# Interner Bericht

# Instant Radiosity

Alexander Keller

287/97

## Fachbereich Informatik

Universität Kaiserslautern · Postfach 3049 · D-67653 Kaiserslautern

# Instant Radiosity

Alexander Keller

287/97

Universität Kaiserslautern
AG Numerische Algorithmen
Postfach 30 49
67653 Kaiserslautern
Germany

Jan. 1997

Herausgeber: AG Numerische Algorithmen
Leiter: Professor Dr. S. Heinrich

# Instant Radiosity

Research Paper

## Abstract

We present a fundamental procedure for instant rendering from the radiance equation. Operating directly on the textured scene description, the very efficient and simple algorithm produces photorealistic images without any kernel or solution discretization of the underlying integral equation. Rendering rates of a few seconds are obtained by exploiting graphics hardware, the deterministic technique of the quasi-random walk for the solution of the global illumination problem, and the new method of jittered low discrepancy sampling.

## 1 Introduction

Provided a realistic scene description, rendering from the radiance integral equation [Kaj86] yields realistic images. Under the assumption of diffuse reflection, the most popular approaches to approximate the solution of the Fredholm integral equation are radiosity algorithms. In the classical algorithms [CW93], the kernel of the radiance integral equation is projected onto some finite base, yielding the form factor matrix which is of quadratic order in the number of scene elements. For its sparse representation, hierarchical methods with hierarchical base functions have been introduced. Nevertheless, these Galerkin algorithms need to store the kernel and solution discretization of the integral equation. In addition to the high complexity of accurate mesh generation for shadow representation [LTG92], such projections introduce a discretization error.

From the domain of Monte Carlo simulation, algorithms without kernel discretization are available, using the random integration scheme for only projecting the solution onto a finite base. Similar to the random approaches, a deterministic particle simulation scheme based on low discrepancy sampling has been introduced in [Kel96]. This deterministic scheme converges smoother at a slightly superior rate and exposes no variance as compared to stochastic algorithms. In bidirectional path tracing [LW93, VG94], even the discretization of the solution of the radiance equation has been avoided, but the rendering time is far from realtime.

On the other hand graphics hardware is capable of illuminating and shadowing textured scenes by extended light sources [HA90, Hei91, SKvW$^+$92] in realtime.

In our new approach we combine the advantages of deterministic particle simulation of light, i.e. the quasi-random walk principle, with the available hardware capabilities to consistently render from the radiance equation, neither projecting the kernel nor the solution of the integral equation, resulting in a very fast, robust and straight forward to implement procedure.

Following this introduction, in the second section of this paper, we briefly resume the mathematical model of the global illumination problem. The third section explains the new rendering procedure and its underlying techniques of quasi-Monte Carlo integration and the quasi-random walk principle. After pointing out some extensions of the basic algorithm for including antialiasing by jittered low discrepancy sampling, specular effects, and modifications for realtime application in section four, the algorithm is discussed in the fifth section. The final section draws the conclusion and points out directions of future research.

## 2 Global Illumination

Our eyes perceive radiance, which is power per unit area per unit solid angle. In vacuum the radiance $L$ fulfills the *radiance equation* [Kaj86]

$$\begin{aligned} L(y, \vec{\omega}_r) &= L_e(y, \vec{\omega}_r) \\ &\quad + \int_\Omega f_r(\vec{\omega}_i, y, \vec{\omega}_r) L(h(y, \vec{\omega}_i), -\vec{\omega}_i) \cos\theta_i d\vec{\omega}_i, \end{aligned}$$

where $\Omega$ is the set of all directions $\vec{\omega} = (\theta, \phi)$ of the unit hemisphere aligned normal to the surface in point $y$. $S$ is the surface of the scene modeled as boundary representation. The function $h$ returns the first point hit when shooting a ray from $y$ into direction $\vec{\omega}_i$. The term $\cos\theta_i$ projects the incoming radiance normal to the surface, where $\theta_i$ if the azimuth angle between the surface normal in $y$ and the direction of incidence $\vec{\omega}_i$. The radiance $L$ in a point $y \in S$ into direction $\vec{\omega}_r \in \Omega$ so is the sum of source radiance $L_e$ and reflected radiance. Using operator notation we have the shorthand

$$L = L_e + T_{f_r} L.$$

The bidirectional reflectance distribution function $f_r$ accounts for the surface properties like color and gloss. In the general setting this function depends on the incident direction $\vec{\omega}_i$ and reflected direction $\vec{\omega}_r$ of radiance and the location $y$. In the radiosity setting

$f_r = f_d(y) := \frac{\rho_d(y)}{\pi}$ is restricted to only diffuse reflection. Then the radiance becomes isotropic, too:

$$L(y) = L_e(y) + \frac{\rho_d(y)}{\pi} \int_\Omega L(h(y, \omega_i)) \cos\theta_i d\omega_i,$$

where $\rho_d(y)$ is the reflectivity of the diffuse surface texture.

Given the quadruple $(S, f_r, L_e, \Psi)$, the *global illumination problem* consists in calculating functionals of the form

$$\langle L, \Psi \rangle := \int_S \int_\Omega L(y, \omega)\Psi(y, \omega) \cos\theta d\omega dy$$

either in the radiosity setting or for the full radiance equation. There are various choices for the detector functional $\Psi$, e.g. the sum of orthonormal base vectors of a finite vector space, as used in classical or hierarchical radiosity approaches [CW93]. Instead of discretizing the solution of the integral equation and then having to render it in a separate pass, we directly select

$$\Psi_{mn}(y, \vec{\omega}) := \frac{\delta(\vec{\omega} - \vec{\omega}_{y_f})}{\cos\theta} \frac{1}{|P_{nm}|} \chi_{P_{mn}}(h(y, \vec{\omega}))$$

detecting the average radiance passing through the pixel $P_{mn}$ of the image matrix as seen by a pinhole camera[1] from the focal point $y_f$. $\vec{\omega}_{y_f} = P - y_f$ is the direction of a point $P$ in the support of $P_{mn}$ through $y_f$. $\chi_{P_{mn}}$ is the characteristic function of the pixel's support and $\delta$ the Kronecker delta function.

