Unbiased Global Illumination with Participating Media

Matthias Raab[1], Daniel Seibert[2], and Alexander Keller[3]

[1] Institut für Medieninformatik, Universität Ulm matthias.raab@uni-ulm.de
[2] mental images GmbH daniel@mental.com
[3] Institut für Medieninformatik, Universität Ulm alexander.keller@uni-ulm.de

Summary. We present techniques that enhance global illumination algorithms by incorporating the effects of participating media. Instead of ray marching we use a sophisticated Monte Carlo method for the realization of propagation events and transmittance estimations. The presented techniques lead to unbiased estimators of the light transport equation with participating media.

1 Introduction

A complete computation of all illumination effects is critical for the synthesis of photorealistic images. This means that the *global illumination* problem has to be solved. Global illumination is an important problem in graphics and of high interest for applications like architecture, industrial design, and even production. Accordingly, there are a great number of approaches that attempt to solve the task of simulating light transport through virtual three-dimensional scenes in an unbiased and physically correct way. Nevertheless, most algorithms lack the ability to correctly estimate the effects caused by interactions with media like smoke, fog, or dust.

The most sophisticated unbiased approaches are Bidirectional Path Tracing [LW93, VG94] and the Metropolis Light Transport algorithm [VG97, KSKAC02]. All of these algorithms are robust and can capture a wide variety of illumination effects, albeit with varying efficiency. Additionally, extensions to scenes with participating media were presented in [LW96, PKK00]. Note, however, that these techniques are not unbiased as they rely on ray marching techniques [PH89] to sample distances and to approximate the transmittance of participating media.

2 Light Transport with Participating Media

In the theory of radiative transfer there is generally a distinction between solid objects, i.e. those that do not allow light to pass through them, and diaphanous media like gases and liquids. Scenes are therefore often modeled as a volume $\mathcal{V}$ and its boundary, i.e. the surface of solid objects $\partial\mathcal{V}$. Note that we assume $\mathcal{V}$ to be an open set, so that $\mathcal{V} \cap \partial\mathcal{V} = \emptyset$. On the surface $\partial\mathcal{V}$ the *local scattering equation* governs light transport

$$L(x,\omega) = L_{e,\partial\mathcal{V}}(x,\omega) + \int_{\mathcal{S}^2} f_s(\omega, x, \omega') L(x,\omega') |\cos\theta_x| d\sigma(\omega'). \tag{1}$$

Here, $\mathcal{S}^2$ is the set of all directions, f_s is the *bidirectional scattering distribution function* (BSDF), which describes the scattering behavior at x, and $\cos\theta_x$ is the cosine of the angle between direction ω' and the surface normal in x. This formulation is sufficient for scenes in a vacuum, where all interaction events occur on the surface.

In order to account for effects caused by participating media inside the volume, we have to consider the *equation of transfer*

$$\begin{aligned} \frac{\partial}{\partial\omega} L(x,\omega) =& L_{e,\mathcal{V}}(x,\omega) - \sigma_t(x) L(x,\omega) \\ &+ \sigma_s(x) \int_{\mathcal{S}^2} f_p(\omega, x, \omega') L(x,\omega') d\sigma(\omega'), \end{aligned} \tag{2}$$

which describes the radiance change at position x in direction ω due to volume emission and interaction events. The medium's scattering and absorption characteristics are given by the phase function f_p, the scattering coefficient σ_s, and the absorption coefficient σ_a. The latter two form the extinction coefficient $\sigma_t := \sigma_s + \sigma_a$. Usually, equation (2) is integrated along straight light rays to the next surface interaction point $x_{\mathcal{S}} = h(x, -\omega)$, which is found by the *ray casting function* h. This approach yields a Fredholm integral equation of the second kind, which can be handled by Monte Carlo techniques.

2.1 Path Integral Formulation

A very general formulation of light transport is given in [Vea97] and has been extended to handle participating media in [PKK00] and [Kol04]. The local description by equations (1) and (2) is recursively expanded to obtain an integral over an abstract space of complete transport paths.

The path space $\mathcal{P}$ can be modeled as the union of spaces containing paths of a specific finite length, i.e.

$$\mathcal{P} := \bigcup_{k \in \mathbb{N}} \mathcal{P}_k \text{ where } \mathcal{P}_k := \left\{ \bar{x} = x_0 \ldots x_k : x_i \in \mathbb{R}^3 \right\}.$$

For each of these spaces we use a product measure μ_k, defined for a set $\mathcal{M}_k \subseteq \mathcal{P}_k$ by

$$\mu_k(\mathcal{M}_k) := \int_{\mathcal{M}_k} d\lambda(x_0) d\lambda(x_1) \cdots d\lambda(x_k),$$

of the corresponding Lebesgue measure on the volume and on the surface, i.e.

$$d\lambda(x) := \begin{cases} dA(x) & \text{if } x \in \partial\mathcal{V} \\ dV(x) & \text{if } x \in \mathcal{V}. \end{cases}$$

