Parallel Monte Carlo Approach for Integration of the Rendering Equation*

Ivan T. Dimov[1], Anton A. Penzov[2], and Stanislava S. Stoilova[3]

[1] Institute for Parallel Processing, Bulgarian Academy of Sciences, Acad. G. Bonchev Str., bl. 25 A, 1113 Sofia, Bulgaria and ACET Centre, University of Reading, Whiteknights, PO Box 217, Reading, RG6 6AH, UK
I.T.Dimov@reading.ac.uk

[2] Institute for Parallel Processing, Bulgarian Academy of Sciences, Acad. G. Bonchev Str., bl. 25 A, 1113 Sofia, Bulgaria
apenzov@parallel.bas.bg

[3] Institute of Mathematics and Informatics, Bulgarian Academy of Sciences, Acad. G. Bonchev Str., bl. 8, 1113 Sofia, Bulgaria
stoilova@math.bas.bg

Abstract. This paper is addressed to the numerical solving of the rendering equation in realistic image creation. The rendering equation is integral equation describing the light propagation in a scene accordingly to a given illumination model. The used illumination model determines the kernel of the equation under consideration. Nowadays, widely used are the Monte Carlo methods for solving the rendering equation in order to create photorealistic images.

In this work we consider the Monte Carlo solving of the rendering equation in the context of the parallel sampling scheme for hemisphere. Our aim is to apply this sampling scheme to stratified Monte Carlo integration method for parallel solving of the rendering equation. The domain for integration of the rendering equation is a hemisphere. We divide the hemispherical domain into a number of equal sub-domains of orthogonal spherical triangles. This domain partitioning allows to solve the rendering equation in parallel. It is known that the Neumann series represent the solution of the integral equation as a infinity sum of integrals. We approximate this sum with a desired truncation error (systematic error) receiving the fixed number of iteration. Then the rendering equation is solved iteratively using Monte Carlo approach. At each iteration we solve multi-dimensional integrals using uniform hemisphere partitioning scheme. An estimate of the rate of convergence is obtained using the stratified Monte Carlo method.

This domain partitioning allows easy parallel realization and leads to convergence improvement of the Monte Carlo method. The high performance and Grid computing of the corresponding Monte Carlo scheme are discussed.

* Supported by the Ministry of Education and Science of Bulgaria under Grand No. I1405/04 and by FP6 INCO Grand 016639/2005 Project BIS-21++.

T. Boyanov et al. (Eds.): NMA 2006, LNCS 4310, pp. 140–147, 2007.

1 Introduction

The main task in the area of computer graphics is photorealistic image creation. From mathematical point of view, photorealistic image synthesis is equivalent to the solution of the rendering equation [9]. The rendering equation is a Fredholm type integral equation of second kind. It describes the light propagation in closed domains called scenes. The kernel of the rendering equation is determined by the used illumination model. The illumination model (see [11] for a survey of illumination models) describes the interaction of the light with a point on the surface in the scene. Each illumination model approximates the BRDF (bidirectional reflectance distribution function), taking into account the material surface characteristics. The physical properties like reflectivity, roughness, and colour of the surface material are characterized by the BRDF. This function describes the light reflection from a surface point as a ratio of outgoing to incoming light. It depends on the wavelength of the light, incoming, outgoing light directions and location of the reflection point. The BRDF expression receives various initial values for the objects with different material properties. Philip Dutré in [4] presents a good survey of the different BRDF models for realistic image synthesis.

One possible approach for the solution of rendering equation is the Monte Carlo methods, which has been in the focus of mathematical research for several decades. Frequently the Monte Carlo methods for numerical integration of the rendering equation are the only practical method for multi-dimensional integrals. The convergence rate of conventional Monte Carlo method is $O(N^{-\frac{1}{2}})$ which gives relatively slow performance at realistic image synthesis of complex scenes and physical phenomena simulation. In order to improve Monte Carlo method and speed up the computation much of the efforts are directed to the variance reduction techniques. The separation of the integration domain [12] is widely used Monte Carlo variance reduction method. Monte Carlo algorithms using importance separation of the integration domain are presented in [8], [7], [2], [5] and [6]. The method of importance separation uses a special partition of the domain and computes the given integral as a sum of the integrals on the sub-domains. An adaptive sub-division technique of spherical triangle domains is proposed by Urena in [14]. Keller [10] suggests the usage of low discrepancy sequences for solving the rendering equation and proposes Quasi Monte Carlo approach. The idea is to distribute the samples into the domain of integration, as uniformly as possible in order to improve the convergence rate.