# 3 The new Algorithm

Our new algorithm generates a particle approximation of the diffuse radiance in the scene. Then the graphics hardware renders an image with shadows for each particle used as point light source. Global illumination finally is obtained by summing up the single images in an accumulation buffer [HA90] and displaying the result. The particle density is generated by the quasi-random walk [Kel96] based on the method of quasi-Monte Carlo integration, both of which are explained in the next two sections. Before we derive the algorithm.

The average radiance passing through a pixel $P_{mn}$ is

$$\begin{aligned} \overline{L}_{mn} &:= \langle L, \Psi_{mn} \rangle = \langle L_e, \Psi_{mn} \rangle + \langle T_{f_r} L, \Psi_{mn} \rangle \\ &= \langle L_e, \Psi_{mn} \rangle + T_{mn} L, \end{aligned} \tag{1}$$

where the shorthand $T_{mn}L$ defines the rendering operator, which determines the at least once reflected radiance through $P_{mn}$. If the radiance $L$ in the radiosity setting can be approximated by a discrete density of $M$ point light sources

$$L(y) \approx \sum_{i=0}^{M-1} L_i \delta(y - P_i), \tag{2}$$

where $L_i$ is the radiance and $P_i$ is the position of the $i$-th light source, the application of $T_{nm}$ to the particle approximation, yields the very fast rendering algorithm

$$\overline{L}_{mn} \approx \langle L_e, \Psi_{mn} \rangle + \sum_{i=0}^{M-1} T_{mn} L_i \delta(y - P_i).$$

$T_{mn}$ applied to a point light source simultaneously can be evaluated for all pixels of the image matrix by calling a standard graphics hardware illumination routine in the manner of [Hei91, SKvW$^+$92], producing the shaded image of the textured scene including shadows. The directly visible light sources in $\langle L_e, \Psi_{mn} \rangle$ are rendered on the fly by assigning emission to the corresponding surface elements. The algorithm so directly operates on the textured scene description in image space and does not apply any kernel or solution discretization to the integral equation. In consequence no mesh artefacts will occur and also no topological data structure like an e.g. winged-edge representation is required for interpolation or overlapping coefficients evaluation. A small number $M$ of point light sources will be sufficient, since from multipass rendering with expensive local pass calculations [CSSD94], it is known, that a very coarse radiosity solution suffices to produce realistic images. So the speed of the algorithm in main depends on the frame generation rate of the graphics hardware, promising interactive rates of photorealistic image generation.

## 3.1 Quasi-Monte Carlo Integration

When integrating functions $f$ over the $s$-dimensional unitcube $[0, 1)^s$, usual Newton-Cotes style quadrature formulas are not applicable in high dimensions due to an exponential number of nodes or do not work at maximum efficiency when applied to discontinuous integrands. Independent of dimension and smoothness the *Monte Carlo method* [Nie92]

$$\int_{[0,1)^s} f(x)dx \approx \frac{1}{N} \sum_{i=0}^{N-1} f(x_i)$$

estimates the integral by averaging $N$ integrand values sampled at randomly chosen locations $x_i \in [0, 1)^s$. The error of the estimation is expected to be less than $\frac{\sigma(f)}{\sqrt{N}}$, where $\sigma^2(f)$ is the variance of $f$. Based on the central limit theorem, for sufficiently large $N$ with a probability of 99.7% the error of the estimation is less than $\frac{3\sigma(f)}{\sqrt{N}}$, but in computer graphics often $N$ is too small to guarantee this probability.

[1] For more elaborate camera models we refer to [KMH95].

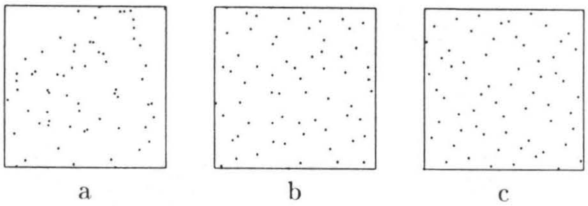

Figure 1: Two dimensional uniform sampling patterns: a) random, b) jittered, and c) Halton for $N = 64$ samples.

Since a computer is a deterministic automata, no real random numbers are available, and deterministic algorithms, like the linear congruence generator [Nie92], have to be used to simulate random numbers by their statistical properties. However profound investigations yield, that not randomness, but uniformity of the sampling pattern is important for the fast convergence of the quadrature.

A sequence of points $P_N = \{x_0, \ldots, x_{N-1}\}$ is uniformly distributed in $[0,1)^s$ if and only if its discrepancy

$$D^*(P_N) := \sup_{J \in \mathcal{J}^*} \left| \int_{I^s} \chi_J(x)\, dx - \frac{1}{N} \sum_{i=0}^{N-1} \chi_J(x_i) \right|$$

vanishes for $N \to \infty$ [Nie92]. The discrepancy is defined as the worst case error for integrating the characteristic functions $\chi_J$ of all axis-aligned subcubes $J$, including the origin, of the family $\mathcal{J}^* := \{\prod_{j=1}^{s} [0, a_j) \subset [0,1)^s\}$, using the sample points $P_N$. In the *quasi-Monte Carlo method* [Nie92] the random samples with a discrepancy of almost surely $\mathcal{O}\left(\sqrt{\frac{\log \log N}{N}}\right)$ are replaced by deterministic ones, especially designed for integration. Lacking almost any statistical properties, these so-called *low discrepancy points* are more uniformly distributed, expose faster convergence rates and allow deterministic error bounds as opposed to the Monte Carlo method.

