Quasi-Monte Carlo Progressive Photon Mapping

Alexander Keller, Leonhard Grünschloß, and Marc Droske

Abstract The simulation of light transport often involves specular and transmissive surfaces, which are modeled by functions that are not square integrable. However, in many practical cases unbiased Monte Carlo methods are not able to handle such functions efficiently and consistent Monte Carlo methods are applied. Based on quasi-Monte Carlo integration, a deterministic alternative to the stochastic approaches is introduced. The new method for deterministic consistent functional approximation uses deterministic consistent density estimation.

1 Introduction

Photorealistic image synthesis aims at simulating the process of taking photographs. In principle, such simulations sum up the contributions of all transport paths which connect light sources with sensors.

An obvious approach to numerical simulation are bidirectional path tracing algorithms, where random walk methods are used to generate paths from the sensors and lights in order to connect them (as illustrated in Figure 1). However, there are common situations, where establishing such connections by checking visibility using so-called shadow rays can be arbitrarily inefficient.

As an example, one might think of light entering a car through a window, hitting the interior, and being transported back through the window to an outside observer. A similarly difficult situation is the observation of a room through a mirror (see

Alexander Keller
NVIDIA, e-mail: keller.alexander@gmail.com

Leonhard Grünschloß
NVIDIA / Weta Digital, e-mail: leonhard@gruenschloss.org

Marc Droske
NVIDIA, e-mail: marc.droske@gmail.com

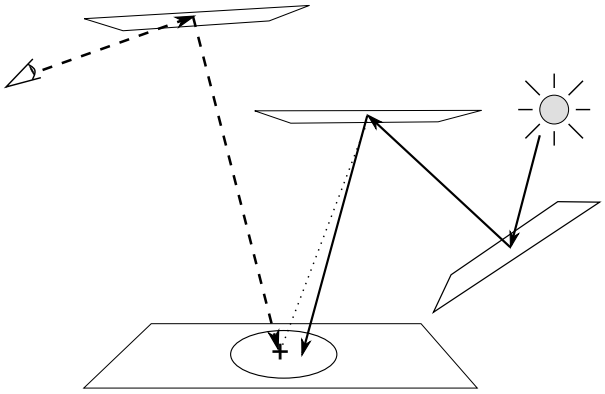

Fig. 1 Bidirectional generation of light transport paths: A path started from the eye (dashed rays) and a path started from a light source (solid rays) can be connected by a shadow ray (dotted line), which checks whether the vertices to connect are mutually visible. Alternatively, the basic idea of photon mapping is to relax the precise visibility check by allowing for a connection of both paths if their end points are sufficiently close as indicated by the circle.

Figure 2), where substantial illumination of the room is due to a small light source through the mirror, too. Such light transport paths cannot be established efficiently, because the direction of the connecting shadow ray has to coincide with the direction of a reflection on the mirror, which in fact happens with probability zero. In the context of bidirectional path tracing this problem has been characterized as the problem of "insufficient techniques" [12, Fig.2].

Key to efficiency is a shift of paradigm: Instead of considering unbiased Monte Carlo algorithms, allowing for a certain bias that vanishes in the limit opens up a new class of more efficient algorithms. Such algorithms are called consistent.

In computer graphics, photon mapping [6] has been developed in order to deal with the problem of "insufficient techniques". While in its first formulation, the technique was consistent only within infinite memory, progressive photon mapping [3] was introduced as an algorithm that converges pointwise within finite memory. In a follow-up article [2], a stochastic algorithm was derived that converges globally. Both latter publications provide example implementations. In [11] the stochastic arguments have been simplified, resulting in a simplified algorithm as well.

In contrast to previous work, which we detail in the next section, we introduce a deterministic photon mapping algorithm and prove its convergence. The method is based on sampling path space using quasi-Monte Carlo methods [13], which on the average allow for faster convergence as compared to Monte Carlo methods [20]. As a consequence of the deterministic nature, parallelization is simplified and results can be exactly reproduced even in a heterogeneous computing environment [10].