Further in this paper we consider the Monte Carlo solving of rendering equation with uniform hemisphere separation. The technique of uniform hemisphere partition was introduced by us and described in [3]. The uniform separation of the integration domain into *uniformly small by probability as well as by size* subdomains fulfills the conditions of the *Theorem for super convergence* presented in [12]. We show that the variance is bounded for numerical solving of the multidimensional integrals. Due to uniform separation of integration domain, this approach has hierarchical parallelism which is suitable for Grid implementations.

2 Rendering Equation for Photorealistic Image Creation

The light propagation in a scene is described by rendering equation [9], which is a second kind Fredholm integral equation. The radiance L, leaving from a point x on the surface of the scene in direction $\omega \in \Omega_x$ (see Fig. 1), where Ω_x is the hemisphere in point x, is the sum of the self radiating light source radiance L^e and all reflected radiance:

$$L(x,\omega) = L^e(x,\omega) + \int_{\Omega_x} L(h(x,\omega'),-\omega') f_r(-\omega',x,\omega) \cos\theta' d\omega'.$$

The point $y = h(x,\omega')$ indicates the first point that is hit when shooting a ray from x into direction ω'. The radiance L^e has non-zero value if the considered point x is a point from solid light source. Therefore, the reflected radiance in direction ω is an integral of the radiance incoming from all points, which can be seen through the hemisphere Ω_x in point x attenuated by the surface BRDF $f_r(-\omega',x,\omega)$ and the projection $\cos\theta'$, which puts the surface perpendicular to the ray (x,ω'). The angle θ' is the angle between surface normal in x and the direction ω'. The law for energy conservation holds, because a real scene always reflects less light than it receives from the light sources due to light absorption of the objects, i.e.: $\int_{\Omega_x} f_r(-\omega',x,\omega)\cos\theta' d\omega' < 1$. That means the incoming photon is reflected with a probability less than 1, because the selected energy is less than the total incoming energy. Another important property of the BDRF is the Helmholtz principle: the value of the BRDF will not change if the incident and reflected directions are interchanged, $f_r(-\omega',x,\omega) = f_r(-\omega,x,\omega')$.

Many BRDF for realistic image synthesis are based on surface microfacet theory. They are considered as function defined over all directions $\omega' \in \Omega_x$ (see [15]). For example, the BRDF function of Cook-Torrance (see [1], [4] and [11]) depends on the product of three components: Fresnel term - F, microfacets distribution function - D and geometrical attenuation factor - G; all depending on ω'. More detailed look at those functions gives us the assumption that Cook-Torrance BRDF has continuous first derivative.