Based on the radical inverse function

$$\Phi_b(i) := \sum_{j=0}^{\infty} a_j(i)\, b^{-j-1} \in [0,1) \Leftrightarrow i = \sum_{j=0}^{\infty} a_j(i)\, b^j,$$

the $s$-dimensional Halton low discrepancy sequence (see figure 1) is

$$x_i = (\Phi_{b_1}(i), \ldots, \Phi_{b_s}(i))\,,\ i \in \mathbb{N},$$

where $b_j$ is the $j$-th prime number. This sequence has a discrepancy of $\mathcal{O}\left(\frac{\log^s N}{N}\right)$. Note, that each segment $P_{N'}$ of a larger segment $P_N$, $N' < N$, of successive points of the Halton sequence is of low discrepancy, too, which is not the case for variance reduced sampling methods like jittered or $N$-rooks sampling. In addition the Halton points are available for any choice of $N$ independent of dimension $s$. Compared to $N$-rooks sampling, where for each dimension a random permutation of size $\mathcal{O}(N)$ and unknown quality has to be stored, the Halton points can be generated for arbitrary $i$ (see figure 2) or successively (see figure 3) by the algorithm [HW64] at a speed comparable to usual pseudo-random generators without additional storage.

```
void $\Phi$(int $b$, int $i$)
{
    double $x$ = 0.0, $f$ = $\frac{1}{b}$;

    while($i$)
    {
        $x$ += $f$ * (double) ($i$ % $b$);
        $i$ /= $b$;
        $f$ *= $\frac{1}{b}$;
    }

    return $x$;
}
```

Figure 2: Pseudocode for the direct calculation of the radical inverse function.

The discrepancy is the order of convergence for functions of certain smoothness (for details see [Nie92]), promising quasi-Monte Carlo integration to be roughly quadratically faster than the Monte Carlo method. The integrands in computer graphics, however, are discontinuous, allowing only very pessimistic upper error bounds. Nevertheless, the numerical evidence in [Kel96] shows that the calculation of functionals of the solution of the radiance equation by means of low discrepancy sequences results in a smoother convergence and slightly superior rate as compared to random sampling. In [PTVF92] a plausibility argument gives the rate of $\mathcal{O}(N^{-\frac{s+1}{2s}})$ as upper bound for the quasi-Monte Carlo method applied to discontinuous functions. For high dimension, this rate converges to the random rate of $\mathcal{O}(N^{-\frac{1}{2}})$, but since $s \ll \infty$, the rate of sampling with Halton points is superior to random sampling. The above arguments also apply to jittered sampling [Mit96], but the low discrepancy pattern is deterministic and therefore works without variance!

## 3.2 The Quasi-Random Walk

In realistic applications the transport operator norm $\|T_{f_r}\| < 1$, meaning that less than 100% of the incident radiance is reflected. So the Neumann series

$$L = (I - T_{f_r})^{-1} L_e = \sum_{j=0}^{\infty} T_{f_r}^j L_e$$

converges and can be used to solve the integral equation. Inserted in (1), after some transformations, for

```
void Φ(int $b$, double $x$)
{
    double $h$, $hh$, $r$ = 1.0 - $x$ - 1e - 10;

    if($\frac{1}{b}$ < $r$)
        $x$ += $\frac{1}{b}$;
    else
    {
        $h$ = $\frac{1}{b}$;

        do
        {
            $hh$ = $h$;
            $h$ *= $\frac{1}{b}$;
        }
        while($h$ >= $r$);

        $x$ += $hh$ + $h$ - 1.0;
    }

    return $x$;
}
```

Figure 3: Pseudocode for the incremental calculation of the radical inverse function.

the radiosity setting we get

$$T_{mn}L = \frac{1}{|P_{mn}|}\sum_{j=0}^{\infty}\int_{P_{mn}}\int_{\Omega^j}\int_{S_e} p_j(y_0,\vec{\omega}_0,\ldots,\vec{\omega}_j)$$
$$V(y_j,y')f_d(y')\frac{\cos\theta_j\cos\theta'}{|y_j-y'|^2}dy_0 d\omega_0\cdots d\omega_j dP.(3)$$

Here $S_e$ is the support of the light sources, and $y' = h(y_f, P - y_f)$ is the first point hit, when shooting a ray from the eye at $y_f$ through the point $P \in P_{mn}$ into the scene. $V(y_j, y')$ checks the mutual visibility of the points $y_j$ and $y'$, yielding 1 in case of visibility and 0 else. Similar to [Kel96], the radiance density

$$p_j(y_0,\vec{\omega}_0,\ldots,\vec{\omega}_j) := L_e(y_0)\prod_{l=1}^{j}(\cos\theta_{l-1}f_d(y_l)),$$

is simulated by its particle nature using the technique of the quasi-random walk. Here $y_0 \in S_e$ is a point on a light source, and the subsequent points $y_{j+1} := h(y_j, \vec{\omega}_j)$, $j \in \mathbb{N}$, of a path are determined by ray shooting. Taking the diffuse part of the scene, the operator norm can be estimated by the mean reflectivity

$$\overline{\rho} := \frac{\sum_{k=1}^{K}\rho_{d,k}|A_k|}{\sum_{k=1}^{K}|A_k|} \approx \|T_{f_d}\|,$$