The measure μ of a set $\mathcal{M} \subseteq \mathcal{P}$ then is the natural expansion of those disjoint spaces' measures

$$\mu(\mathcal{M}) := \sum_{k \in \mathbb{N}} \mu_k \left(\mathcal{M} \cap \mathcal{P}_k \right).$$

In this context, the sensor response I_j of pixel j can be expressed as an integral over $\mathcal{P}$,

$$I_j = \int_{\mathcal{P}} f_j(\bar{x}) d\mu(\bar{x}), \tag{3}$$

where f_j is called the *measurement contribution function*. In order to find this function, we describe light transport in a slightly different manner. Let $L(y \to z)$ denote the radiance scattered and emitted from point y in direction $\overrightarrow{yz} := \frac{z-y}{\|z-y\|}$. Inside the volume this quantity is given by

$$L(y \to z) = L_{e,\mathcal{V}}(y, \overrightarrow{yz}) + \int_{\mathcal{S}^2} \sigma_s(y) f_p(\overrightarrow{yz}, y, \omega) L(y, \omega) d\sigma(\omega). \tag{4}$$

On the surface we obtain $L(y \to z)$ directly by equation (1), i.e. $L(y \to z) = L(y, \overrightarrow{yz}) \; \forall y \in \partial\mathcal{V}$.

Using these notions in the integration of (2) with boundary condition (1) and changing the integration domain to $\mathbb{R}^3$ yields the *three point form*:

$$\begin{aligned} L(y \to z) &= L_e(y \to z) \\ &\quad + \int_{\mathbb{R}^3} L(x \to y) f(x \to y \to z) G(x \leftrightarrow y) V(x \leftrightarrow y) d\lambda(x), \end{aligned} \tag{5}$$

where the following abbreviations are used (see [Kol04] for a full derivation):

- Three point scattering function

$$f(x \to y \to z) := \begin{cases} f_r(\overrightarrow{yz}, y, \overrightarrow{xy}) & \text{if } y \in \partial\mathcal{V} \\ \sigma_s(y) f_p(\overrightarrow{yz}, y, \overrightarrow{xy}) & \text{if } y \in \mathcal{V} \end{cases} \tag{6}$$

- Source radiance distribution

$$L_e(x \to y) := \begin{cases} L_{e,\partial\mathcal{V}}(x, \overrightarrow{xy}) & \text{if } y \in \partial\mathcal{V} \\ L_{e,\mathcal{V}}(x, \overrightarrow{xy}) & \text{if } y \in \mathcal{V} \end{cases} \tag{7}$$

- Geometric term

$$G(x \leftrightarrow y) := \begin{cases} \frac{|\cos\theta_x||\cos\theta_y|}{\|y-x\|^2} & \text{if } x, y \in \partial\mathcal{V} \\ \frac{|\cos\theta_x|}{\|y-x\|^2} & \text{if } x \in \partial\mathcal{V}, y \in \mathcal{V} \\ \frac{|\cos\theta_y|}{\|y-x\|^2} & \text{if } y \in \partial\mathcal{V}, x \in \mathcal{V} \\ \frac{1}{\|y-x\|^2} & \text{if } x, y \in \mathcal{V} \end{cases} \tag{8}$$

  with θ_x and θ_y the angles between $\overrightarrow{xy}$ and the surface normals at the respective points
- Attenuated visibility function

$$\begin{aligned} V(x \leftrightarrow y) :&= V'(x \leftrightarrow y)\tau(x \leftrightarrow y) \\ &= V'(x \leftrightarrow y)e^{-\int_0^{\|y-x\|}\sigma_t(x+t\overrightarrow{xy})dt} \end{aligned} \tag{9}$$

  with the standard binary visibility function

$$V'(x \leftrightarrow y) = \begin{cases} 1 & \text{if } \|y-x\| \leq \|h(x, \overrightarrow{xy}) - x\| \\ 0 & \text{otherwise} \end{cases}$$

Finally, we need a function $\psi_j(x_{k-1} \to x_k)$ to describe the sensor response of a pixel j to radiance arriving from x_{k-1} at the point x_k on the image plane. Along with the recursive expansion of the three point form (5) this yields the measurement contribution function for a path $\bar{x} = x_0 \ldots x_k$ beginning on a light source and ending on the image plane:

$$\begin{aligned} f_j(\bar{x}) := &L_e(x_0 \to x_1)G(x_0 \leftrightarrow x_1)V(x_0 \leftrightarrow x_1) \\ &\cdot \prod_{l=1}^{k-1}\big(f(x_{l-1} \to x_l \to x_{l+1})G(x_l \leftrightarrow x_{l+1})V(x_l \leftrightarrow x_{l+1})\big) \\ &\cdot \psi_j(x_{k-1} \to x_k) \end{aligned} \tag{10}$$

3 Unbiased Techniques for Transport Path Sampling

In a vacuum the next interaction point is fixed as the closest surface point along the ray. With participating media, however, the position of this point is no longer given deterministically but described by a stochastic process. From the structure of equation (10) it is obvious that we have two separate operators that describe the transport: one governing the distance to the next interaction point (the τ term) and the other governing the scattering behavior (the phase function or the BSDF). We treat these factors independently as in [PKK00] by first sampling a distance and then sampling a direction, as described in the next section. Since the processes are clearly independent, the resulting density is simply the product of the two densities involved.

