ON THE START-UP BIAS PROBLEM OF METROPOLIS SAMPLING [1]

László Szirmay-Kalos, Péter Dornbach, Werner Purgathofer

Department of Control Engineering and Information Technology, TU of Budapest
Budapest, Műegyetem rkp. 11, H-1111, HUNGARY
szirmay@fsz.bme.hu

ABSTRACT

The paper presents an analysis of the start-up bias problem of Metropolis sampling. The analysis is carried out both theoretically and using simulations. In order to allow theoretical treatment, a simplified model is established and the convergence of the Metropolis sampling is examined by Fourier analysis. It is concluded that the start-up bias can be quite significant if the integrand is relatively uniform, thus Metropolis sampling becomes really efficient only for difficult integrands, that is for difficult lighting situations. The theoretical results are then validated for two different integral formulations of the rendering equation. The first is based on bi-directional path tracing, and the second is on ray-bundle tracing.

Keywords: Metropolis sampling, importance sampling, Monte-Carlo integration, rendering equation

1 Introduction

Random walk solutions of the rendering equation obtain the radiance in the form of an infinite Neumann series. The terms of this Neumann series are high-dimensional integrals that are evaluated by Monte-Carlo or quasi-Monte Carlo quadrature in order to avoid dimensional explosion.

The fundamental idea of Monte-Carlo quadrature is to convert the integral to an expected value, which is then estimated by the average of samples:

$$\int_V f(\mathbf{z})\ d\mathbf{z} = \int_V \frac{f(\mathbf{z})}{p(\mathbf{z})} \cdot p(\mathbf{z})\ d\mathbf{z} = E\left[\frac{f(\mathbf{z})}{p(\mathbf{z})}\right] \approx$$

$$\frac{1}{M} \cdot \sum_{i=1}^{M} \frac{f(\mathbf{z}_i)}{p(\mathbf{z}_i)} \pm \frac{\sigma}{\sqrt{M}}, \qquad (1)$$

where $p(\mathbf{z})$ is a probability density in V, the $\mathbf{z}_i$ points are selected according to this probability density, and σ is the variance of the estimation. The probability density $p(\mathbf{z})$ should be selected to minimise the variance. Thus in Monte-Carlo integration it is worth applying probability distributions that are large where the integrand is large and small where the integrand is negligible. This variance reduction technique is called the *importance sampling* [Sobol91].

In the context of the rendering equation, importance sampling prefers useful paths along which significant radiance is transferred. Since a path carries energy on several different wavelengths, the integrand $f(\mathbf{z})$ is a vector, thus the selection of a "proportional" probability density requires further considerations. In order to express where the ele-

[1] This work has been supported by the National Scientific Research Fund (OTKA ref.No.: F015884, T029135), the Austrian-Hungarian Action Fund (ref.No.: 29ö4 and 32öu9) and the Spanish-Hungarian Fund (ref.No.: E9)

ments of vector $f(\mathbf{z})$ are large, a scalar importance function $\mathcal{I}(\mathbf{z})$ is defined. This importance function can, for example, represent the luminance of the carried light.

Although the contribution on the image is a function of the complete path, computer graphics applications usually assign estimated importance to individual steps of this path, which might be quite inaccurate. In a single step the importance is usually selected according to the BRDF [Dutre93, Lafor93], or according to the direction of the direct lightsources [Shirl96]. Combined methods that find the important directions using both the BRDF and the incident illumination have been proposed in [Veach94, Lafor96]. Just recently, Veach and Guibas [Veach97] proposed the *Metropolis method* [Metro53] to be used in the solution of the rendering equation. Unlike other approaches, Metropolis sampling can assign importance to a complete walk not just to the steps of this walk, and it explores important regions of the domain adaptively while running the algorithm. Thus no a-priori knowledge is required about the important rays to construct a probability density function in advance. Instead, the algorithm converges to this probability density automatically.

1.1 Metropolis sampling

The Metropolis algorithm [Metro53] converges to the optimal probability density that is proportional to the importance, that is in the limiting case $\mathcal{I}(\mathbf{z}) = b \cdot p(\mathbf{z})$.