where the scene is composed out of $K$ surface elements $A_k$ with average diffuse reflectivity of $\rho_{d,k}$. Assuming $\overline{\rho}$ as overall reflectivity for the moment, from $N$ particles started at the light sources, $\overline{\rho}N$ particles are supposed not to be absorbed by the first reflection, $\overline{\rho}^2 N$ survive the second reflection and so on. Since in realistic scene models, the actual diffuse reflectivity has only small deviation from $\overline{\rho}$, we can use fractional absorption and avoid Russian Roulette absorption [AK90]. Applying a random walk scheme, $T_{mn}L_e$ is evaluated using $N$ point lights, $T_{mn}T_{f_d}L_e$ by using $\lfloor\overline{\rho}N\rfloor$ point lights, and so on. To generate this discrete density approximation of $p_j$ by low discrepancy points, we first fix the number $N$ of particles to start off the light source. By an isometry the first two components of the Halton sequence are mapped from the unitsquare onto the surface of the light source, yielding the starting point $y_0(\Phi_2, \Phi_3)$ with power $L_e(y_0)$ supp $L_e$. In the case of multiple light sources, first a light source is selected by the composition method identical to [Kel96], then the isometry is applied. Exploiting the property of the Halton sequence, that segments of the sequence have small discrepancy, too, the $\lfloor\overline{\rho}N\rfloor$ first points are used to shoot a ray into direction $\vec{\omega}_0$ using

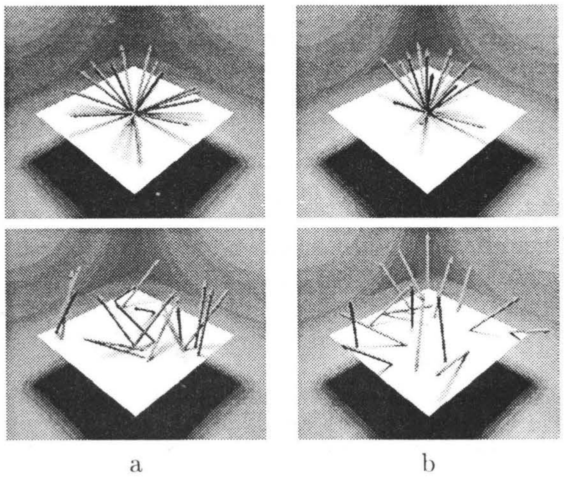

Figure 4: Diffuse scattering modeled by a) pseudo-random and b) Halton samples for $N = 16$.

$$\vec{\omega}_j = \vec{\omega}(\Phi_{b_{2j+2}}, \Phi_{b_{2j+3}}) = (\arcsin\sqrt{\Phi_{b_{2j+2}}}, 2\pi\Phi_{b_{2j+3}}),$$

where the direction already is distributed with respect to the cos-term in the density $p_j$. In $y_1 = h(y_0, \vec{\omega}_0)$ the particle's radiance is attenuated by $f_d(y_1)$. From these particles the first $\lfloor\overline{\rho}^2 N\rfloor$ continue their paths, repeating the procedure until no particles remain. The starting points $y_0$, and the subsequent hitpoints of the above quasi-random walk then form the discrete density approximation (2), which is used for the hardware lighting in our algorithm. In order to use the full accuracy of the frame buffer, the attenuation by $\prod_{l=1}^{j} f_d(y_l)$ is compensated by the factor $\frac{N}{\lfloor\overline{\rho}^j N\rfloor}$, making the contribution of each image equally important. The quasi-Monte Carlo integration of all images is performed by accumulating the images with the weight $\frac{1}{N}$ (see pseudo-code

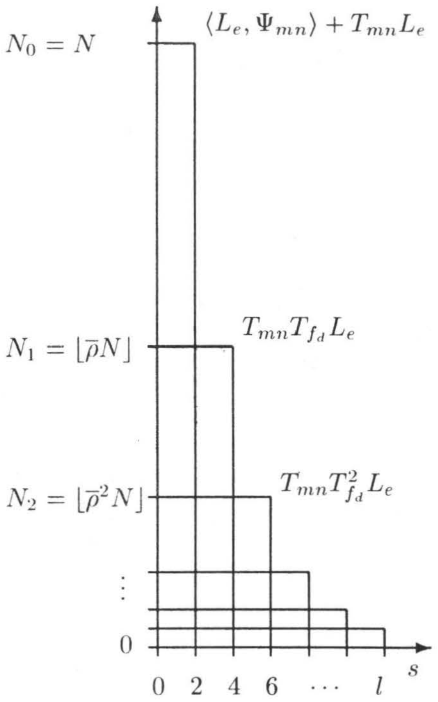
$N_0 = N$
$\langle L_e, \Psi_{mn} \rangle + T_{mn} L_e$
$N_1 = \lfloor \overline{\rho} N \rfloor$
$T_{mn} T_{f_d} L_e$
$N_2 = \lfloor \overline{\rho}^2 N \rfloor$
$T_{mn} T_{f_d}^2 L_e$
$\vdots$
0
0 2 4 6 $\cdots$ $l$
$s$

Figure 5: The quasi-random walk integration scheme.

in figure 6). The number $M$ of radiance points generated is bounded by

$$M < \sum_{j=0}^{\infty} \overline{\rho}^j N = \frac{1}{1 - \overline{\rho}} N =: \overline{l} N,$$

and so is linear in $N$ depending on the average scene reflectivity $\overline{\rho}$, where $\overline{l}$ is the mean path length.

The quasi-random walk concentrates the particles in the lower powers of the reflection operator, which due to the operator norm contribute the most important parts of the image, thus fully exploiting the advantages of low discrepancy sampling. The few particles remaining for higher order reflections, show at worst random convergence behavior (see previous section). This integration scheme is illustrated in figure 5. The starting points and directions, and the scattering directions according to the $\cos\theta$-distribution generated by random and Halton samples can be seen in figure 4. Obviously the deterministic low discrepancy samples are more uniformly distributed than the random samples.