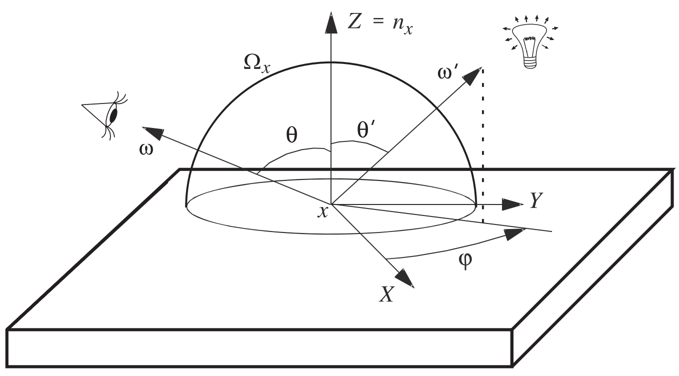

Fig. 1. The geometry for the rendering equation

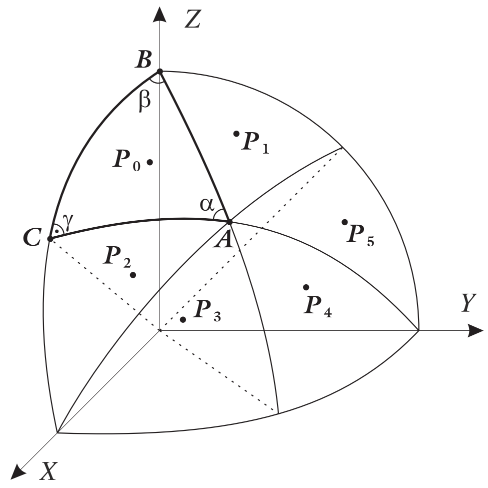

Fig. 2. Partitioning of the domain of integration

3 Parallel Monte Carlo Approach for the Rendering Equation

In order to solve the rendering equation by classical Monte Carlo approach we estimate the integral over the domain Ω_x. This is done by independently sampling N points according to some convenient probability density function $p(\omega')$, and then computing the Monte Carlo estimator ξ_N. Let us consider the sampling of the hemisphere Ω_x with $p(\omega') = \frac{1}{\Omega_x} = \frac{1}{2\pi}$, where $p = \int\limits_{\Omega_x} p(\omega')d\omega' = 1$. It is known that the estimator ξ_N has the following form:

$$\xi_N = \frac{2\pi}{N} \sum_{i=1}^{N} L^e(h(x, {\omega_i}'), -{\omega_i}') f_r(-{\omega_i}', x, \omega) \cos {\theta_i}'.$$

The parallel Monte Carlo approach for solving the rendering equation is based on the strategy for separation of the integration domain Ω_x into non-overlapping sub-domains, as described in [3]. We apply the symmetry property for partitioning of the hemisphere Ω_x . The coordinate planes partition the hemisphere into 4 equal areas. The partitioning of each one area into 6 equal sub-domains is continued by the three bisector planes. In Fig. 2 is shown the partitioning of the area with positive coordinate values of X, Y and Z into 6 equal sub-domains.

Let us now apply the partitioning of the hemisphere Ω_x into 24 non-overlapping equal size sub-domains of orthogonal spherical triangles Ω_{i_x}, where $\Omega_{\triangle ABC} = \Omega_{i_x} = \frac{1}{24}\Omega_x = \frac{\pi}{12}$ for $i_x = 1, 2, \ldots, 24$.

We can rewrite the rendering equation as:

$$L(x, \omega) = L^e(x, \omega) + \sum_{i_x=1}^{24} \int\limits_{\Omega_{i_x}} L(h(x, \omega'), -\omega') f_r(-\omega', x, \omega) \cos\theta' d\omega',$$

where $\Omega_x = \bigcup_{i_x=1}^{24} \Omega_{i_x}$. Therefore, the solution of rendering equation can be find as a sum of integrals over equal size non-overlapping sub-domains Ω_{i_x}.

Consider the probability:

$$p = \int_{\Omega_x} p(\omega')d\omega' = \sum_{i_x=1}^{24} \int_{\Omega_{i_x}} p(\omega')d\omega' = \sum_{i_x=1}^{24} p_{i_x} = 1,$$

it is obvious that $p_{i_x} = \int_{\Omega_{i_x}} p(\omega')d\omega' = \frac{1}{24}$ for $i_x = 1, 2, \ldots, 24$. Each sub-domain is sampled by random points $N_{i_x} \in \Omega_{i_x}$ with a density function $p(\omega')/p_{i_x}$. For all sub-domains N independent sampling points are generated in parallel using the sampling scheme for hemisphere from [3], where $N = 24N_{i_x}$ for $i_x = 1, 2, \ldots, 24$, t.e. the random sampling point are equal number in each sub-domain. In this case the sum of integrals for solving the separate rendering equation is estimated (see [12]) by:

$$\xi_N^* = \sum_{i_x=1}^{24} \frac{\pi}{12N_{i_x}} \sum_{s=1}^{N_{i_x}} L^e(h(x, \omega_{i_x,s}'), -\omega_{i_x,s}')f_r(-\omega_{i_x,s}', x, \omega) \cos\theta_{i_x,s}'.$$

Comparing the two approach it is known (see in [12]) that the variance of ξ_N^* is not bigger to the variance of ξ_N or always $Var[\xi_N^*] \leq Var[\xi_N]$. Somting more, one can see that in our case of domain separation $Var[\xi_N^*] = \frac{1}{24}Var[\xi_N]$. Also, the main advantages of this stratified sampling approach is the easy parallel realization.

4 Monte Carlo Solving of Multi-dimensional Integrals

The global illumination (see in Fig. 3) in realistic image synthesis can be modeled as stationary linear iterative process. According to the Neumann series, the numerical solving of rendering equation is iterative [13] process, where multi-dimensional integrals are considered.