However, this probability density cannot be stored, thus in the Monte-Carlo formula the importance should be used instead, in the following way:

$$I = \int_V \frac{f(\mathbf{z})}{\mathcal{I}(\mathbf{z})} \cdot \mathcal{I}(\mathbf{z})\; d\mathbf{z} = b \cdot \int_V \frac{f(\mathbf{z})}{\mathcal{I}(\mathbf{z})} \cdot p(\mathbf{z})\; d\mathbf{z} =$$

$$b \cdot E\left[\frac{f(\mathbf{z})}{\mathcal{I}(\mathbf{z})}\right] \approx \frac{b}{M} \cdot \sum_{i=1}^{M} \frac{f(\mathbf{z}_i)}{\mathcal{I}(\mathbf{z}_i)} \qquad (2)$$

In order to generate samples according to $p(\mathbf{z}) = 1/b \cdot \mathcal{I}(\mathbf{z})$, a Markovian process is constructed whose stationary distribution is just $p(\mathbf{z})$. The next state $\mathbf{z}_{i+1}$ of this process is found by letting an almost arbitrary *tentative transition function* $T(\mathbf{z}_i \to \mathbf{z}_t)$ generate a *tentative sample* $\mathbf{z}_t$ which is either accepted as the real next state or rejected making the next state equal to the actual state. The decision uses the "*acceptance probability*" $a(\mathbf{z}_i \to \mathbf{z}_t)$ that expresses the increase of the importance (if this "acceptance probability" is greater that 1, then the sample is accepted deterministically). The formal definition of this Markovian process $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_i, \ldots\}$ is as follows:

for $i = 1$ **to** M **do**
  Based on $\mathbf{z}_i$, choose a tentative point $\mathbf{z}_t$
    using $T(\mathbf{z}_i \to \mathbf{z}_t))$
  $a(\mathbf{z}_i \to \mathbf{z}_t) = \frac{\mathcal{I}(\mathbf{z}_t)\cdot T(\mathbf{z}_t \to \mathbf{z}_i)}{(\mathcal{I}(\mathbf{z}_i)\cdot T(\mathbf{z}_i \to \mathbf{z}_t)}$
  if $a(\mathbf{z}_i \to \mathbf{z}_t) \geq 1$ **then** $(\mathbf{z}_{i+1} = \mathbf{z}_t)$
  else // *accept with probability* $a(\mathbf{z}_i \to \mathbf{z}_t)$
    Generate random number r in $[0, 1]$.
    if $r < a(\mathbf{z}_i \to \mathbf{z}_t)$ **then** $\mathbf{z}_{i+1} = \mathbf{z}_t$
    else $\mathbf{z}_{i+1} = \mathbf{z}_i$
  endif
endfor

1.2 Start-up bias problem

The Metropolis sampling adaptively converges to the desired probability distribution. It also means that at the beginning of the process, the samples are not selected with the required probabilities, which introduces some error in the estimation. This error is generally called as the *start-up bias*. The main objective of this paper is to analyse the start-up bias problem both theoretically and using simulations.

2 Theoretical evaluation of the start-up bias

In order to theoretically evaluate the start-up bias, let us examine a simplified, 1-dimensional case when the importance is constant, thus the transition proposed by the tentative transition function is always accepted.

In this case, the probability density in the equilibrium is constant. The question is how quickly the Metropolis method approaches to this constant density (figure 1).

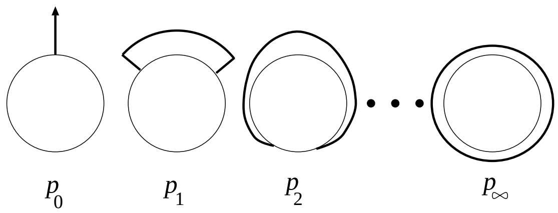

Figure 1: Evaluation of the uniform distribution

Metropolis method can generate samples following a given probability density in a closed interval. Since random mutations may result in points that are outside the closed interval, the boundaries should be handled in a special way.