# 4 Extensions

The fast, deterministic radiosity algorithm introduced in the previous section consistently renders diffuse global illumination for still images. By the new concept of jittered low discrepancy sampling, we treat issues of antialiasing in order to improve image quality at low sampling rates by random elements. Then specular effects are added to the algorithm. Finally modifications for realtime walkthroughs are indicated.

```
double Weight, Start = $N$;
int i, j, End, Reflections = 0;

while(True)
{
    End = (int) Start;
    Start *= $\overline{\rho}$;

    if((int) Start >= End)
        break;

    for($i$ = (int) Start; $i$ < End; $i$++)
    {
        $y$ = $y(\Phi_2(i), \Phi_3(i))$;
        $L$ = $L_e(y)$ * supp $L_e$;
        Weight = $N$;

        for($j$ = 0; $j$ <= Reflections; $j$++)
        {
            RenderSceneWithLight($L$, $y$);
            AccumulateWeighted(1.0 / $N$);
            $\vec{\omega} = \vec{\omega}(\Phi_{b_{2j+2}}(i), \Phi_{b_{2j+3}}(i))$;
            $y = h(y, \vec{\omega})$;
            Weight *= $\overline{\rho}$;
            $L$ *= $f_d(y) \frac{\text{(double) } N}{\text{floor(Weight)}}$;
        }
    }

    Reflections++;
}

DisplayAccBuffer();
```

Figure 6: Pseudocode for the new algorithm using the quasi-random walk for discrete density approximation of radiance.

## 4.1 Jittered Low Discrepancy Sampling

The quasi-random walk density approximation is deterministically modeled by the Halton sequence. Taking a look at the two-dimensional Hammersley sequence $(\frac{i}{N}, \Phi_2(i))_{i=0}^{N-1}$ in figure 7, it becomes obvious, that the low discrepancy points based on radical inversion are aligned to a grid. This grid structure guarantees a minimum distance property, and as such an implicit stratification, but has high aliasing amplitudes in its Fourier transform. Choosing $N = b^n$ samples, we have

$$\inf_{i \neq j} (\Phi_b(i) - \Phi_b(j)) = \frac{1}{N}$$

as the grid resolution for base $b$. Since $N$ usually is not a power of the base, the grid resolution $\frac{1}{b^n}$ is determined by $b^{n-1} < N <= b^n$.

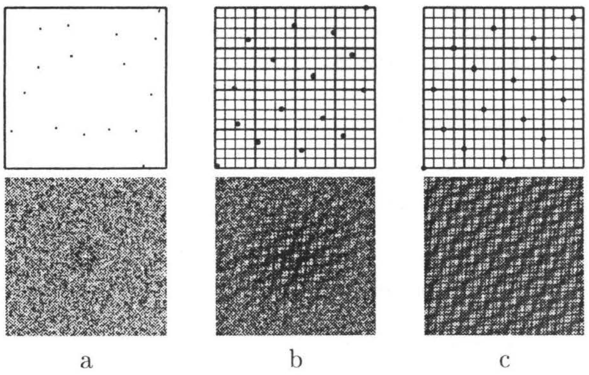

Figure 7: Two dimensional uniform sampling patterns for $N = 16$: a) Poisson disk, b) jittered Hammersley, and c) Hammersley and their Fourier transforms for $N = 64$ samples.

The new concept of jittered low discrepancy sampling now joins the two worlds of Monte Carlo and quasi-Monte Carlo integration, by using low discrepancy point sets as stratification. This is done by randomizing each low discrepancy point in its raster cell (see figure 7), replacing the radical inverse $\Phi_b$ by $\Phi_b + \frac{\xi}{b^n}$, where $\xi$ is a random variable. Assuring that the sample remains in the unit interval, the grid resolution can be approximated by $\frac{1}{N}$. Since a finite set of samples always will alias when applied to not bandlimited functions, random elements are required to attenuate the aliasing effects. At low sampling rates, jittered low discrepancy sampling so maps high frequencies to noise and reproduces low frequencies correctly, approximating the properties of the Poisson disk pattern. From the images in figure 7 it also becomes obvious, that the jittered Hammersley sequence in two dimensions is a special case of $N$-rooks sampling. But the Hammersley points can be generated without storing a random permutation; they implicitly are a permutation with low discrepancy.

Jittered low discrepancy sampling can be applied twice in our algorithm. Applied to pixel supersampling for antialiasing as in [HA90], the two dimensional jittered Hammersley sequence exposes an even faster convergence than jittered or $N$-rooks sampling. This can be seen in figure 8, where the RMS-error is plotted versus the sampling rate $N$ for an experiment where images of the textured scene in figure 11 at sampling rate $N = 1 \dots 64$ were compared to a master calculation at 640 samples. In our new algorithm then, according to the path number $i$, the corresponding $x_i$ of the Hammersley sequence is fixed, and jittered for each image produced by the particles of the path. Since $N$ usually is large enough, this way hardware antialiasing, available on some graphics accelerators, becomes redundant. Concerning the quasi-random walk, the components of the Halton vector have to be jittered

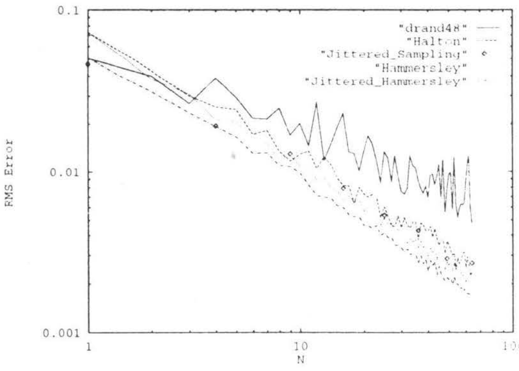

Figure 8: Convergence of different sampling patterns.

by different ranges. Starting on the light sources we have $N$ particles, where the coordinates $(\Phi_2, \Phi_3)$ will be jittered by $\frac{1}{N}$, in the next step only $\lfloor \bar{\rho} N \rfloor$ particles are traced, so the jitter range is $\frac{1}{\lfloor \bar{\rho} N \rfloor}$, and so on. Using this procedure, the grid structure of the Halton sequence is resolved, even improving the discrepancy.

