AN EMPIRICAL COMPARISON OF MONTE CARLO RADIOSITY ALGORITHMS

Philippe Bekaert, Ronald Cools, Yves D. Willems

Department of Computer Science, Katholieke Universiteit Leuven
Celestijnenlaan 200 A, B-3001 Heverlee, Belgium
e-mail: Philippe.Bekaert@cs.kuleuven.ac.be

ABSTRACT

Monte Carlo radiosity algorithms are radiosity algorithms in which the radiosity integral equation or system of linear equations is solved using Monte Carlo random walk techniques. Since explicit form factor computation and storage is completely avoided in Monte Carlo radiosity algorithms, these algorithms are more reliable and require significantly less storage than other radiosity algorithms, making it feasible to render much more complex scenes. This paper presents a comparative study of four main aspects in which proposed Monte Carlo radiosity algorithms differ: whether the discrete or continuous equation is being solved, the random walk state transition simulation technique, sampling order and the sample number generator.

Keywords: Global Illumination, Radiosity, Quasi Monte Carlo, Random Walk

1 Monte Carlo Radiosity

This section gives an overview of Monte Carlo radiosity (MCR) algorithms and clarifies our standpoint that all MCR algorithms are random walk algorithms. A description of the experiments is given next in §2. The results of our experiments are presented and discussed in §3 to §6.

1.1 Continuous versus discrete and why Monte Carlo?

Radiosity algorithms compute and store a view-independent representation of the illumination in a diffuse (Lambertian) environment without transparency or participating media. Light transport in such environments is described by the following integral equation, which is a special case of the general rendering equation [8]:

$$B(x) = B^e(x) + \rho(x) \int_A G(x,y) B(y) dA_y \quad (1)$$

with integral kernel

$$G(x,y) = \frac{\cos\theta_x \cos\theta_y}{\pi r_{xy}^2} \mathrm{vis}(x,y) \quad (2)$$

where

- $B(x)$ denotes the total radiosity $[W/m^2]$ at a point x on the surfaces A of the scene;
- $B^e(x)$ denotes the self-emitted radiosity at x (non-zero if x is a point on a light source only);
- $\rho(x)$ denotes the reflectivity (dimension-less number $0 < \rho(x) < 1$) of the surface at x;
- dA_y denotes a differential surface area around a point y;
- θ_x and θ_y are the angles between the line connecting points x and y and the surface normal at x and y respectively;
- r_{xy} is the distance between the points x and y;
- $\mathrm{vis}(x,y)$ is 1 if x and y are mutually visible and 0 if occluded from each other.

We shall refer to equation (1) as the *continuous radiosity equation*. The solution of this equation is a function of location x on the surfaces A in the scene. Since it can in general not be solved analytically (one exception is used in our experiments, see §2), numerical methods are used to approximately compute the solution of (1).

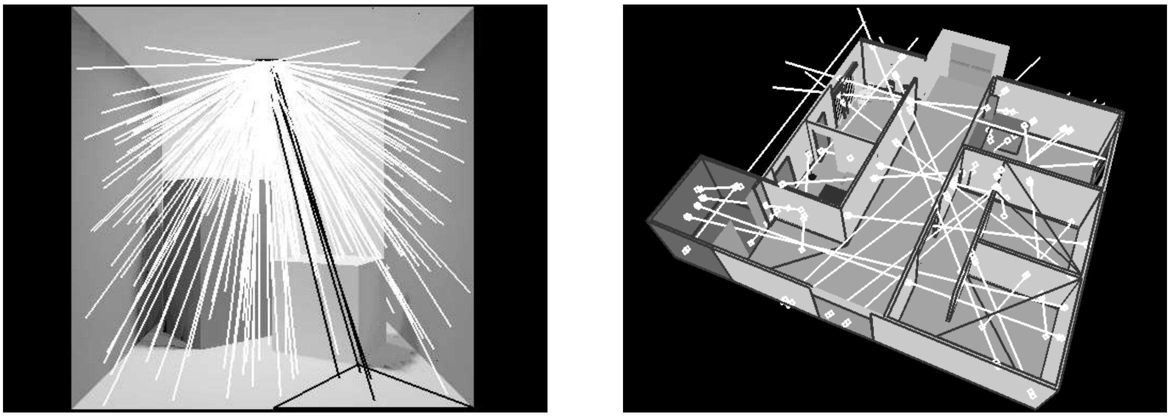

Figure 1: Local (left) and global (right) sample line generation. The left image shows 200 "local" lines from a light source. Four of these, indicated in black, hit a given patch in the floor plane. The form factor between the light source and the patch thus is approximately 4/200. In the right image, a number of "global" lines are shown. These lines are generated irrespective of the patches in the scene. Their intersection points with the surfaces in the scene form spans along which particles can be exchanged.