If the variable of an integrand denotes "angle of direction", then the interval can be assumed to be "circular", that is, the external points close to one boundary are equivalent to the internal points of the other boundary. Using this assumption, let us suppose that the domain of the integration is $[-\pi, \pi]$ and the integrand is periodic with 2π.

Let the probability distribution at step n be p_n. The Metropolis method is initiated from a single seed at 0, thus $p_0 = \delta(x)$. Assume that transition probability $P(y \to x)$, which is equal to the tentative transition probability for constant importance, is homogeneous, that is $P(y \to x) = P(x - y)$. Using the total probability theorem, the following recursion can be established for the sequence of p_n:

$$p_{n+1}(x) = \int_{-\infty}^{\infty} p_n(y) \cdot P(x \to y)\ dy =$$

$$\int_{-\infty}^{\infty} p_n(y) \cdot P(x - y)\ dy = p_n * P, \quad (3)$$

where $*$ denotes the convolution operation.

Applying Fourier transformation to this iteration formula, we can obtain:

$$p^*_{n+1} = p^*_n \cdot P^*, \quad (4)$$

where $p^*_{n+1} = \mathcal{F}p_{n+1}$, $p^*_n = \mathcal{F}p_n$ and $P^* = \mathcal{F}P$.

Since the domain is "circular", i.e. x denotes the sample point as $x + 2k\pi$ for any integer k, the probability density is periodic, thus it can be obtained as a Fourier series:

$$p_n(x) = \sum_{k=-\infty}^{\infty} a_k^{(n)} e^{\jmath kx}, \quad (5)$$

where $\jmath = \sqrt{-1}$. The Fourier transform is thus a discrete spectrum:

$$p^*_n(f) = \sum_{k=-\infty}^{\infty} a_k^{(n)} \cdot \delta(f - k) \quad (6)$$

Substituting this into equation (4), we get

$$p^*_{n+1}(f) = \left(\sum_{k=-\infty}^{\infty} a_k^{(n)} \cdot \delta(f - k) \right) \cdot P^*(f) =$$

$$\sum_{k=-\infty}^{\infty} a_k^{(n)} \cdot P^*(k) \cdot \delta(f - k), \quad (7)$$

thus $a_k^{(n+1)} = a_k^{(n)} \cdot P^*(k)$.

Using the same concept n times, and taking into account that the initial distribution is $\delta(x)$, we can obtain:

$$p^*_n(f) = \sum_{k=-\infty}^{\infty} (P^*(k))^n \cdot \delta(f - k) \quad (8)$$

thus in the original domain

$$p_n(x) = \sum_{k=-\infty}^{\infty} (P^*(k))^n \cdot e^{\jmath kx} \quad (9)$$

The L_2 error between p_n and the stationary distribution is then

$$||p_n - p_\infty||_2 = \sqrt{\int_0^1 |p_n(x) - a_0^{(\infty)}|^2\ dx} \quad (10)$$

Note that according to the definition of the Fourier series

$$a_0^{(n)} = \frac{1}{2\pi} \cdot \int_{-\pi}^{\pi} p_n(x)\ dx = 1 \quad (11)$$

independently of n, thus $a_0^{(\infty)}$ is also 1. Using this and substituting equation (9) in equation (10), we get the following error for the distribution:

$$||p_n - p_\infty||_2 = \sqrt{\sum_{k=-\infty, k\neq 0}^{\infty} |P^*(k)|^{2n}} \quad (12)$$

2.1 Starting from multiple seeds

So far we have assumed that the integrand is estimated from a single random walk governed by the Markovian process. One way of reducing the start-up bias is to use several walks initiated from different starting points, called seeds, and combine their results.