## 4.2 Specular Effects

The particle density generated by the quasi-random walk only approximates the diffuse radiance. To add the specular effects we first let $T_{mn}$ use the full BRDF $f_r$ in the hardware lighting pass, enabling specular highlights as can be seen in figure 11. In the particle generation phase, by a random decision each surface is tested to be specular or diffuse according to its BRDF [CRMT91]. In case of specular reflection a virtual light source is generated, i.e. the origin af the ray is mirrored by the specular surface under consideration. The virtual light source now illuminates the part of the scene inside the pyramid spanned by itself and the contour of the reflecting element using techniques of [DB94]. Note that virtual light sources can only be applied at planar polygon level. Then the incoming particle is reflected by a random decision according to the specular part of the BRDF. Particles hitting specular surfaces so produce a virtual light source raising $M$ and cause a lengthening of the low discrepancy path by a random piece. If the graphics hardware supports spot lights and the particles $(L_i, P_i, \vec{\omega}_i)$ are equipped with their direction of incidence, even more general light source emission, i.e. with a $\cos^d$-distribution, and specular scattering can be simulated. Finally the visible specular objects have to be treated separately by ray tracing or advanced hardware techniques of [DB94].

## 4.3 Realtime Walkthroughs

The algorithm designed so far produces still images. In an animated environment, the variance reduction by avoiding Russian Roulette absorption now must be abandoned. The scattering decisions of the quasi-random walk then still are taken by (jittered) low discrepancy points to optimally exploit spatial coherence, but the absorption has to be modeled by pseudo-random numbers [AK90]. All images produced by one path are accumulated and the resulting image is stored with its time of generation. Keeping the last $N$ images of the last $N$ paths, each time a new path is completed, the oldest image is replaced. The current $N$ images then are accumulated and displayed, so implicitly performing temporal antialiasing. Still using $1 - \overline{\rho}$ as overall absorption probability, the mean path length $\overline{l}$ usually ranges from 1 to 10, and only $\overline{l}$ images are expected to be generated for one time step, allowing for real-time rendering rates! Since our algorithm operates on the scene graph, animated environments can be treated without further effort.

Using texture mapping hardware for displaying illumination maps, generalizing the method of [HH96] yields another approach to realtime walkthroughs. Instead of using jittered sampling and $\overline{\rho} = \theta$, i.e. only direct illumination, we replace the light samples of [HH96] by our discrete density approximation of radiance. By this simple enhancement the algorithm of [HH96] renders the global diffuse illumination into textures, which then interactively can be displayed. Besides considerable memory consumption, the solution of the radiance equation now is discretized in textures, which may result in visible artefacts, if the texture resolution has been chosen to small.

# 5 Discussion of the Algorithm

The algorithm is illustrated in figure 9 for a path number of $N = 9$, where the single images created by the point light sources are shown in analogy to the graph in figure 5. The result of accumulating $N = 128$ paths (i.e. 296 images in PAL-resolution 720x576 pixels) is shown in figure 10. This image of a scene of 402 quadrangles has been produced on a Silicon Graphics Onyx with Reality Engine$^2$ graphics and a 75MHz R8000 processor in 24 seconds seconds by the shadow algorithm of [Hei91]. Using the shadow techniques of [SKvW$^+$92] would result in an at least twice as fast algorithm. This emphasizes the fact, that the performance of our approach in main depends on the hardware illumination speed, since the particle approximation can be generated instantly. Note, that the smooth shadows and the indirect illumination are obtained without any meshing.

Exploiting shadow caching and eye ray coherence, the point light sources are also suited for evaluating $T_{mn}$ by ray tracing. Since our access to graphics hardware was restricted to the above example, we used this method to simulate graphics hardware for the conference room[2] of 39584 scene primitives. Independent of the number of extended light sources, the image in figure 12 demonstrates the power of the method even for the large number of 248 light sources for only $N = 128$ paths.

Since the algorithm directly operates on the scene graph without additional storage for discretizations, animated environments or cyclic graphs as used for plant modeling easily can be rendered in a photorealistic way. The only additional data structure required is a space order like e.g. a BSP-tree for accelerating the ray shooting $h(y, \vec{\omega})$. On the one hand it is possible to generate the discrete density approximation $(L_i, P_i)_{i=0}^{M-1}$ of the radiance and to affix these light sources to the scene description, e.g. MGF or VRML. Then the final rendering process, i.e. loading the scene graph and illuminating it by the point light sources, does not need the space order for ray shooting. On the other hand the BSP can be used for rendering with imposters similar to [SLS$^+$96] and hierarchical clipping, speeding up the frames-per-second rate. These techniques have not yet been included in the implementation used for the time measurements, but reduce the constant preceeding the time complexity of $\mathcal{O}(NK)$ of our algorithm, where the number $N$ of paths can be freely chosen, with respect to the frame and accumulation buffer accuracy, and $K$ is the number of elements in the scene.

Two minor problems of the algorithm become apparent at very low sampling rates $N$. The first problem is the weak singularity of the operator $T_{mn}$, i.e. when the distance $|y_j - y'|^2$ of point light source $y_j$ and point $y'$ to be lit comes close to zero (see figure 9). Then the value to be entered into the frame buffer is overmodulated and will be clipped to the maximal representable value. The second problem is that each light point colored by a texture has a large influence on the overall color of the scene. But since all images are weighted by $\frac{1}{N}$ in the accumulated image, the impact of one of the above cases is at most of order $\frac{1}{N}$, which in the most cases is hardly perceivable.