In the radiosity method, a piecewise low-order polynomial approximation, most often a piecewise constant approximation, is sought for the solution. Some algorithms [14, 9, 20] compute such an approximate solution directly using equation 1. We shall refer to these algorithms as *continuous radiosity algorithms*, because they avoid kernel discretisation during the computations.

In most radiosity algorithms, equation (1) is discretised into a system of linear equations, by using the Nystrom- or quadrature method, point collocation or the Galerkin method. The surfaces of the scene are discretised into patches i, with surface area A_i. When a constant approximation B_i for the radiosity $B(x)$ on each patch i is sought, the Galerkin form of weighted residuals yields the following system of linear equations:

$$B_i = B_i^e + \rho_i \sum_j F_{ij} B_j \qquad (3)$$

or equivalently

$$P_i = P_i^e + \rho_i \sum_j P_j F_{ji} \qquad (4)$$

with form factors

$$F_{ij} = \frac{1}{A_i} \int_{A_i} \int_{A_j} G(x,y) dA_y dA_x. \qquad (5)$$

B_i^e and ρ_i are the average self-emitted radiosity and reflectivity of patch i. P_i and P_i^e are the total and self-emitted power ($P_i = A_i B_i$) of the patches. The sum is over all patches in the environment. Similar sets of linear equations result for non-constant approximations or other discretisation methods. We shall refer to algorithms that solve a discretised radiosity equation as *discrete radiosity algorithms*.

Traditional full matrix radiosity [6, 13, 4], progressive refinement radiosity [3], and also hierarchical radiosity [7], basically first compute the form factors and next solve the system of linear equations using an iterative method such as Jacobi or Southwell iterations[1]. The main problems of these approaches however concern reliability and storage requirements: the form factors (5) require the solution of a difficult integral, and their number is $\mathcal{O}(n^2)$ where n is the number of patches. In Monte Carlo radiosity [18, 5, 15, 10, 12, 11, 17, 16], explicit form factor computation and storage is completely avoided by using Monte Carlo random walk simulations instead of an iterative solution method in order to solve the radiosity equations. Because of this, Monte Carlo radiosity algorithms are more reliable and require significantly less storage than traditional radiosity algorithms, making them suited to render much more complex environments.

1.2 Random walk state transition simulation

All MCR algorithms basically carry out a number of random walk simulations: particles are generated on a light source and their path in the environment is simulated until they get absorbed. The score of the random walks on each patch, yields an approximation of the power P_i. The random walk transition probabilities, that is: the probabilities that a state (patch) j will be visited next when currently visiting state i, are given by the form factors F_{ij}. Fortunately, there exist constructions that do not require explicit form

[1] In practice, the steps of form factor computation and system solution are intertwined as to yield feedback to the user as soon as possible.

factor knowledge in order to simulate such a state transition.

A first construction is based on Nusselt's analogy [4]: a ray is generated with starting point uniformly chosen over the surface area of patch i and with direction chosen such that the angle w.r.t. the surface normal is cosine distributed. The probability that such a ray has nearest intersection with the surfaces in the scene on a second patch j, is given by the form factor F_{ij} (see figure 1). When given a particle on patch i, the choice what patch the particle will visit next can be made by explicitly constructing and tracing such a ray. We shall refer to this construction as *local line* generation.

Mateu Sbert [15, 17] has introduced alternative state transition simulation techniques, which are named *global line* generation techniques. In these techniques, lines are generated irrespective of patches in the scene, e.g. by connecting two uniformly distributed points on a sphere bounding the scene (see figure 1). The intersection points of such a line with the surfaces in the scene have uniform distributed location on the hit patches. Their angles w.r.t. the surface normals are cosine distributed as well. The intersection points form mutually visible pairs, defining spans on the global line along which random walk state transitions can be simulated.

1.3 Sampling order

The state transitions of the random walks correspond to local or global sample lines. Because each random walk has a certain probability of being terminated at each hit patch and also never more than one state transition from a given state (patch) is considered, the set of sample lines can be envisaged as a non-branching tree structure. The root of the tree structure corresponds with the light sources in the scene, where all random walks are initiated. The ratio of the number of branches of length k over the number of branches of length $k-1$ is about equal to the average reflectivity of the surfaces in the scene[2].

Each Monte Carlo radiosity algorithm can be viewed as enumerating the sample lines in the tree structure in some order (see figure 2). Pattanaik's continuous and Sbert's discrete particle tracing [14, 16] are *depth-first* sampling order algorithms: the sample lines corresponding to a first branch are enumerated in order before those of the next branch. Shirley's algorithm [18], a *breadth-first* sampling order algorithm, will first enumerate all sample lines from the light sources, at the root of the tree structure. Next all sample lines corresponding to the second branch segments are enumerated and so on. Other algorithms [5, 12, 11] use an intermediate sampling order strategy: in each iteration step, sample lines at different depths and belonging to different branches are generated.