If the initial point is generated from seed-points $x_1, x_2, \ldots, x_N$ randomly selecting x_i with probability α_i ($\sum_{i=1}^{N} \alpha_i = 1$), then the initial probability distribution is the following

$$p_0(x) = \sum_{i=1}^{N} \alpha_i \cdot \delta(x - x_i). \qquad (13)$$

Using the same concept as before, the probability density after n steps can be obtained in the following form

$$p_n(x) = \sum_{k=-\infty}^{\infty} \sum_{i=1}^{N} \alpha_i \cdot (P^*(k))^n \cdot e^{\jmath k(x-x_i)}. \qquad (14)$$

The error of the probability distribution after step n is then

$$\sqrt{\sum_{k=-\infty, k\neq 0}^{\infty} \left| P^*(k) \cdot \sum_{i=1}^{N} \alpha_i \cdot e^{\jmath k(x-x_i)} \right|^{2n}}. \qquad (15)$$

2.2 Analysis of uniform random perturbations

Let the perturbation be the selection of a point following uniform distribution from an interval of size Δ centered by the current point. Formally the transition probability is

$$P(x \to y) = \begin{cases} 1/\Delta & \text{if } |x-y| < \Delta, \\ 0 & \text{otherwise.} \end{cases} \qquad (16)$$

The Fourier transform of this function is

$$P^*(k) = \frac{\sin k\pi\Delta}{k\pi\Delta} \qquad (17)$$

which can be rather big even for large k values. This formula, together with equation (12) allows to generate the graph of the start-up errors for different sample numbers and for different perturbation size (figure 2). Note that the probability density is not accurate for many iterations if the perturbation size is small compared to the size of the domain. This situation gets just worse for higher dimensions.

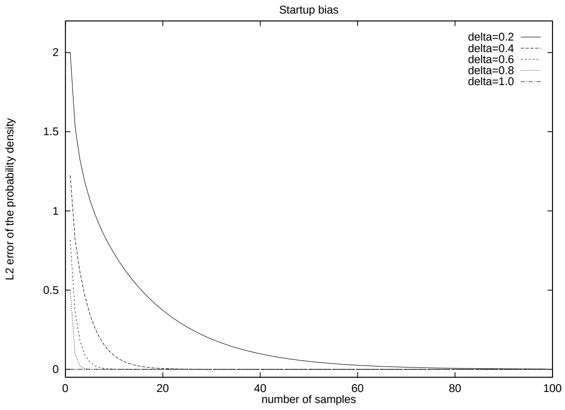

Figure 2: Start-up error for different perturbation size Δ

2.3 Selection of an optimal perturbation size

In order to find the extent of the random perturbation, several, contradicting requirements must be taken into consideration.

First of all, in order to cover the whole statespace of unit size, the number of mutations should be much greater than the ratio of the total integration domain and the size of the neighbourhood of the perturbation. This states that the mutations cannot be very small. Small mutations also emphasise the start-up bias problem which is a consequence of the fact that the Markovian process only converges to the desired probability density. On the other hand, if the mutations are large, then the Markovian process “forgets” which regions are important, thus the quality of importance sampling will decrease. Finally, another argument against small mutations is that it makes the subsequent samples strongly correlated. Note that Monte-Carlo quadrature rules usually assume that the random samples are statistically independent, which guarantees that if the variance of random variable $f(\mathbf{z})$ is σ, then the variance of the Monte-Carlo quadrature will be

$\sigma/\sqrt{M}$ after evaluating M samples. Since Metropolis method uses statistically correlated samples, the variance of the quadrature can be determined using the Bernstein theorem [Rényi62], which states that the variance of the quadrature is

$$\sigma \cdot \sqrt{\frac{1 + 2\sum_{k=1}^{M} R(k)}{M}} \qquad (18)$$

where $R(k)$ is an upperbund of the correlation between $f(\mathbf{z}_i)$ and $f(\mathbf{z}_{i+k})$. It means that strong correlation also increases the variance of the integral estimate.

3 Experimental evaluation of the start-up bias

In order to study the effectiveness of Metropolis sampling for realistic problems, two algorithms are taken into consideration. The first is *bi-directional path tracing*, for which Veach and Guibas have proposed the application of Metropolis sampling in [Veach97]. The second is *ray-bundle tracing* [SlllK98d]. The main difference of the two methods is that the integrand of the second method is smoother (in fact, of finite variation) but more expensive computationally.

