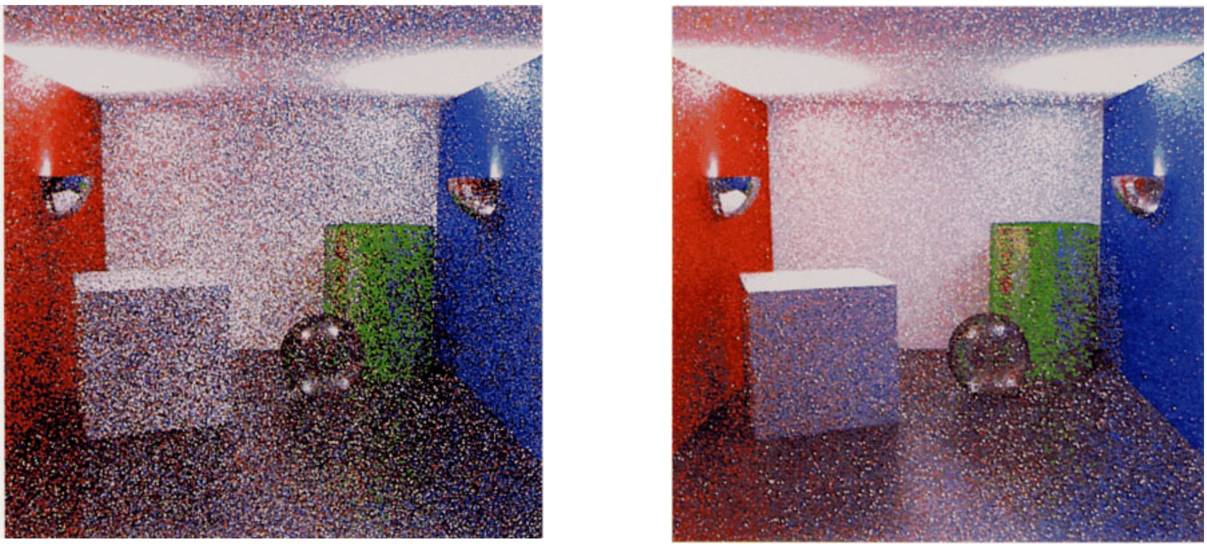

Figure B: *An indirectly illuminated scene rendered using path tracing and bidirectional path tracing respectively. The latter method results in visibly less noise for the same amount of work.*