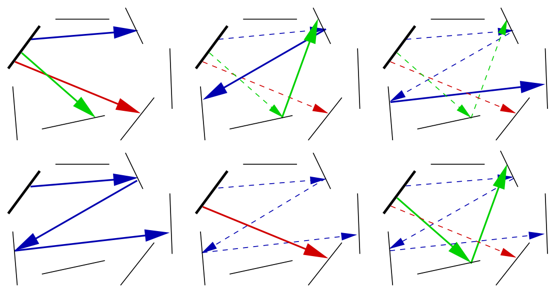

Figure 2: Sampling order: with breadth-first sampling (top), all path segments originating at the light source are generated first, next the path segments corresponding to first order indirect illumination and so on. With depth first sampling (bottom), random walks are fully generated until absorption before starting with the next random walk. The wide line top-left represents a light source.

1.4 Low discrepancy sample number generation

A final aspect with large influence on the convergence of MCR algorithms is the choice of the sample number generator. Regardless of whether local or global lines are used, a quadruple of sample numbers $\xi_{1\ldots4}$, most often in the range $[0,1]$, needs to be generated. These quadruples uniquely determine a local or global line. The most common way of generating such quadruples is to use random sample numbers. Keller [9] has introduced the use of *quasi-Monte Carlo* or *low discrepancy* sample numbers for radiosity computations. By giving up randomness in favour of better uniformity, a faster than $\mathcal{O}(1/\sqrt{N})$ convergence can be obtained in Monte Carlo integration. The theory predicts convergence rates up to $\mathcal{O}(\frac{1}{N})$, with N the number of samples [19].

2 Experiment description

In the remainder of this paper, four experiments are presented measuring the influence of each of the four MCR algorithm aspects explained above on convergence rate: sampling order (§3), local versus global sample line generation (§4), discrete versus continuous (§5), and finally the influence of the sample number generator (§6).

[2]This is only true when the reflectivities are chosen as random walk survival probabilities, which is however the case in all algorithms considered in this paper.

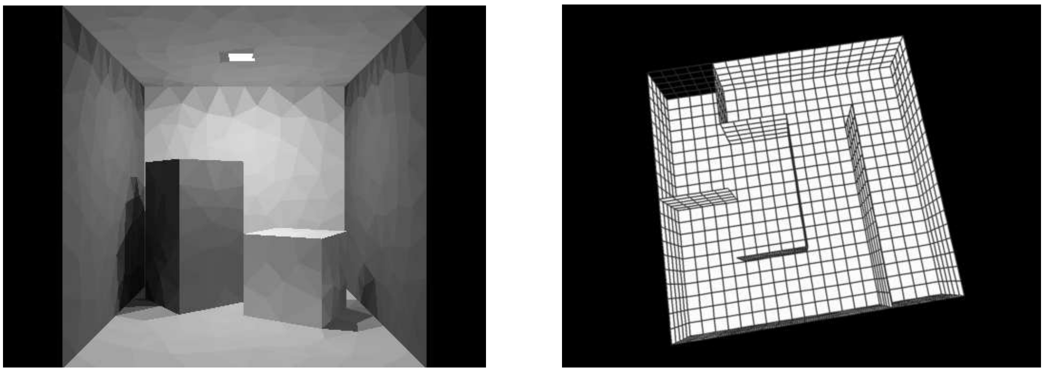

Figure 3: Test scenes used in the experiments: box (left) and labyrinth (right). The right image shows the reflectivities ρ_i of the patches in the high average reflectivity case $\rho_{av} \approx 0.9$. The self-emitted radiosities B_i^e were chosen such that $B_i^e + \rho_i = 1$. The radiosity solution B_i is constant and equal to one everywhere.

In each experiment, we have measured the RMS error in the result obtained with varying algorithmic choices as a function of the number of samples. If local line generation is used, each local line counts as one sample. In case global lines are used, each global line *span* counts as two samples because of bidirectionality. The number of samples is only a rough measure of computation cost as the cost per global line span is lower than the cost per local line. The cost ratio is bounded by the average number of spans per global line [15]. Studying error as a function of number of samples provides more insight in the algorithms.

The test scenes used are shown in figure 3. The labyrinth scene is a case in which the radiosity solution is analytically known to be constant and equal to one everywhere. This will always be the case regardless of geometry as long as for each individual patch $B_i^e + \rho_i = 1$. The degree of freedom in varying ρ_i per patch was used to create three labyrinth instances with different average reflectivity: 0.1, 0.5 and 0.9. Since the solution is known, the RMS error can be determined exactly. These tests exhibit better the theoretically expected behaviour of the algorithms.