3.1 Metropolis sampling in bi-directional path tracing

Bi-directional path tracing [Lafor93, Veach95] is based on the combination of shooting and gathering walks.

Walks are initiated at the same time from a selected light source and from the viewpoint. After some steps, either a single deterministic shadow ray is used to connect the two types of walks [Veach95], or all points of the gathering walk are connected to all points of the shooting walk using deterministic rays [Lafor93]. If the deterministic shadow ray detects that the two points are occluded from each other, then the contribution of this path is zero.

The transformation of the combined walk to a single gathering walk requires a multiplication by $\cos\theta'_k \cdot \cos\theta_{n-k+1}/r_k^2$, where θ'_k is the angle of the direction of the deterministic step and the normal of the last point visited by the gathering walk, $\cos\theta_{n-k+1}$ is the angle of the direction of the deterministic step and the normal of the last point visited by the shooting walk, and r_k is the length of the connection ray. Figure 3 shows an example when $k = 2$ and $n = 4$.

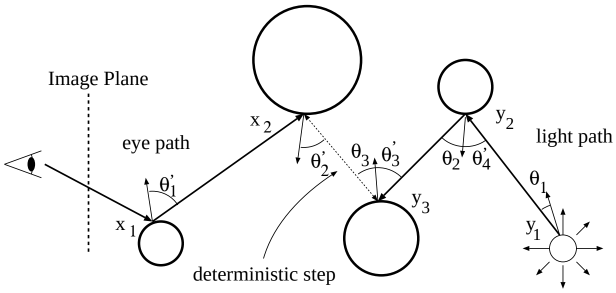

Figure 3: Bi-directional path tracing

3.1.1 Mutation strategies of bi-directional path tracing

In order to generate the next tentative sample from the actual gathering and shooting walk, Veach and Guibas [Veach97] have proposed several different alternatives, from which we consider only angular perturbation of the direction of the rays in the walk.

3.2 Evaluation of the performance of the Metropolis method for bi-directional path tracing

The performance of the Metropolis sampling has been measured using the test scene of figure 4. The apex angle of the perturbation has been set from 0.05 to 0.6 radians to evaluate the effect of the size of perturbation. The error curves are shown in figure 4, while the respective noisy images are in figure 5. The measurement results justify the theoretical considerations. If the perturbation size is too small, then the image tends to have incorrect bright and dark spots caused by the start-up bias.

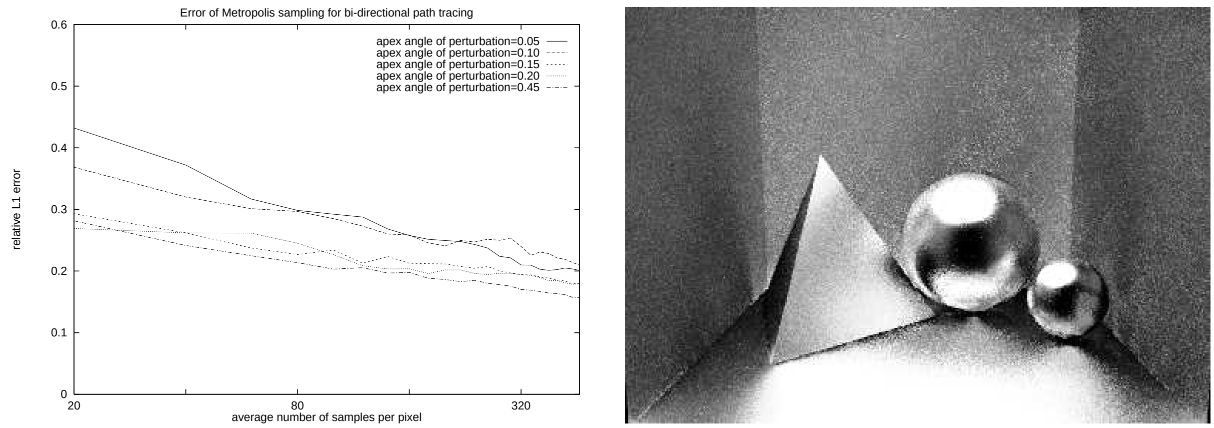

Figure 4: Error as a function of the average number of samples per pixel for different perturbation size, and the reference image