For solving multi-dimensional integrals, let us suppose that k_ε is maximum level of recursion (recursion depth or number of iterations, see [1]) sufficient for

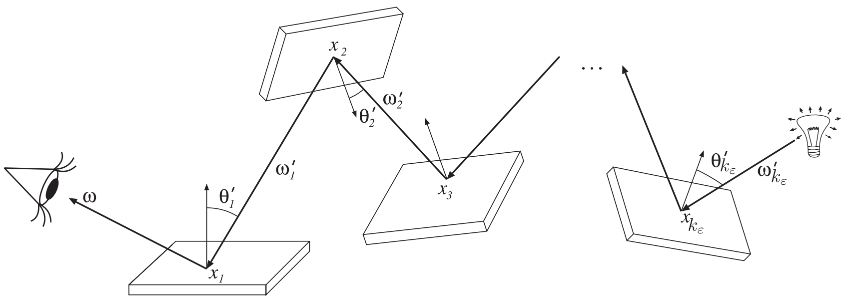

Fig. 3. Global illumination as iterative process

numerical solving of the integral with a desired truncation error ε. In this case on each iteration we have to solve the multi-dimensional integrals of the following type:

$$L^{(j)} = L_j - L_{j-1} = \int\limits_{\Omega_{x_1}} \dots \int\limits_{\Omega_{x_j}} K_1(x_1, \omega_1') \dots K_j(x_j, \omega_j') L^e(x_{j+1}, \omega_j') d\omega_1' \dots d\omega_j',$$

where $K_j(x_j, \omega_j') = f_r(-\omega_j', x_j, \omega_{j-1}') cos\theta_j'$ for $j = 1, \dots, k_\varepsilon$ and $L_0 = L^e(x_1, \omega)$ (note that $L_0 = L^e(x_1, \omega) = 0$ if the point x_1 is not a point from solid light source). The total domain of integration Ω_x can be represented as:

$$\Omega_x = \Omega_{x_1} \times \Omega_{x_2} \times \dots \times \Omega_{x_{k_\varepsilon}} = \prod_{j=1}^{k_\varepsilon} \left(\bigcup_{i_{x_j}=1}^{24} \Omega_{i_{x_j}} \right) = 24^{k_\varepsilon} \left(\frac{\pi}{12} \right)^{k_\varepsilon}.$$

Let us consider the integral $L^{(j)}$ in the case when $j = k_\varepsilon$ or $L^{(j)} = L^{(k_\varepsilon)}$. Using the partitioning of each domain Ω_{x_j} (for $j = 1, 2, \dots, k_\varepsilon$) of non-overlap equal size spherical triangle sub-domains $\Omega_{x_j} = \bigcup_{i_{x_j}=1}^{24} \Omega_{i_{x_j}}$ with size $\Omega_{i_{x_j}} = \left(\frac{\pi}{12}\right)$ for $i_{x_j} = 1, 2, \dots, 24$; we can rewrite the multi-dimensional integral $L^{(k_\varepsilon)}$ as:

$$L^{(k_\varepsilon)} = \sum_{i_{x_1}=1}^{24} \dots \sum_{i_{x_{k_\varepsilon}}=1}^{24} \int\limits_{\Omega_{i_{x_1}}} \dots \int\limits_{\Omega_{i_{x_{k_\varepsilon}}}} L^e(x_{k_\varepsilon+1}, \omega_{k_\varepsilon}') F(\omega_1', \dots, \omega_{k_\varepsilon}') d\omega_1' \dots d\omega_{k_\varepsilon}',$$

where $F(\omega_1', \dots, \omega_{k_\varepsilon}') = \prod_{j=1}^{k_\varepsilon} K_j(x_j, \omega_j')$. For numerical solving of integral $L^{(k_\varepsilon)}$, we use N realization of random samples and $N = 24^{k_\varepsilon}$. It means that only one random sample is generated in each sub-domain Ω_s for $s = 1, 2, \dots, N$, received after partitioning of Ω_x. Then approximate the integral $L^{(k_\varepsilon)}$ with $\xi_N^{*(k_\varepsilon)}$:

$$\xi_N^{*(k_\varepsilon)} = \left(\frac{\pi}{12} \right)^{k_\varepsilon} \sum_{s=1}^{N} L_s^e(x_{k_\varepsilon+1}, \omega_{k_\varepsilon}') F_s(\omega_1', \dots, \omega_{k_\varepsilon}')$$

with the integral approximation error $\varepsilon_N = \left| \xi_N^{*(k_\varepsilon)} - L^{(k_\varepsilon)} \right| = \sqrt{\frac{Var\left[\xi_N^{*(k_\varepsilon)}\right]}{N}}$.