Scenes with constant illumination are however extremely rare in practice. The second test scene, a simple box containing two other boxes, will provide better indications of the behaviour of the algorithms in practice. Since the solution is not analytically known, the error has to be estimated by comparing with a reference solution. References images were computed using about 100 times more samples than indicated in the graphs. The RMS image difference with the reference image was plotted as an indication for the error. Allthough this includes also the effect of tone mapping and only the parts of the scene that are visible in the images are taken into account, it has no qualitative implications for our conclusions.

3 Experiment 1: Sampling order

In the graphs in figure 4, the measured error of three local discrete MCR algorithms is plotted. The only difference between these algorithms is the sampling order. The graphs show results obtained with random numbers as well as with Halton low discrepancy numbers. The graphs suggest that the **sampling order has no influence on the long term convergence rate** of the algorithms, which however does depend on the sample number generator (see also below §6).

The sampling order has no influence on the long term convergence rate because approximately the same sample lines are generated in each algorithm: as the radiosity solution is approximated better, each algorithm yields approximately the same power for each patch. The algorithms use random walks with particles carrying the same amount of power, so, the number of particles per patch will be approximately equal for each algorithm. If the same sample numbers are used for generating the sample lines leaving each patch, the sample lines themselves will be the same as well. Only the order in which they are enumerated differs.

The **sampling order however has a clear influence on the initial behaviour**. In a breadth-first approach, the effect of k-th order interreflections is only computed after lower order illumination has been completely computed. In a depth-first approach, a complete solution, but with high variance, is obtained already after one single random walk. Feda [5] obtained a more progressive variant of Shirley's purely breadth-first algorithm [18] by breaking the computations into stages. Each stage still is breadth-first, but complete solutions are obtained and displayed after each stage. WDRS radiosity [11] takes a similar approach, improving on stochastic ray radiosity [12].

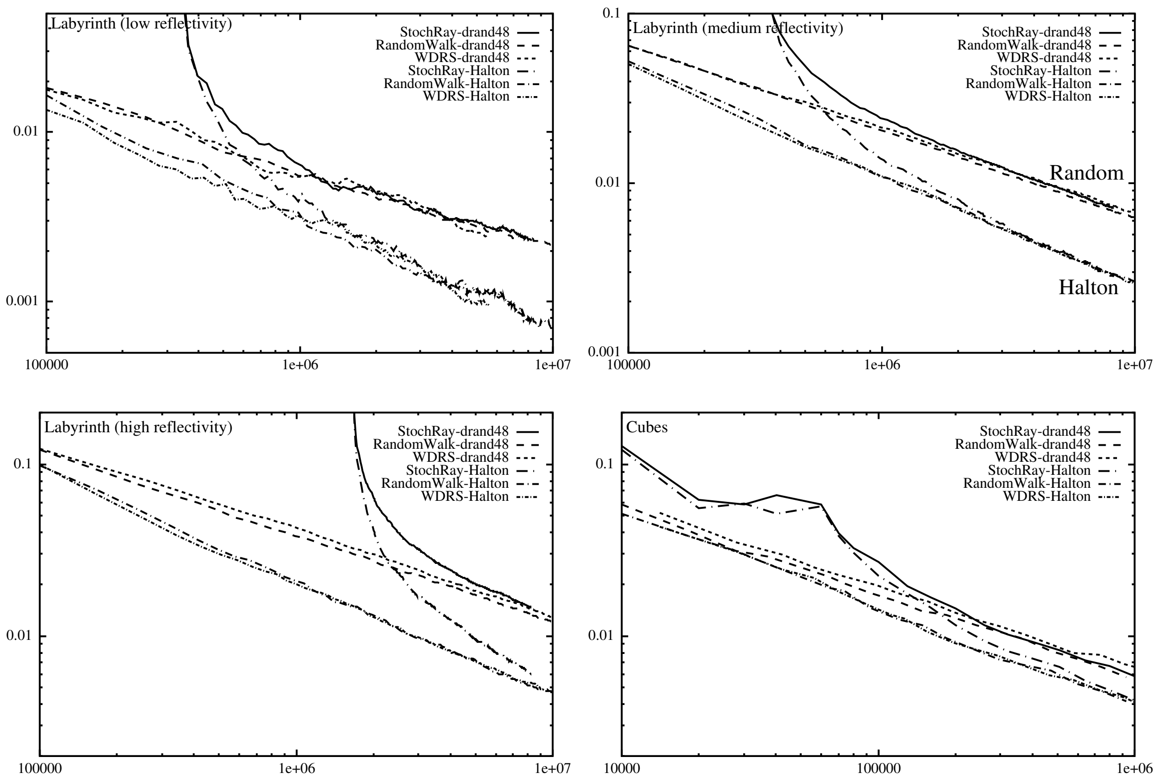

Figure 4: Empirical comparison of sampling order in local discrete Monte Carlo radiosity. The sampling order has no influence on the long-term convergence rate, which however depends strongly on the random number generator. Legend: ShochRay = [12], RandomWalk = [16], WDRS = [11]. Measured RMS error is plotted against the number of samples, see §2.