Homogeneous Media

In the case of a homogeneous medium with $\sigma_t(x) \equiv \sigma_t$, the transmittance is proportional to the exponential distribution's density $p(t) = \sigma_t e^{-\sigma_t t}$. Applying the inversion method we realize the desired distance as

$$t = \frac{-\ln(1-\xi)}{\sigma_t} \tag{11}$$

for a uniformly distributed random number $\xi \in [0, 1)$.

Heterogeneous Media

In the general case, the density proportional to $\tau = e^{-K(t)}$ with $K(t) := \int_0^t \sigma_t(x_0 + t'\omega)dt'$ is given by $p(t) = \left(\frac{d}{dt}K(t)\right) e^{-K(t)} = \sigma_t(x_0 + t\omega)e^{-K(t)}$. Things are more complicated here since the inversion method only yields the implicit equation

$$K(t) = \int_0^t \sigma_t(x_0 + t'\omega)dt' = -\ln(1-\xi) \tag{12}$$

for a uniformly distributed random number $\xi \in [0, 1)$. As the inversion method cannot be applied directly and there is no straightforward Monte Carlo estimator available (as we cannot evaluate τ), this distance is usually sampled using the classic ray marching algorithm [PH89]. Note that this method, which is frequently used in computer graphics, is biased.

```
float sampleDistance(Point x_0, Direction ω)
{
    //sample with the maximum extinction σ_t
    float t = -log(rand()) / σ_t;

    while (σ_t(x_0+tω)/σ_t < rand())
        t -= log(rand()) / σ_t;

    return t;
}
```

Algorithm 1: Unbiased distance sampling for arbitrary media.

A more sophisticated and unbiased approach to sampling this distance can be found in [Col68]. Let σ_t be a constant with $\sigma_t \geq \sigma_t(x)$ for all $x \in \mathcal{V}$ and let $t_1, t_2, \ldots$ be independent random distances sampled according to equation (11) with parameter σ_t. Let, furthermore, $\xi_1, \xi_2, \ldots \in [0, 1]$ be independent uniformly distributed random numbers. Then, the first distance

$T_{i_0} = \sum_{i=1}^{i_0} t_i$ satisfying $\xi_{i_0} \leq \frac{\sigma_t(x_0+T_{i_0}\omega)}{\sigma_t}$ is distributed as desired. Algorithm 1 implements this procedure and thus provides an unbiased distance sampling routine for arbitrary media.

In fact, the condition $\sigma_t \geq \sigma_t(x)$ only has to be satisfied for points along the ray. Choosing a larger σ_t simply yields more iterations in Algorithm 1. However, determining the maximum $\sigma_t = \sup_{x\in\mathcal{V}} \sigma_t(x)$ in the whole volume data set during a preprocessing step is usually easy while determining the maximum along a single ray may be quite complicated.

3.1 Line Integral along a Ray

An explicit estimation of the transmittance τ is quite important in many algorithms. Furthermore, splitting along the primary ray is often beneficial when rendering scenes that contain participating media. We therefore generalize the one-dimensional integration along a ray in the context of Algorithm 1 in order to obtain a generic and unbiased solution.

---

1. Determine the surface intersection point $x_{\partial\mathcal{V}} = h(x, -\omega)$, the distance $t_{\partial\mathcal{V}} = \|x - x_{\partial\mathcal{V}}\|$, and the maximum volume extinction σ_t
2. Compute the probabilities $p_{\mathcal{V}} = 1 - e^{-\sigma_t t_{\partial\mathcal{V}}}$ and $p_{\partial\mathcal{V}} = 1 - p_{\mathcal{V}}$
3. Estimate the surface contribution $c_{\partial\mathcal{V}} = L(x_{\partial\mathcal{V}}, \omega)$ and set $C = p_{\partial\mathcal{V}} \cdot c_{\partial\mathcal{V}}$
4. Generate n randomly shifted equidistant sample points $\Delta \subset [0, p_{\mathcal{V}})$
$$\Delta := \left\{ \frac{(k+\xi)p_{\mathcal{V}}}{n} : k = 0, \ldots, n-1 \right\}$$
for one uniformly distributed random number $\xi \in [0, 1)$
5. Use Δ as initial random numbers for n random walks according to Algorithm 1 resulting in distances $t_1, \ldots, t_n$
6. Add contributions
$$C = C + \begin{cases} \frac{1}{n} \cdot p_{\mathcal{V}} \cdot \frac{c(x-t_k\omega,\omega)}{\sigma_t(x-t_k\omega)}, & \text{if } t_k < t_{\partial\mathcal{V}} \\ \frac{1}{n} \cdot p_{\mathcal{V}} \cdot c_{\partial\mathcal{V}}, & \text{else} \end{cases}$$
for $k = 1, \ldots, n$

Algorithm 2: Unbiased line integration along a ray.