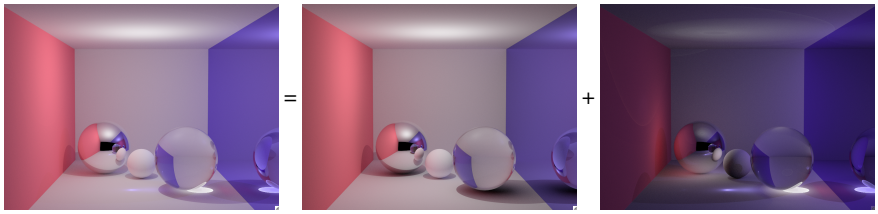

Fig. 2 The complete simulation of light transport (left) is the sum of light transported by square integrable surface properties (middle) and light transported by surfaces, whose physical properties are described by Dirac distributions (right). Unbiased Monte Carlo algorithms fail in robustly simulating the light transport path from the point light source in the center of the ceiling to the mirror onto the floor back to mirror into the camera. Consistent photon mapping efficiently can handle such paths. Caustics, i.e. the focal pattern underneath the glass ball, are another example of such paths.

2 Background on Photon Mapping

The principles of light transport simulation by tracing light transport paths are depicted in Figure 1. The basic idea of bidirectional path tracing [18] is to start paths from both the eye and the light source in order to connect them. Connecting the end points of both paths requires to check their mutual visibility using a shadow ray. As mentioned in the introduction, this technique can be arbitrarily inefficient if the reflection properties of at least one of the end points are not square integrable.

In this quite common situation, it is helpful to give up on precise visibility. Instead of tracing a shadow ray, end points are connected unless they are too distant. Photon mapping algorithms implement this principle by first tracing the trajectories of photons p that are emitted from the light sources and storing the incident energy, direction, and location, whenever a photon interacts with a surface. In a second step, paths are traced from the eye, whose contribution is determined by estimating the radiance [6, Sec. 7.2, Eq. 7.6]

$$L(x,\omega) \approx \frac{1}{\pi r^2} \sum_{p \in \mathscr{B}_x} f_s(\omega, x, \omega_p) \Delta\Phi_p \tag{1}$$

in the end point x of each eye path. The approximation is computed as an average over the area of a disk of radius r (see the circle in Figure 1), where the incident flux $\Delta\Phi_p$ attenuated by the bidirectional scattering distribution function (BSDF) f_s of all photons p in the ball $\mathscr{B}_x$ with respect to the query point x is summed up. The direction ω is the direction from which x is observed, while ω_p is the direction of incidence of the photon p.

Originally, the radius r had been selected as the radius of the ball enclosing the k closest photons around the query point x, which formed the set $\mathscr{B}_x$ [6]. Thus the radius was large in sparsely populated regions and small in regions of high photon density. In practice, this choice can result in numerical problems, because in high photon density regions the radius can approach zero arbitrarily close and thus the

term $\frac{1}{\pi r^2}$ cannot be bounded. The resulting overmodulation usually is not perceived, as it appears in bright regions anyhow.

A new class of algorithms that progressively refine the solution has been introduced in [3]. While the radius remains related to photon density in the aforementioned sense, each query location has its own radius, which is decreased upon photon interaction. As a consequence the radius decreases faster in brighter regions and may remain unchanged if shadowed, which, however, does not affect convergence. This approach has been reformulated in [11, Sec.4.3] such that no longer local statistics are required.

Since effects like perfect mirror reflection or refraction are modeled by Dirac-δ distributions, which are not square-integrable, they should not be part of the numerical evaluation of the reflective or refractive surface properties f_s. Instead, whenever such a component is encountered during tracing a path, Russian roulette is used to either terminate or prolong the path by simulating the perfect reflection or refraction, respectively [15]. Thus in practice the unbounded parts of f_s are never evaluated.