4 Experiment 2: Local versus global line generation

In the graphs of figure 5, the effect of local versus global sample line generation is compared for discrete particle tracing [17, 16]. The three labyrinth tests with random sampling suggest that the choice for **local or global line generation does not affect the convergence rate per se**. In these cases, each sample, a local line or global line span, will be used for equally large energy transfers. Since the cost per sample is lower with global lines, a global line technique will be preferred in these cases.

The "cubes" test however reveals a large difference because the amount of energy transferred by the global line spans may vary enormously corresponding to whether the line happens to hit the light source or not. Mateu Sbert proposed the use of a first, "smoothing", pass before using a global line pass to remedy this problem [17].

When Halton sampling is used instead of random sampling, a significant advantage for local line sampling is observed in the labyrinth tests as well. This may suggest that straightforward replacing the quadruple of random numbers by a 4D QMC sample for global line generation is sub-optimal. Improved global line distribution for light transport and optimal QMC global line construction are topics of ongoing research.

5 Experiment 3: Discrete versus continuous random walk

In the graphs of figure 6, the convergence rate of local discrete versus continuous particle tracing [14, 16] is compared with both random and Halton sample numbers. In an implementation, the difference between discrete and continuous particle tracing is very small: a particle hitting a patch is warped to another, uniformly chosen, location on the patch in discrete particle tracing, while it is scattered from the point of incidence in continuous particle tracing.

The convergence rate with random sampling is equal for discrete and continuous particle tracing: $\mathcal{O}(1/\sqrt{N})$ with N the number of samples. With QMC sampling however, the convergence rate of discrete particle tracing, and because of §3 also of other local discrete Monte Carlo radiosity algorithms, can be significantly better, up to $N^{-0.7}$ versus $N^{-0.6}$, than the convergence rate of continuous particle tracing. The difference appears to be larger for higher average reflectivity. This should not be surprising since direct illumination is computed identically in continuous and discrete particle tracing.