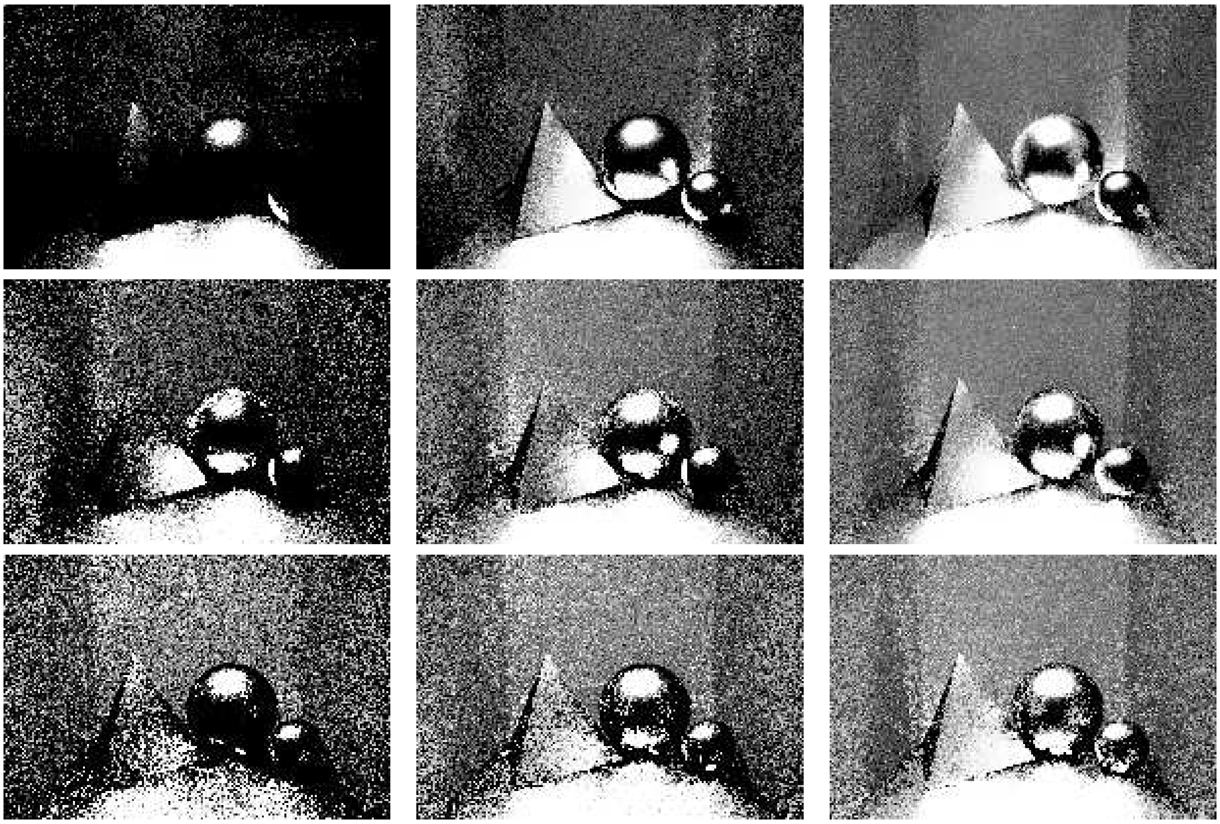

Figure 5: The images of the test scene computed by 20 samples per pixel, 80 samples per pixel and 320 samples per pixel in the left, middle and right columns, and using perturbation sizes 0.05, 0.15 and 0.45 in the top, middle and bottom rows, respectively.