3 Pointwise Consistent Density Estimation

Photon mapping is strongly related to density estimation, where the radius is called smoothing parameter or smoothing length [16, Sec. 3.4]. Proofs of the consistency of density estimation [16, Sec. 3.7.1, Sec. 3.7.2] are based on choosing the radius in reciprocal dependence on a polynomial in the number n of emitted particles. Early work on photon mapping did not establish this reciprocal relationship and therefore only allowed for plausibility arguments [6, Sec. 7.2, Eq. 7.7]. Recent work [3, 2, 11] implicitly includes the reciprocal relationship, which allowed for more profound derivations.

In the following, a simpler and more general argument to prove

$$L(x,\omega) = \lim_{n\to\infty} \frac{1}{\pi \cdot r^2(x,n)} \sum_{p\in\mathscr{B}(x,r(x,n))} f_s(\omega,x,\omega_p)\Delta\Phi_p, \tag{2}$$

where $\mathscr{B}(x,r(x,n))$ is the set of all photons in the ball of radius $r(x,n)$ around the point x, is derived by explicitly choosing a squared radius

$$r^2(x,n) := \frac{r_0^2(x)}{n^\alpha} \qquad \text{for } 0<\alpha<1 \tag{3}$$

that includes the reciprocal dependence on a power of the number n of emitted photons. The explicit dependence on the query location x allows for choosing an initial radius $r_0(x) > 0$ in dependence of an initial photon distribution, similar to [11, Sec.4.3] and early photon mapping work.

The radiance estimator

$$L_n(x,\omega) := \frac{n^\alpha}{n \cdot \pi \cdot r_0^2(x)} \sum_{p \in \mathscr{B}(x,r(x,n))} f_s(\omega, x, \omega_p)\phi_p \tag{4}$$

results from including the number n of emitted photons in the photon flux $\Delta\Phi_p := \frac{\phi_p}{n}$ and inserting it into Equation (2).

The radiance estimator can be generalized by using any other kernel that in the limit results in a Dirac-δ distribution [11]. Such kernels, other than the characteristic function of the set $\mathscr{B}(x,r(x,n))$, are found in [16] or in the domain of smoothed particles hydrodynamics (SPH). In analogy with the SPH approach, using the derivative of such a kernel allows one to compute irradiance gradients.

3.1 Choice of the Parameter α

For $n > 1$ we have $\frac{n^\alpha}{n} < 1$ due to the postulate $0 < \alpha < 1$. As a consequence L_n will always be bounded, because the evaluation of f_s is bounded as established at the end of Section 2.

Since light transport is a linear problem, the number of photons in $\mathscr{B}(x,r(x,n))$ asymptotically must be linear in n: For $\alpha = 1$ doubling the number n of emitted photons results in half the squared radius, meaning half the area, while the number of photons in $\mathscr{B}(x,r(x,n))$ remains the same. For $0 < \alpha < 1$ the squared radius decreases slower than the increase in number of photons. As a consequence, more and more photons are collected with increasing n, which guarantees $L(x,\omega) = \lim_{n\to\infty} L_n(x,\omega)$.

Note that Equation (2) does neither converge for $\alpha = 0$, because the initial radius will not be decreased, nor for $\alpha = 1$ as the noise level does not decrease. This can be easily verified by running the algorithm with either one of the extremal values. Comparing the graphs of $\frac{n^\alpha}{n}$ for the two extreme cases reveals that $\alpha = \frac{1}{2}$ in fact best balances the two interests of fast convergence and noise reduction. However, this choice is not crucial at all as shown in the next section.