According to the statements proofed in [12], the variance $Var\left[\xi_N^{*(k_\varepsilon)}\right]$ can be estimated as:

$$Var\left[\xi_N^{*(k_\varepsilon)}\right] \le c^2 L^2 N^{-1-\frac{2}{k_\varepsilon}},$$

where the first partial derivatives of $F(\omega_1', \dots, \omega_{k_\varepsilon}')$ are limited by an existing constant L, $\left|\frac{\partial F}{\partial \omega_j'}\right| \le L$ for $j = 1, 2, \dots, k_\varepsilon$ and the constant $c = k_\varepsilon c_1 c_2$. Also, there exist constants c_1 and c_2 such that:

$$p_s \le \frac{c_1}{N} \quad \text{and} \quad d_s \le \frac{c_2}{N^{\frac{1}{k_\varepsilon}}},$$

where p_s is the probability and d_s is diameter of the domain Ω_s for each $s = 1, 2, \dots, N$. Since all Ω_s for each $s = 1, 2, \dots, N$ are of equal size, it is obvious that $p = 24^{k_\varepsilon} p_s = 1$ or $p_s = \frac{1}{24^{k_\varepsilon}}$. The diameter d_s for each $s = 1, 2, \ldots, N$ can be calculated as $d_s = \sqrt{k_\varepsilon}\left|\max\left(\widehat{AC}, \widehat{AB}, \widehat{BC}\right)\right| = \sqrt{k_\varepsilon}\left|\widehat{AB}\right|$, where $\left|\widehat{AB}\right|$ is the length of arc $\widehat{AB}$ in the spherical triangle $\triangle ABC$ shown in Fig. 2. We recall the derived in [3] transformations, where $\tan\widehat{AB} = \frac{1}{\cos\varphi}$ at $\varphi = \frac{\pi}{4}$ and therefore the length of arc $\widehat{AB}$ is $\arctan\left(\sqrt{2}\right)$. Therefore, $d_s = \sqrt{k_\varepsilon}\arctan\left(\sqrt{2}\right)$ for $s = 1, 2, \ldots, N$.

Now we estimates the constants c_1 and c_2 by the inequalities:

$$\frac{N}{24^{k_\varepsilon}} \le c_1 \Longrightarrow 1 \le c_1 \qquad \text{and} \qquad d_s N^{\frac{1}{k_\varepsilon}} \le c_2 \Longrightarrow \sqrt{k_\varepsilon}\arctan\left(\sqrt{2}\right) N^{\frac{1}{k_\varepsilon}} \le c_2.$$

Therefore we can write:

$$Var\left[\xi_N^{*(k_\varepsilon)}\right] \le c^2 L^2 N^{-1-\frac{2}{k_\varepsilon}} = k_\varepsilon^2 c_1^2 c_2^2 L^2 N^{-1-\frac{2}{k_\varepsilon}}$$

which is equivalent to:

$$Var\left[\xi_N^{*(k_\varepsilon)}\right] \le k_\varepsilon^3 \arctan^2\left(\sqrt{2}\right) L^2 N^{-1} \Rightarrow Var\left[\xi_N^{*(k_\varepsilon)}\right] \le \arctan^2\left(\sqrt{2}\right) L^2 \frac{k_\varepsilon^3}{24^{k_\varepsilon}}.$$

The variance is bounded if we solve multi-dimensional integrals with uniform hemisphere separation approach. The last inequality shows us that the convergence rate for iterative solution of rendering equation with a desired truncation error ε depends on the sufficient recursion depth k_ε. Also, the multi-dimensional integrals are numerically solved with a rate of convergence $O(N^{-1})$. This is through the uniform separation of the integration domain into *uniformly small by probability as well by size* sub-domains, all of them matching the conditions of the *Theorem for super convergence* (see the proof in [12]). Summing the variance for all k_ε iterations we obtain: $Var\left[\xi_N^*\right] = \sum_{j=1}^{k_\varepsilon} Var\left[\xi_N^{*(j)}\right] = \left(\frac{N-1}{23N}\right) Var\left[\xi_N\right]$, where the variance $Var\left[\xi_N\right]$ indicates the variance for solving the rendering equation by N independent random sampling points without uniform separation of integration domain. Therefore, the total variance for solving of the rendering equation with uniform hemisphere separation is reduced.