3.3 Global ray-bundle tracing

Realising that an accurate solution of the rendering equation requires great many samples, *global ray-bundle tracing* [SllK98c, SllK98d, SllK98a] uses a bundle of very many (e.g. 1 million or even infinite) global parallel rays, which can be traced simultaneously using image coherence techniques. In order to represent the radiance that is transferred by a ray-bundle, finite-element techniques are applied that approximate the positional (but not the directional) dependence of the radiance by piece-wise continuous or piece-wise linear functions [SllK98b]. Substituting this into the rendering equation and projecting that into an adjoint base, we obtain:

$$\mathbf{L}(\omega) = \mathbf{L}^e(\omega) + \int_{\Omega} \mathbf{F}(\omega', \omega) \cdot \mathbf{A}(\omega') \cdot \mathbf{L}(\omega')\, d\omega', \tag{19}$$

where $\mathbf{L}(\omega)$ is the vector of radiance values, $\mathbf{F}(\omega', \omega)$ is a diagonal matrix of BRDFs, and *geometry matrix* $\mathbf{A}$ contains the relative visible areas: $\mathbf{A}(\omega')|_{ij} = A(i, j, \omega')/A_i$.

The visible area $A(i, j, \omega')$ expresses the projected area of patch j that is visible from patch i in direction ω'.

This is also an integral equation, but unlike the original rendering equation it provides the radiance of not only a single point but for all points at once. This integral equation is solved by random or quasi-random shooting type walks.

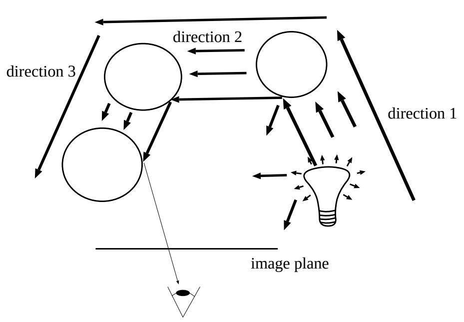

Figure 6: A path of ray-bundles

A single walk starts by selecting a direction either randomly or quasi-randomly, and the emission transfer of all patches is calculated into this direction (figure 6). Then a new direction is found, and the emission is transferred and the incoming radiance generated by the previous transfer is reflected from all patches into this new direction. The algorithm keeps doing this for a few times depending on how many bounces should be considered, then the emission is sent and the incoming radiance caused by the last transfer is reflected towards the eye. Averaging these contributions results in the final image.

3.3.1 Mutation strategies for ray-bundle tracing

The statespace of the Markovian process consists of D-dimensional vectors of directions that define the sequence of directions in the global walks. Thus the tentative transition function is allowed to modify one or more directions in these sequences.

3.3.2 Evaluation of the performance of the Metropolis method for ray-bundle tracing

To evaluate the efficiency of the Metropolis sampling for ray-bundle tracing, a scene having difficult lighting has been selected. The surfaces have both diffuse and specular reflection and the lightsource is well hidden from the camera (figure 7).

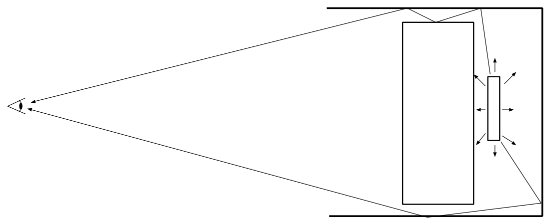

Figure 7: A test scene with difficult lighting conditions

The error measurements of the Metropolis method with different perturbation size, and for Monte-Carlo and quasi-Monte Carlo versions are shown by figure 8.