Assume that we have to estimate a transmittance-weighted integral C along a light ray $(x_{\partial\mathcal{V}}, \omega)$ starting at the surface intersection point $x_{\partial\mathcal{V}}$ as present in the transport equations,

$$C = c_{\partial\mathcal{V}}\,(x_{\partial\mathcal{V}})\,\tau(x \leftrightarrow x_{\partial\mathcal{V}}) + \int_0^{\|x_{\partial\mathcal{V}}-x\|} c_{\mathcal{V}}(x - t\omega)\tau(x \leftrightarrow x - t\omega)dt, \quad (13)$$

where $c_{\mathcal{V}}$ and $c_{\partial\mathcal{V}}$ are volume and surface contributions, respectively. For the simple case $c_{\partial\mathcal{V}} \equiv 1$ and $c_{\mathcal{V}} \equiv 0$ we obtain an estimate of the transmittance itself. We can estimate C by applying Algorithm 1 repeatedly and averaging the corresponding contributions. Instead of using random numbers for the starting points of Algorithm 1 the convergence can be improved by transforming equidistant samples that are shifted by one random offset (see Figure 1). Algorithm 2 is a formulation of this approach where we additionally have seperated what we know from the maximum extinction.

A comparison for splitting along the primary ray in the scenario of shading from a large set of point light sources representing global illumination is given in Figure 2.

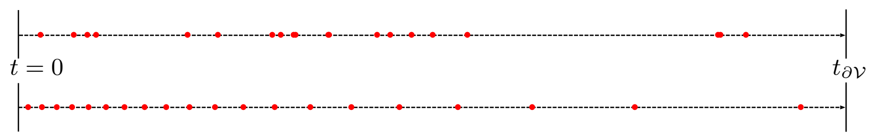
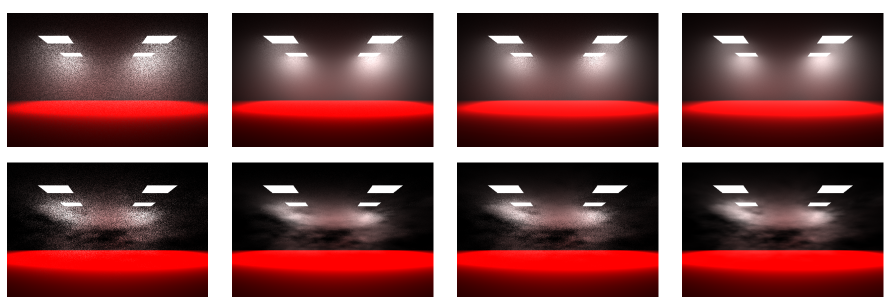


Fig. 1: Samples along a ray $x_0 + t\omega$ for $t \in [0, t_{\partial\mathcal{V}})$: transformed random points (top) and transformed equidistant points (bottom). The latter yield a better distribution for the initial points in Algorihm 1.



(a) Algortihm 1 (repeatedly) (b) Ray marching (random offsets) (c) Algorithm 2 (d) Reference image

Fig. 2: Comparison of splitting techniques along the primary ray for a homogeneous (top) and a highly inhomogeneous (bottom) hazy Mie medium at low sampling rates with the same computation time. Of course, our ray marcher implementation does not perform shading operations in regions where $\sigma_s = 0$. While perfectly unbiased, Algorithm 2 approaches the smoothness of ray marching with increasing homogeneity along the ray. The reference image has been computed using Algorithm 1 with a vast amount of samples.

3.2 Handling Multiple Wavelengths

It is a common notion in computer graphics that solving the transport equations separately for each wavelength is less efficient than simulating various wavelengths at once. While this may be true for moderately saturated colors, the general setting requires some additional considerations for sampling the BSDF. Color-dependent implementations of Russian roulette [SSKK03] can help but still cannot avoid infinite variance in general and are impractical for bidirectional path tracing, as the probability density evalutions for the heuristics become extremely complicated.

Using Algorithm 1 in a context where σ_t depends on the wavelength is also problematic when computing a single solution for several wavelengths. Consider sampling equation (13) with $c_{\mathcal{V}} \equiv 1$ in an infinite homogeneous medium for two wavelengths λ_1 and λ_2 with $\sigma_{t,\lambda_2} > \sigma_{t,\lambda_1}$ simultaneously, using the density $p_{\lambda_2}(t) = \sigma_{t,\lambda_2} e^{-\sigma_{t,\lambda_2} t}$. The variance on wavelength λ_2 is now zero, whereas the variance on λ_1 can be arbitrarily high:

$$V\left(\frac{\tau_{\lambda_1}(x \leftrightarrow y)}{p_{\lambda_2}}\right) = \begin{cases} \infty & \text{if } \sigma_{t,\lambda_2} \geq 2\sigma_{t,\lambda_1} \\ \left(2\sigma_{t,\lambda_1}\sigma_{t,\lambda_2} - \sigma_{t,\lambda_2}^2\right)^{-1} - \left(\sigma_{t,\lambda_1}\right)^{-2} & \text{otherwise.} \end{cases}$$