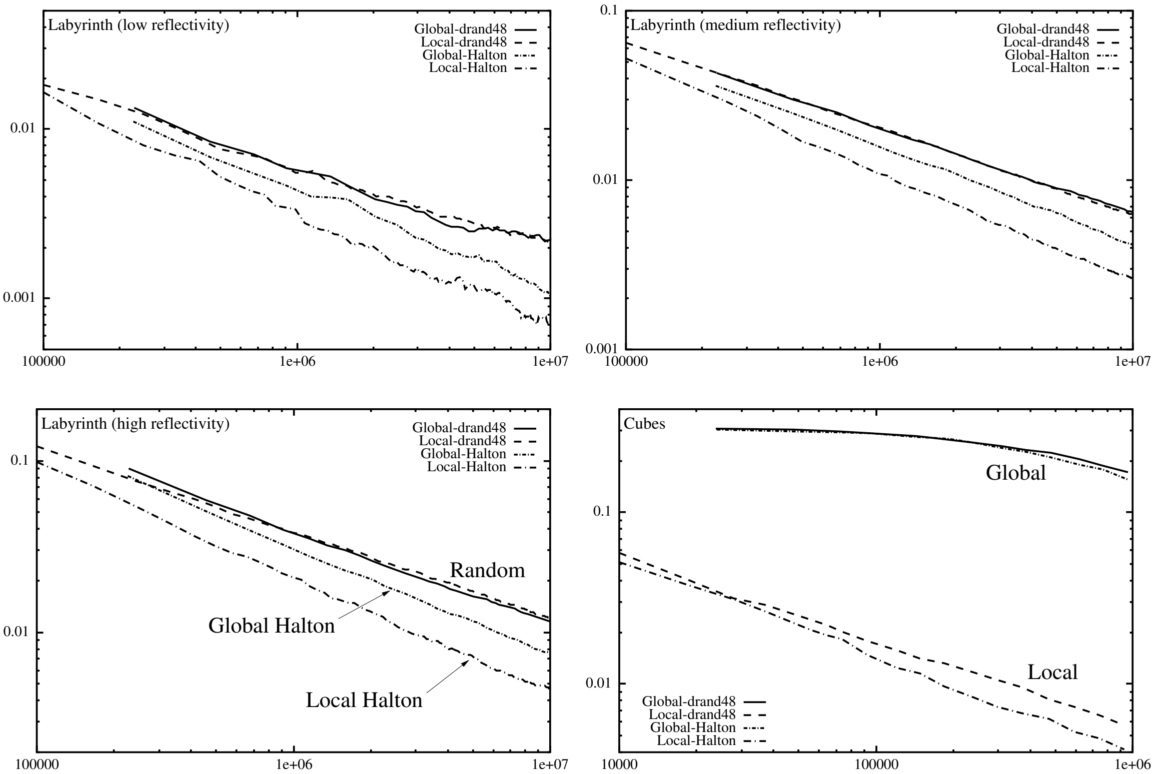

Figure 5: Empirical comparison of local versus global line generation in discrete Monte Carlo radiosity. Note that the cost per sample is lower for global lines than for local lines (not reflected in these graphs).

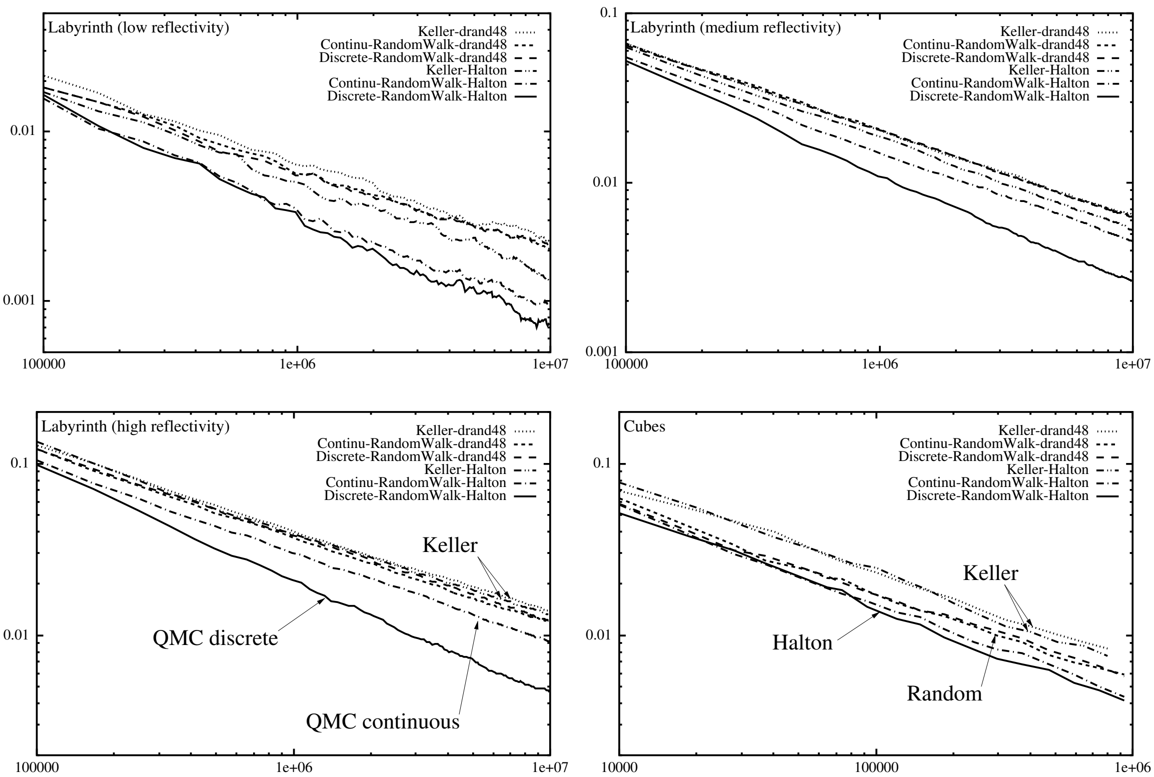

Figure 6: Empirical comparison of discrete versus continuous local particle tracing and Keller's QMC radiosity algorithm [9]. Discrete random walk can be significantly superior to a continuous random walk when QMC sampling is used.