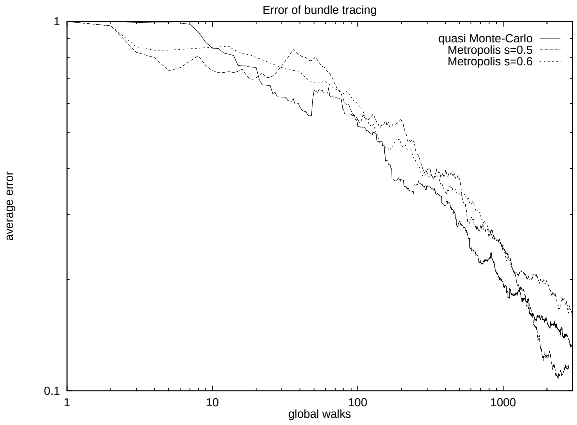

Figure 8: Error measurements for the "difficult scene"

4 Conclusions

This paper analysed the performance of the Metropolis method focusing on the start-up bias problem. The theoretical arguments came to the conclusion that for relatively homogeneous scenes the size of perturbation should be quite large to avoid the unacceptable start-up bias error. For ray-bundle tracing, where the integrand is of finite variation and the number of samples is quite low, the advantages of Metropolis sampling is even smaller. For so few samples the Metropolis method suffers from the initial bias problem. Due to the smooth integrand, the drawback of the initial bias is not compensated by the importance sampling.

Future research should concentrate on more general theoretical analysis of the start-up bias problem. The generalisation to multi-dimensional integrals is quite straightforward. In order to take into account more general integrands, a Laplace transform method is considered on an exponential integrand. The ultimate goal is to provide estimates for an "optimal" perturbation size.

REFERENCES

[Dutre93] P. Dutre, E. Lafortune, and Y. D. Willems. Monte Carlo light tracing with direct computation of pixel intensities. In *Compugraphics '93*, pages 128–137, Alvor, 1993.

[Lafor93] E. Lafortune and Y. D. Willems. Bi-directional path-tracing. In *Compugraphics '93*, pages 145–153, Alvor, 1993.

[Lafor96] E. Lafortune and Y. D. Willems. A 5D tree to reduce the variance of Monte Carlo ray tracing. In *Rendering Techniques '96*, pages 11–19, 1996.

[Metro53] N. Metropolis, A. Rosenbluth, M. Rosenbluth, A. Teller, and E. Teller. Equations of state calculations by fast computing machines. *Journal of Chemical Physics*, 21:1087–1091, 1953.

[Rényi62] Alfréd Rényi. *Wahrscheinlichkeitsrechnung*. VEB Deutscher Verlag der Wissenschaften, Berlin, 1962.

[Shirl96] P. Shirley, C. Wang, and K. Zimmerman. Monte Carlo techniques for direct lighting calculations. *ACM Transactions on Graphics*, 15(1):1–36, 1996.

[SlllK98a] L. Szirmay-Kalos. Global ray-bundle tracing. Technical Report TR-186-2-98-18, Institute of Computer Graphics, Vienna University of Technology, 1998. www.cg.tuwien.ac.at/.

[SlllK98b] L. Szirmay-Kalos, T. Fóris, and W. Purgathofer. Non-diffuse, random-walk radiosity algorithm with linear basis functions. *Machine Graphics and Vision*, 7(1):475–484, 1998.

[SlllK98c] L. Szirmay-Kalos, T. Fóris, and W. Purgathofer. Quasi-Monte Carlo global ray-bundle tracing with infinite number of rays. In *Winter School of Computer Graphics '98*, pages 386–393, Plzen, Czech Republic, 1998.

[SlllK98d] L. Szirmay-Kalos and W. Purgathofer. Global ray-bundle tracing with hardware acceleration. In *Rendering Techniques '98*, pages 247–258, 1998.

[Sobol91] I. Sobol. *Die Monte-Carlo Methode*. Deutscher Verlag der Wissenschaften, 1991.

[Veach94] E. Veach and L. Guibas. Optimally combining sampling techniques for Monte Carlo rendering. In *Rendering Techniques '94*, pages 147–162, 1994.

[Veach95] E. Veach and L. Guibas. Bidirectional estimators for light transport. In *Computer Graphics (SIGGRAPH '95 Proceedings)*, pages 419–428, 1995.

[Veach97] E. Veach and L. Guibas. Metropolis light transport. *Computer Graphics (SIGGRAPH '97 Proceedings)*, pages 65–76, 1997.