The problem can be avoided by limiting the extinction coefficient to a scalar value, i.e. by forcing $\sigma_{a,\lambda_1}(x)+\sigma_{s,\lambda_1}(x) = \sigma_{a,\lambda_2}(x)+\sigma_{s,\lambda_2}(x) = \sigma_t(x)$ for every pair of wavelengths λ_1, λ_2. This corresponds to situations where the extinction coefficent is a quanitity depending on volume particle density and size only. Then, the color of a medium is due to the reflection, refraction, and absorption probabilities of the particles themselves (in analogy to the BSDF) and may vary within this constraint. However, in scenarios where this restriction cannot be applied (e.g. atmospheric scattering) solving the transport equations for each wavelength separately should be preferred.

4 Applications

In order to obtain truly unbiased estimators for light transport, we incorporate the presented techniques into several Monte Carlo global illumination algorithms. These include simple and Bidirectional Path Tracing, an unbiased variant of Instant Radiosity for Participating Media, and a very robust version of the Metropolis Light Transport algorithm.

4.1 Path Tracing

Path Tracing [Kaj86] is one of the most basic global illumination algorithms. Due to its simplicity, it is still used frequently to compute reference solutions or handle complex scenes. A recently presented variant called Adjoint Photon

Tracing [MBE$^+$06] is capable of rendering difficult settings with participating media accurately. The simple form of *pure* Path Tracing without next event estimation, which does not estimate direct illumination at each path vertex explicitly, can be extended to handle participating media without much effort. The only modification is an additional distance sampling call for each ray that is cast.

The spatial sampling found in Path Tracing with next event estimation requires an estimate of the transmittance. This quantity is easily obtained with Algorithm 2. Note that the stability of Path Tracing may be compromised in this case: a path vertex can be sampled arbitrarily close to a light source, in which case the geometric term of the connection yields a weakly singular integrand with infinite variance. This is generally a problem for algorithms with next event estimation, though it is often avoided in a vacuum by only modeling light sources with a certain minimum distance to other surface points. With participating media the same problems arise from non-vacuum volume points near light sources. Such cases are usually much harder to avoid. However, the techniques presented below can be applied to handle the weak singularity in an unbiased manner.

4.2 Instant Radiosity

Instant Radiosity [Kel97] is a popular global illumination algorithm for scenes with predominantly diffuse surfaces. A set of transport paths is started from the light source in a preprocessing pass and point light sources are stored at each path vertex. The point lights, which represent path space samples, are then used to shade each camera ray's first interaction point.

Adapting this process for participating media using Algorithm 1 is straightforward. Note that the volume point light sources have to be equipped with a phase function instead of a BSDF. Shading the primary ray can be done by sampling a first interaction or, preferably, applying some splitting in the sense of Algorithm 2.

Solutions computed with Instant Radiosity can converge quite quickly, given that the weak singularity found in the shading path is avoided by bounding the geometric term [Kol04]. However, this approach introduces bias.

Bias Compensation

In order to handle the singularity without introducing bias we extend the method from [KK04] to participating media. We set the geometric term G as defined in equation (8) to G' by letting

$$G'(x \leftrightarrow y) := \begin{cases} G(x \leftrightarrow y) & \text{if } G(x \leftrightarrow y) < b \\ b & \text{otherwise} \end{cases} \tag{14}$$

for an arbitrary positive bound $b \in \mathbb{R}^+$. The bias introduced into the evaluation of the three point form $L(y \to z)$ as defined in equation (5) by replacing G with G' is

$$\begin{aligned} & L(y \to z) - L'(y \to z) \\ &= \int_{\mathbb{R}^3} L(x \to y) f(x \to y \to z) \cdot \max\{G(x \leftrightarrow y) - b, 0\} V(x \leftrightarrow y) d\lambda(x) \\ &= \int_{\mathbb{R}^3} L(x \to y) f(x \to y \to z) \frac{\max\{G(x \leftrightarrow y) - b, 0\}}{G(x \leftrightarrow y)} \\ &\qquad \cdot G(x \leftrightarrow y) V(x \leftrightarrow y) d\lambda(x) \\ &= \int_{\mathcal{S}^2} \int_0^{\|h(y,-\omega)-y\|} \tau(y - t\omega \leftrightarrow y) L(y - t\omega \to y) f(y - t\omega \to y \to z) \\ &\qquad \cdot \frac{\max\{G(y - t\omega \leftrightarrow y) - b, 0\}}{G(y - t\omega \leftrightarrow y)} dt d\sigma^*(\omega). \end{aligned} \tag{15}$$