The graphs however only show computational error and not the discretisation error, which is the error caused by approximating the true radiosity solution $B(x)$ by a piecewise constant function [1]. The discretisation error in discrete radiosity algorithms is given by $|\varepsilon(x)|$ where ε is the solution of an equation that is very similar to (1) [2]:

$$\varepsilon(x) = \delta(x) + \rho(x) \int_A G(x,y)\varepsilon(y) dA_y$$

with $\delta(x)$ given by

$$\rho(x) \sum_j B_j \int_{A_j} \left[G(x,y) - \frac{1}{A_i} \int_{A_i} G(z,y) dA_z \right] dA_y. \tag{6}$$

The discretisation error in a continuous algorithm is $|\tilde{\varepsilon}(x)|$ with

$$\tilde{\varepsilon}(x) = \tilde{\delta}(x)$$

where $\tilde{\delta}(x)$ is very similar to $\delta(x)$ (6):

$$\rho(x) \sum_j \int_{A_j} B(y) \left[G(x,y) - \frac{1}{A_i} \int_{A_i} G(z,y) dA_z \right] dA_y.$$

Discretisation error will be lower with continuous particle tracing than with discrete particle tracing mainly because there is no propagation of discretisation error in the former. Note however that the Galerkin discretisation method minimises exactly this kind of error.

We conclude that **with random sampling, the sum of computational and discretisation error will be higher with discrete algorithms than with continuous algorithms for the same amount of work. With QMC sampling however, lower computational error will often compensate higher discretisation error with discrete algorithms, so discrete algorithms may be preferred.**

The graphs in figure 6 also show the convergence rate obtained with Keller's algorithm [9] with both random and Halton sampling. The convergence rate obtained with this algorithm are inferior to those of [14, 16] mainly because the particle paths in [9] are fully determined by their sequence number without taking into account the local surface scattering properties at each visited patch. A QMC *integration* approach is followed in [9] rather than QMC random walk *simulation*.

6 Experiment 4: Sample number generator

In the previous experiments, we have always used the Halton number sequence to characterise QMC sampling. The graphs in figure 7 compare the convergence rate of local discrete random walk with 4D random, Halton, scrambled Halton, Faure, generalised Faure and Sobol and Niederreiter sample number sequences [19]. While a speed difference of up two a factor 2 may be observed with different QMC sequences, the graphs suggest that **no QMC sequence behaves systematically superior to the other sequences. The simple Halton sequence behaves well in all cases.**

Acknowledgements

We are very grateful to László Neumann and Mateu Sbert, for discussions that became the incentive for effectively carrying out the experiments in this paper. Mateu Sbert was the first to recognize that all MCR algorithms are random walk algorithms. Stijn Tastenhoye performed the comparison of QMC sequences which is presented in §6 as a part of his bachelor thesis at the K.U.Leuven. The first author also acknowledges financial support by a grant from the "Flemish Institute for the Promotion of Scientific-Technological Research in Industry" (IWT#941120).

REFERENCES

[1] J. Arvo, K. Torrance, and B. Smits. A framework for the analysis of error in global illumination algorithms. In *SIGGRAPH '94 Proceedings*, pages 75–84, July 1994.

[2] Ph. Bekaert and Y. D. Willems. Error control for radiosity. In *Eurographics Rendering Workshop 1996*, pages 153–164, June 1996.

[3] M. F. Cohen, S. E. Chen, J. R. Wallace, and D. P. Greenberg. A progressive refinement approach to fast radiosity image generation. In *Computer Graphics (SIGGRAPH '88 Proceedings)*, volume 22, pages 75–84, August 1988.

[4] M. F. Cohen and D. P. Greenberg. The Hemi-Cube: A radiosity solution for complex environments. In *Computer Graphics (SIGGRAPH '85 Proceedings)*, volume 19, pages 31–40, August 1985.

[5] M. Feda and W. Purgathofer. Progressive ray refinement for Monte Carlo radiosity. In *Fourth Eurographics Workshop on Rendering*, pages 15–26, June 1993.

[6] C. M. Goral, K. E. Torrance, D. P. Greenberg, and B. Battaile. Modelling the interaction of light between diffuse surfaces. *Computer Graphics (SIGGRAPH '84 Proceedings)*, 18(3):212–22, July 1984.