# 6 Conclusion and Future Work

A new method for rendering from the radiance equation has been introduced. Based on the quasi-random walk, point light sources are generated for fast hardware illumination. The single images are superimposed, yielding one of the fastest and physically correct rendering procedures. Working in image space, the algorithm does not need any storage for kernel or solution discretization or topological information. The

[2] The models of this paper (except for figure 11) are taken from the *material and geometry format* (MGF)-package of Greg Ward (available via `http://radsite.lbl.gov/mgf/HOME.html`). The greyscale scene has been modeled by Peter Shirley. The conference room has been modeled by Anat Grynberg and Greg Ward. A photograph of the original room can be found in [FvDFH90].

efficient algorithm itself is very compact and easily implemented using a standard graphics API, requiring only a ray intersection routine and an isometry from the unit square onto the surface of each light source. The deterministic algorithm has been extended by the new concept of jittered low discrepancy sampling.

Since our method already includes importance sampling, stratification of low discrepancy, and jittering for antialiasing, future work will be spent on the dual integral equation for efficiently sampling the light sources with high impact on the final image. A further reduction of the number $M$ of required point lights can be obtained by using iteration schemes, where approximate solutions for small $N'$ are used for variance reduction in a pass using $N > N'$. Provided, that the rendering hardware supplies fog attenuation, even an extension to participating media is possible, since the direct simulation is easily extended for volume scattering.

## 7 Acknoledgements

We would like to thank Stefan Heinrich and Hans-Christian Rodrian for the discussions. Special thanks go to Marc Stamminger, providing access to the Reality Engine$^2$, and to the group of Hans Hagen for letting me use their graphics equipment.

## References

[AK90] J. Arvo and D. Kirk. Particle Transport and Image Synthesis. In *Computer Graphics (ACM SIGGRAPH Proceedings)*, pages 63 – 66, 1990.

[CRMT91] S. Chen, H. Rushmeier, G. Miller, and D. Turner. A progressive Multi-Pass Method for Global Illumination. In *Computer Graphics (ACM SIGGRAPH Proceedings)*, pages 165 – 174, 1991.

[CSSD94] P. Christensen, E. Stollnitz, D. Salesin, and T. DeRose. Wavelet Radiance. In *Proc. 5th Eurographics Workshop on Rendering*, pages 287–302, 1994.

[CW93] M. Cohen and J. Wallace. *Radiosity and Realistic Image Synthesis.* Academic Press Professional, Cambridge, 1993.

[DB94] P. Diefenbach and N. Badler. Pipeline Rendering: Interactive Refractions, Reflections, and Shadows. *Displays: Special Issue on Interactive Computer Graphics*, 15(3):173–180, 1994.

[FvDFH90] J. Foley, A. van Dam, S. Feiner, and J. Hughes. *Computer Graphics, Principles and Practice, 2nd Edition.* Addison-Wesley, 1990.

[HA90] P. Haeberli and K. Akeley. The Accumulation Buffer: Hardware Support for High-Quality Rendering. In *Computer Graphics (ACM SIGGRAPH Proceedings)*, pages 309–318, 1990.

[Hei91] T. Heidmann. Real Shadows - Real Time. *Iris Universe*, (18):28–31, 1991.

[HH96] M. Herf and P. Heckbert. Fast Soft Shadows. In *Technical Sketches (ACM SIGGRAPH Visual Proceedings)*, page 145, 1996.

[HW64] J. Halton and G. Weller. Algorithm 247: Radical-inverse quasi-random point sequence. *Comm. ACM*, 7(12):701–702, 1964.

[Kaj86] J. Kajiya. The Rendering Equation. In *Computer Graphics (ACM SIGGRAPH Proceedings)*, pages 143–150, 1986.

[Kel96] A. Keller. Quasi-Monte Carlo Radiosity. In X. Pueyo and P. Schröder, editors, *Rendering Techniques '96 (Proc. 7th Eurographics Workshop on Rendering)*, pages 101–110. Springer, 1996.

[KMH95] C. Kolb, D. Mitchell, and P. Hanrahan. A Realistic Camera Model for Computer Graphics. In *Computer Graphics (ACM SIGGRAPH Annual Conference Series)*, pages 317–324, 1995.

[LTG92] D. Lischinski, F. Tampieri, and D. Greenberg. Discontinuity Meshing for Accurate Radiosity. *IEEE Computer Graphics & Applications*, 12(6):25–39, 1992.

[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.

[Mit96] D. Mitchell. Consequences of Stratified Sampling in Graphics. In *Computer Graphics (ACM SIGGRAPH Annual Conference Series)*, pages 277–280, 1996.

[Nie92] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods.* SIAM, Pennsylvania, 1992.

[PTVF92] H. Press, S. Teukolsky, T. Vetterling, and B. Flannery. *Numerical Recipes in C.* Cambridge University Press, 1992.

[SKvW+92] M. Segal, C. Korobkin, R. van Widenfelt, J. Foran, and P. Haeberli. Fast shadows and lighting effects using texture mapping. In *Computer Graphics (ACM SIGGRAPH Proceedings)*, pages 249–252, 1992.

[SLS+96] J. Shade, D. Lischinski, D. Salesin, T. DeRose, and J. Snyder. Hierarchical Image Caching for Accelerated Walkthroughs of Complex Environments. In *Computer Graphics (ACM SIGGRAPH Annual Conference Series)*, pages 75–82, 1996.

[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.