In the last step we have changed the integration domain from $\mathbb{R}^3$ back to spherical coordinates, and depending on whether y is a surface or a volume point we have

$$d\sigma^*(\omega) = \begin{cases} |\cos\theta_y| d\sigma(\omega) & \text{if } y \in \partial\mathcal{V} \\ d\sigma(\omega) & \text{if } y \in \mathcal{V}. \end{cases}$$

This reformulation directly leads to a recursive algorithm for computing the bias which does not suffer from any singularities. We simply sample a direction ω according to the projected BSDF or the phase function, and a distance t according to τ. At the resulting next vertex $x = y - t\omega$, we recursively estimate $L(x \to y)$ and weight the result by $\frac{\max\{G(x \leftrightarrow y) - b, 0\}}{G(x \leftrightarrow y)}$. This weight is 0 for $G(x \leftrightarrow y) \le b$. If the next vertex is not close enough, we can thus terminate the path from the camera. In fact, the bounding and bias compensation step can be interpreted as s special case of Bidirectional Path Tracing where the weighting functions are constructed with respect to the value of the geometric term.

The efficiency of the approach is, of course, highly dependent on the choice of the bound b. Generally, one wants avoid bright spots in weakly illuminated areas caused by close by point lights. Options to achieve this include bounding point light contributions and bias compensation to the same maximum value [KK04] or using radiance estimates (e.g. based on direct illumination) and bound contributions of the point light sources to values below. Note that the contribution of the bias compensation step is always bounded by the brightness of the scene's light source.

4.3 Bidirectional Path Tracing

Combining Path Tracing and its adjoint approach, Light Tracing, leads to Bidirectional Path Tracing (BDPT) [LW93, VG94]. The algorithm uses a whole family of sampling techniques (Path Tracing and Instant Radiosity are two of them), which are combined using *multiple importance sampling* [VG95]. The *multi-sample estimator*

$$F = \sum_{i=1}^{n} \frac{1}{n_i} \sum_{j=1}^{n_i} w_i(x_{i,j}) \frac{f(x)}{p_i(x_{i,j})} \tag{16}$$

joins the samples $x_{i,j}$ that were created according to the density p_i. The estimator is unbiased as long as the weights $w_i(x)$ sum up to 1 for $f(x) \neq 0$ and are 0 for $p_i(x) = 0$. In fact, this is also true if the weights are only normalized in expectation, i.e.

$$E_y\Big(\sum_{i=1}^{n} w_i(x,y)\Big) = \int_{\Omega} \Big(\sum_{i=1}^{n} w_i(x,y)\Big) d\mu(y) = 1 \qquad \forall x. \tag{17}$$

A good choice for a weighting function that satisfies these conditions is the *power heuristic*

$$w_s(x) = \frac{p_s^{\beta}(x)}{\sum_i p_i^{\beta}(x)} \quad \text{ for } \beta \in \mathbb{R}^+. \tag{18}$$

For $\beta = 1$ we obtain the *balance heuristic* and $\beta \to \infty$ results in the *maximum heuristic*. As shown in [Vea97], these heuristics guarantee a fairly low variance.

The application of BDPT to participating media is given in [LW96] and can easily be modified to handle propagation events and transmittance estimations in an unbiased way by utilizing Algorithm 1 and Algorithm 2. However, the weighting heuristics need some additional consideration. For heterogeneous media, the densities $p_i \propto \tau$ due to propagation are not analytically computable and must therefore be approximated. Using Algorithm 2 for this purpose is generally not an option, as equation (17) fails for all but the maximum heuristic. However, note that no bias is introduced if we approximate τ by a deterministic quadrature rule, since the weights produced by equation (18) will definitely sum up to one then. Presuming a decent approximation, the good properties of the heuristics are preserved.

4.4 Metropolis Light Transport

Metropolis Light Transport (MLT) is a powerful alternative to the previous rendering approaches. The algorithm, which was first presented in [VG97] and strongly modified in [KSKAC02], uses Metropolis sampling [MRR+53] to sample the path space. Whereas ordinary importance sampling only considers factors of the measurement contribution in the previous algorithms, the flexibility of Metropolis sampling allows us to generate paths according to

$$p(\bar{x}) = \frac{f(\bar{x})}{b} \quad \text{with } b = \int_{\mathcal{P}} f(\bar{y})\, d\mu(\bar{y}).$$

This yields the importance sampling estimator

$$F_{j,N} = \frac{1}{N} \sum_{i=1}^{N} h_j\left(\bar{X}_i\right) b,$$

where the measurement contribution function $f_j(\bar{x})$ has been split into a pixel filter function $h_j(\bar{x})$ and a remainder $f(\bar{x})$, which is the same for every pixel.

The value of b can be estimated using any suitable rendering algorithm and a fairly low number of samples. Usually, 10^4–10^5 samples yield a sufficiently accurate estimate, which makes the cost of the initialization phase negligible. One of the approximation samples is selected as the initial state for the Metropolis phase of the algorithm with probability proportional to f/p. The selected sample is expected to be distributed according to the stationary density and thus avoids start-up bias without the need to discard any samples.