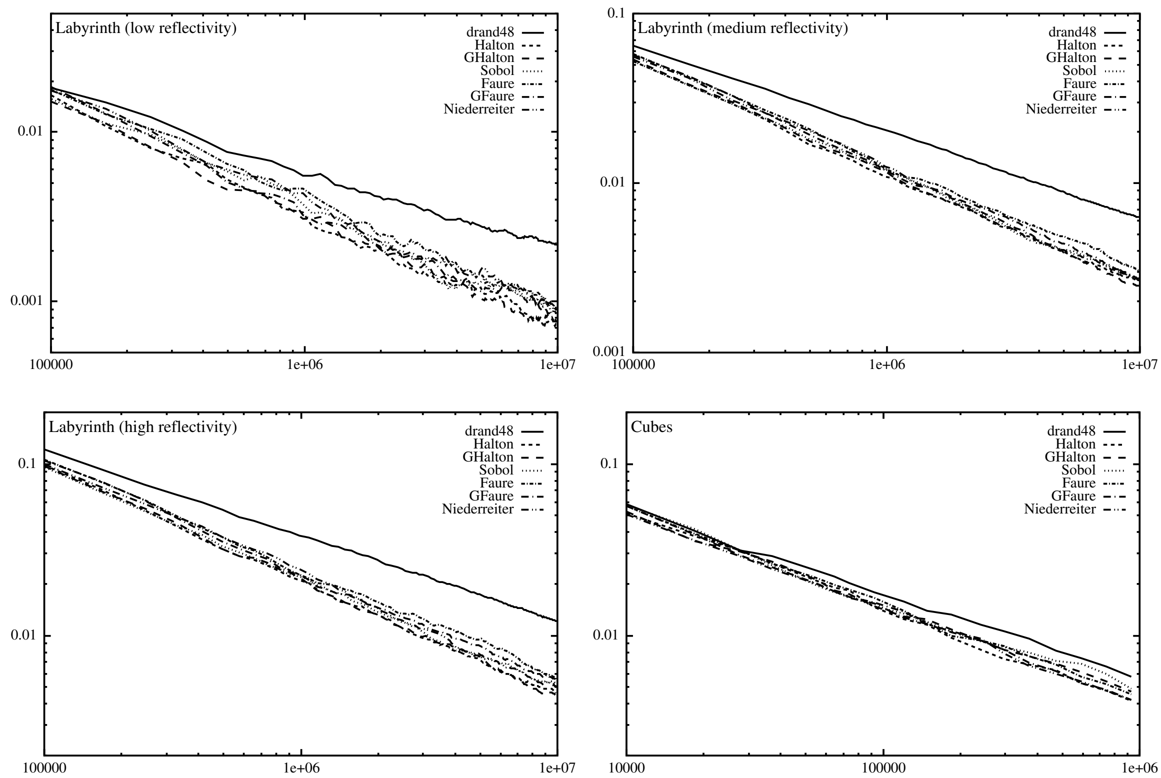

Figure 7: The influence of the QMC sample number sequence in discrete local random walk radiosity. No QMC sequence is systematically superior to the other sequences. The simple Halton sequence behaves well in all cases.

[7] P. Hanrahan, D. Salzman, and L. Aupperle. A rapid hierarchical radiosity algorithm. In *Computer Graphics (SIGGRAPH '91 Proceedings)*, volume 25, pages 197–206, July 1991.

[8] J. T. Kajiya. The rendering equation. In *Computer Graphics (SIGGRAPH '86 Proceedings)*, volume 20, pages 143–150, August 1986.

[9] A. Keller. Quasi-Monte Carlo radiosity. In *Eurographics Rendering Workshop 1996*, pages 101–110, June 1996.

[10] L. Neumann. Monte Carlo radiosity. *Computing*, 55(1):23–42, 1995.

[11] L. Neumann, A. Neumann, and Ph. Bekaert. Radiosity with well distributed ray sets. *Computer Graphics Forum*, 16(3), 1997.

[12] L. Neumann, W. Purgathofer, R. Tobler, A. Neumann, P. Elias, M. Feda, and X. Pueyo. The stochastic ray method for radiosity. In *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, July 1995.

[13] T. Nishita and E. Nakamae. Continuous tone representation of three-dimensional objects taking account of shadows and interreflection. In *Computer Graphics (SIGGRAPH '85 Proceedings)*, volume 19, pages 23–30, July 1985.

[14] S. Pattanaik and S. Mudur. Computation of global illumination by Monte Carlo simulation of the particle model of light. *Third Eurographics Workshop on Rendering*, pages 71–83, May 1992.

[15] M. Sbert. An integral geometry based method for fast form-factor computation. *Computer Graphics Forum*, 12(3):C409–C420, 1993.

[16] M. Sbert. Error and complexity of random walk Monte Carlo radiosity. *IEEE Transactions on Visualization and Computer Graphics*, 3(1):23–38, March 1997.

[17] M. Sbert, X. Pueyo, L. Neumann, and W. Purgathofer. Global multipath Monte Carlo algorithms for radiosity. *The Visual Computer*, 12(2):47–61, 1996.

[18] P. Shirley. Radiosity via ray tracing. In James Arvo, editor, *Graphics Gems II*, pages 306–310. Academic Press, San Diego, 1991.

[19] S. Tezuka. *Uniform random numbers: theory and practice.* Kluwer Academic Publishers, 1995.

[20] B. Walter, Ph. Hubbard, P. Shirley, and D. Greenberg. Global illumination using local linear density estimation. *ACM Transactions on Graphics*, 16(3):217–259, July 1997.