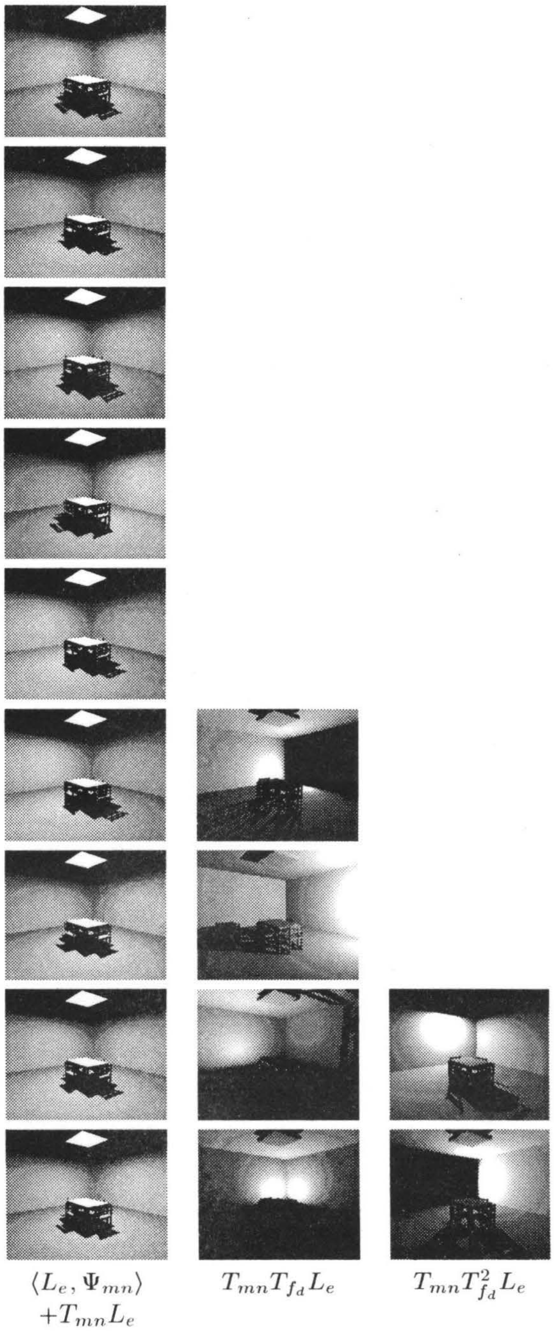

Figure 9: Illustration of the integration scheme in figure 6 by the intermediate images accumulated by the quasi-random walk for $N = 9$ paths.

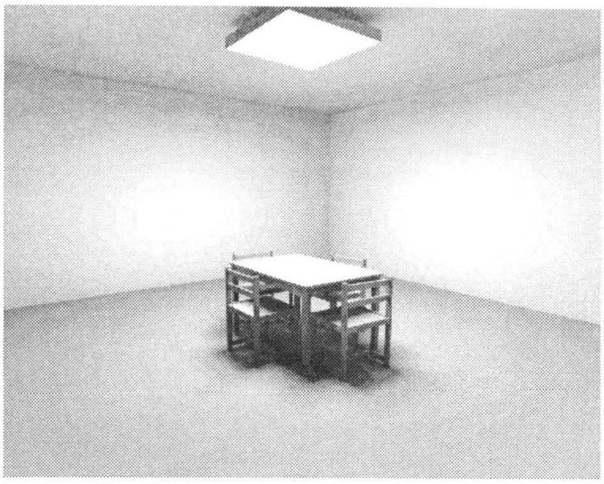

Figure 10: Final image as result of the accumulation in figure 9, but for $N = 128$.

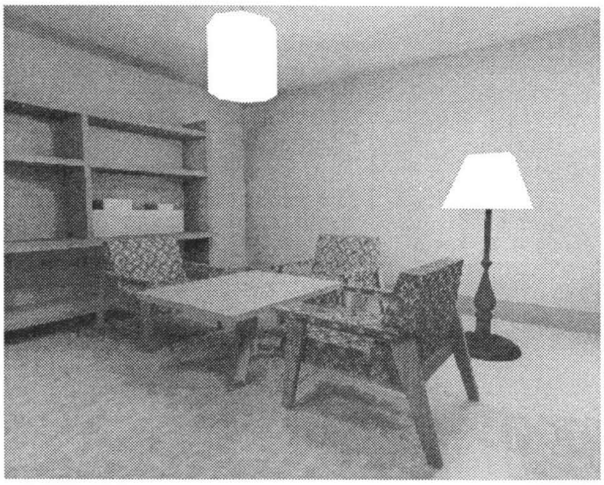

Figure 11: Specular effects (on the floor) by using full BRDF $f_r$ in $T_{mn}$ for $N = 128$.

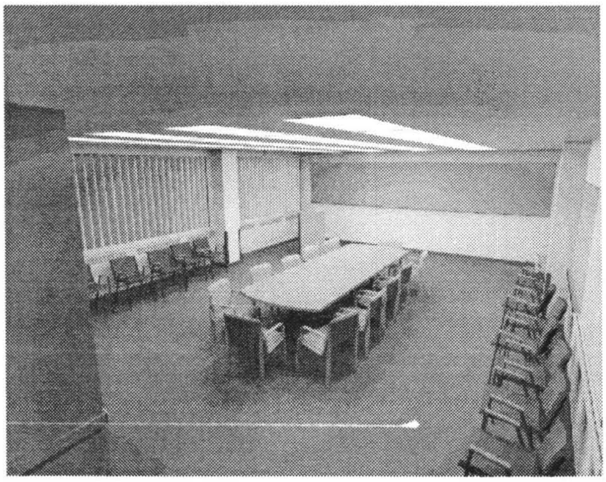

Figure 12: Conference room image for $N = 128$.