Once the first state has been selected, each new state is found by generating a tentative sample $\bar{y}$ from the current sample $\bar{x}$ according to the tentative transition function $T(\bar{x} \to \bar{y})$ and either accepting or rejecting the proposal according to the acceptance function

$$a(\bar{x} \to \bar{y}) := \min\left\{1, \frac{f(\bar{y})T(\bar{y} \to \bar{x})}{f(\bar{x})T(\bar{x} \to \bar{y})}\right\}.$$

A good proposal strategy is of paramount importance to the success of the algorithm. Key features of a sound strategy are the ability to exploit the coherence in the scene and a low correlation between subsequent samples.

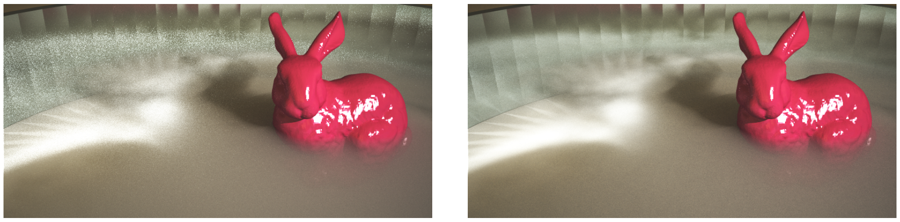
A number of important optimizations of the basic algorithm may be found in [Vea97]. While they will not be discussed here, they are of great importance to a successful implementation of MLT.

Adaptive Mutation

Kelemen et al. present a novel implementation of the MLT algorithm in [KSKAC02], which we adapt to handle participating media. The new mutation is simpler to implement than that proposed by Veach, reduces the correlation between samples, and is considerably more robust. Furthermore, the inclusion of participating media and other phenomena does not require extensive modifications to the mutation.

For any path tracing algorithm—e.g. classic Path Tracing or BDPT—, a path is uniquely defined by the set of random numbers used to create it. These numbers can be interpreted as a point in the infinitely-dimensional unit cube $[0,1)^\infty$, called the *primary sample space* $\mathcal{U}$. The transformation between the spaces, $s\colon \mathcal{U} \to \mathcal{P}$, is determined by the path tracing algorithm. The path integral presented in equation (3) can then be transformed to an integral over the primary sample space:

$$I_j = \int_{\mathcal{P}} f_j(\bar{x})\, d\mu(\bar{x}) = \int_{\mathcal{U}} \frac{f_j(s(\bar{u}))}{p(s(\bar{u}))} d\mu^*(\bar{u})\,,$$



(a) Bidirectional Path Tracing (b) Metropolis Light Transport

Fig. 3: A scene with a heterogeneous medium featuring caustics seen indirectly through a reflection. Both images were rendered with the same number of samples in approximately the same time. Splitting along the primary ray was employed to speed up the estimation of direct light for MLT.

where f and p are defined as before.

If the chosen path tracing algorithm properly employs importance sampling, the transformed integrand, $f^*(\bar{u}) := f(s(\bar{u}))\,/p(s(\bar{u}))$, will vary only moderately. Any mutation performed in the primary sample space will thus automatically adapt to the modalities of the integrand.

The proposed mutation generates a new path by perturbing the primary sample point that corresponds to the current path by a small exponentially distributed amount. This perturbation is symmetric, so that the acceptance probability simplifies to $a(\bar{u} \to \bar{v}) = \min\{1, f^*(\bar{v})\,/f^*(\bar{u})\}$.

Like Veach's original perturbations, the new mutation cannot ensure ergodicity by itself. To guarantee that the algorithm cannot get stuck in isolated areas of the scene, independent paths are generated at random intervals by the underlying path tracer. This is done by simply feeding a fresh set of uniform random numbers into the algorithm. The mutation type is chosen at random before each mutation step.

Transformation

The presented Metropolis algorithm only works efficiently if the transformation $s\colon \mathcal{U} \to \mathcal{P}$ ensures that a small change in $\mathcal{U}$ corresponds to a small change in the path space. Because of the robustness and efficiency of the approach, we use BDPT as the basic sample generation technique. Due to the separate generation of subpaths by BDPT and the inclusion of participating media, it is not possible to use values from successive dimensions of $\mathcal{U}$ as random numbers for the generation of samples and still satisfy the requirement of corresponding small changes. Rather, the values driving distinct parts of the path generation must be separated from each other.