5 Conclusion

The parallel Monte Carlo approach for solving of the rendering equation presented in this paper is based on partitioning of the hemispherical domain of integration by a way introduced by us in [3]. Essentially, this approach accumulates the stratified sampling by uniform separation of the integration domain. In fact, the uniform separation scheme is variance reduction approach and speed up the computations. The uniform separation of the integration domain hints for the applying of low discrepancy sequences as shown in [10]. The combination of uniform separation with the usage of low discrepancy sequences for numerical solving of the rendering equation could improve the uniformity of sampling points distribution and more so to reduce the variance. On the other hand this

Monte Carlo approach includes hierarchical parallelism. Therefore, it is suitable for implementation in algorithms with parallel realization of computations and completely can utilize the power of Grid computations. Thus, the main advantages of this approach lie in the efficiency of parallel computations. The future research of the parallel Monte Carlo approach under consideration for rendering equation could be developed in the following directions: 1) Investigation of the utilization of low discrepancy sequences with the uniform separation of integration domain. 2) Development of computational parallel Monte Carlo algorithms for creation of photorealistic images. 3) Creation of parallel Monte Carlo and Quasi Monte Carlo algorithms for high performance and Grid computing.

References

1. Dimov, I. T., Gurov, T. V. and Penzov, A. A., A Monte Carlo Approach for the Cook-Torrance Model, NAA 2004, LNCS 3401, pp. 257–265, Springer-Verlag Berlin Heidelberg, (2005).
2. Dimov, I., Karaivanova A., Georgieva, R. and Ivanovska, S., Parallel Importance Separation and Adaptive Monte Carlo Algorithms for Multiple Integrals, NMA 2002, LNCS 2542, pp. 99–107, Springer-Verlag Berlin Heidelberg, (2003).
3. Dimov, I. T., Penzov, A. A. and Stoilova, S. S., Parallel Monte Carlo Sampling Scheme for Sphere and Hemisphere, NMA 2006, August 20-24, 2006, Borovets, Bulgaria, (accepted for publication in LNCS).
4. Dutré, Philip, Global Illumination Compendium, Script of September 29 2003, http://www.cs.kuleuven.ac.be/∼phil/GI/TotalCompendium.pdf
5. Georgieva, Rayna and Ivanovska, Sofiya, Importance Separation for Solving Integral Equations, LSSC 2003, LNCS 2907, pp. 144–152, Springer-Verlag Berlin Heidelberg, (2004).
6. Ivanovska, Sofiya and Karaivanova, Aneta, Parallel Importance Separation for Multiple Integrals and Integral Equations, ICCS 2004, LNCS 3039, pp. 499–506, Springer-Verlag Berlin Heidelberg, (2004).
7. Karaivanova, Aneta, Adaptive Monte Carlo methods for numerical integration, Mathematica Balkanica, vol. 11, pp. 201 213, (1997).
8. Karaivanova, A. and Dimov, I., Error analysis of an adaptive Monte Carlo method for numerical integration.Mathematics and Computers in Simulation, vol. 47, pp. 391 406, (1998).
9. Kajiya, J. T., The Rendering Equation, Computer Graphics, vol. 20, No. 4, pp. 143–150, Proceedings of SIGGRAPH‘86, (1986).
10. Keller, A., Quasi-Monte Carlo Methods in Computer Graphics: The Global Illumination Problem, Lectures in Applied Mathematics, vol. 32, pp. 455–469, (1996).
11. Penzov, Anton A., Shading and Illumination Models in Computer Graphics - a literature survey, MTA SZTAKI, Research Report CG-4, Budapest, (1992).
12. Sobol, I., Monte Carlo Numerical Methods, (in Russian), Nauka, Moscow, (1975).
13. Szirmay-Kalos, Laszlo, Monte-Carlo Methods in Global Illumination, Script in WS of 1999/2000, http://www.fsz.bme.hu/ szirmay/script.pdf
14. Urena, Carlos, Computation of Irradiance from Triangles by Adaptive Sampling, Computer Graphics Forum, Vol. 19, No. 2, pp. 165–171, (2000).
15. Veach, Eric, Robust Monte Carlo Methods for Light Transport Simulation, Ph.D. Dissertation, Stanford University, December (1997).