Random numbers used for the generation of eye and light subpaths can simply be separated e.g. by assigning positive indices to one type of subpath

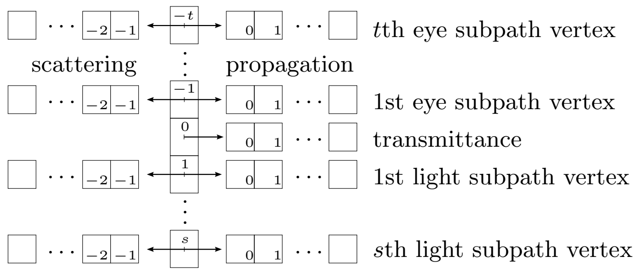


Fig. 4: A primary sample stored as an array of arrays. Random numbers are separated between light and eye subpaths, as well as between scattering and propagation.

and negative indices to the other type. While the number of random values needed to sample distances in homogeneous media is fixed, the amount of random input needed to drive the presented distance sampling routine cannot be determined in advance in scenes that contain heterogeneous media. In such cases, each deterministic connection between two subpaths also needs an additional unknown amount of random values to estimate the transmittance as described in section 3.1. Finally, some scattering models may need more input than others when sampling a direction.

We therefore propose storing the current primary sample $\bar{u}$ in an array of arrays as outlined in Figure 4. The ith row vector provides input to the sampling that is done at the $|i|$th vertex of the respective subpath. The elements of each vector are accessed in sequential order. This separation ensures that the random numbers that generated each vertex remain associated to that vertex. The estimation of the various path transmittance values may be driven by the row at index 0. A further separation is not necessary because the transmittance is fairly smooth in realistic settings. Large variations in the random values running the estimation thus have very little effect on the result.

5 Conclusion

We have presented unbiased global illumination algorithms for scenes with participating media. We thus close a gap in computer graphics where many algorithms are labeled as unbiased despite the fact that this claim was previously not true for heterogeneous media. The new approaches presented here allow for the physically accurate and efficient visualization of a wide range of scenes.

6 Acknowledgements

The first author has been funded by the project *information at your fingertips - Interaktive Visualisierung für Gigapixel Displays*, Forschungsverbund im Rahmen des Förderprogramms Informationstechnik in Baden-Württemberg (BW-FIT).

References

[Col68] W. Coleman. Mathematical Verification of a certain Monte Carlo Sampling Technique and Applications of the Technique to Radiation Transport Problems. *Nuclear Science and Engineering*, 32:76–81, 1968.

[Kaj86] J. Kajiya. The Rendering Equation. In *SIGGRAPH 86 Conference Proceedings*, volume 20 of *Computer Graphics*, pages 143–150, 1986.

[Kel97] A. Keller. Instant Radiosity. In *SIGGRAPH 97 Conference Proceedings*, Annual Conference Series, pages 49–56, 1997.

[KK04] T. Kollig and A. Keller. Illumination in the Presence of Weak Singularities. In D. Talay and H. Niederreiter, editors, *Monte Carlo and Quasi-Monte Carlo Methods*, pages 243–256. Springer, 2004.

[Kol04] T. Kollig. *Efficient Sampling and Robust Algorithms for Photorealistic Image Synthesis*. PhD thesis, University of Kaiserslautern, Germany, 2004.

[KSKAC02] C. Kelemen, L. Szirmay-Kalos, G. Antal, and F. Csonka. A Simple and Robust Mutation Strategy for the Metropolis Light Transport Algorithm. *Computer Graphics Forum*, 21(3):531–540, September 2002.

[LW93] E. Lafortune and Y. Willems. Bidirectional Path Tracing. In *Proc. 3rd International Conference on Computational Graphics and Visualization Techniques (Compugraphics)*, pages 145–153, 1993.

[LW96] E. Lafortune and Y. Willems. Rendering Participating Media with Bidirectional Path Tracing. *Rendering Techniques '96 (Proc. 7th Eurographics Workshop on Rendering)*, pages 91–100, 1996.

[MBE+06] R. Morley, S. Boulos, D. Edwards, J. Johnson, P. Shirley, M. Ashikhmin, and S. Premože. Image Synthesis using Adjoint Photons. In *Graphics Interface '06*, pages 179–186, June 2006.

[MRR+53] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of State Calculations by Fast Computing Machine. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[PH89] K. Perlin and E. Hoffert. Hypertexture. In *SIGGRAPH '89: Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 253–262, 1989.

[PKK00] M. Pauly, T. Kollig, and A. Keller. Metropolis Light Transport for Participating Media. In B. Péroche and H. Rushmeier, editors, *Rendering Techniques 2000 (Proc. 11th Eurographics Workshop on Rendering)*, pages 11–22. Springer, 2000.

[SSKK03] L. Szécsi, L. Szirmay-Kalos, and C. Kelemen. Variance Reduction for Russian Roulette. *Journal of WSCG*, 2003.

[Vea97] E. Veach. *Robust Monte Carlo Methods for Light Transport Simulation.* PhD thesis, Stanford University, 1997.

[VG94] E. Veach and L. Guibas. Bidirectional Estimators for Light Transport. In *Proc. 5th Eurographics Worshop on Rendering*, pages 147 – 161, Darmstadt, Germany, June 1994.

[VG95] E. Veach and L. Guibas. Optimally Combining Sampling Rechniques for Monte Carlo Rendering. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 419–428, New York, NY, USA, 1995. ACM Press.

[VG97] E. Veach and L. Guibas. Metropolis Light Transport. *Computer Graphics*, 31:65–76, 1997.