3.2 Choice of the Initial Radius r_0

The limit of the ratio of the $(n+1)$-st and n-th squared radius reveals that the squared radius is vanishing arbitrarily slowly:

$$\lim_{n\to\infty} \frac{r^2(x,n+1)}{r^2(x,n)} = \lim_{n\to\infty} \frac{n^\alpha}{(n+1)^\alpha} = \lim_{n\to\infty} \left(\frac{n}{n+1}\right)^\alpha = 1$$

As a consequence, a larger value of α is only effective for smaller n and therefore the initial radius r_0 becomes most influential. However, competing goals need to be satisfied: In favor of efficiency, a smaller radius requires less photons to be averaged,

while on the other hand a larger radius allows for more efficient high frequency noise reduction by averaging more photons.

While a local initial radius allows for adapting to the photon density and thus a better trade-off between noise and smoothing, it requires the retrieval of $r_0(x)$ [11, Sec.4.3]. For example $r_0(x)$ can be obtained from an initial set of photons in analogy to the original photon mapping algorithm. Alternatively, an individual radius can be stored for each functional, for example for each pixel to be computed. If in addition $r_0(x)$ can be bounded efficiently, for example by determining its maximum, the efficiency of nearest neighbor search can be improved.

Of course the simplest choice is a global initial radius r_0, which we prefer to choose rather smaller than larger, as the human visual system is more comfortable with high frequency noise than blotchy low-frequency averaging artifacts.

4 Consistent Functional Approximation

In fact, Equation (4) can be considered an integro-approximation problem: Given one set of photons generated by n paths started at the light sources, the radiance L_n is defined for any location x and any direction ω.

This allows one to compute the color

$$\begin{aligned} L_P &:= \lim_{m\to\infty} \frac{|P|}{m} \sum_{q=0}^{m-1} \chi_P(x_q) W(x_q) L(h(x_q), \omega(x_q)) \\ &= \lim_{m\to\infty} \frac{|P|}{m} \sum_{q=0}^{m-1} \chi_P(x_q) W(x_q) \lim_{n\to\infty} L_n(h(x_q), \omega(x_q)) \qquad (5) \\ &\approx \frac{|P|}{mn} \sum_{q=0}^{m-1} \chi_P(x_q) W(x_q) \frac{n^\alpha}{\pi \cdot r_0^2(h(x_q))} \qquad (6) \\ &\qquad \cdot \sum_{p\in\mathscr{B}(h(x_q),r(h(x_q),n))} f_s(\omega(x_q), h(x_q), \omega_p)\phi_p \end{aligned}$$

of a pixel P using an infinite sequence of uniform samples x_q to determine query locations $h(x_q)$: The x_q define eye paths, where $W(x_q)$ is the accumulated weight along the path, which is multiplied by the radiance $L(h(x_q), \omega(x_q))$. The paths associated with the pixel P are selected by the characteristic function χ_P, while $|P|$ is the area of pixel P.

Computing the functional (5) requires the enumeration of all pairs of indices (q, p) of query paths and photons (see Figure 3). This way each query location $h(x_q)$ can interact with all photons, which guarantees the pointwise convergence of Equation (4) and consequently the approximation (6) is consistent.

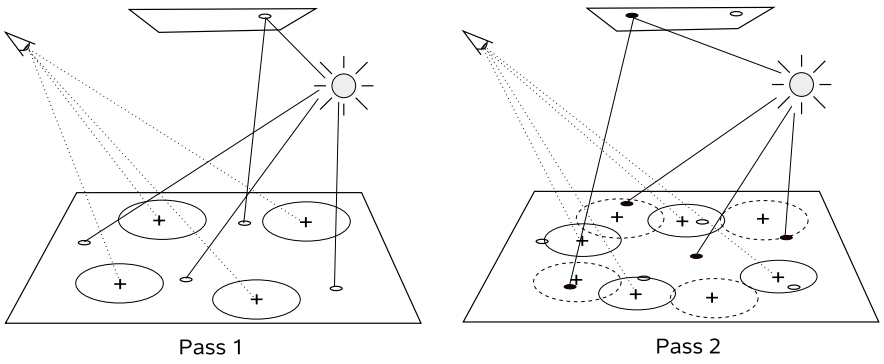

Fig. 3 Just taking two deterministic point sequences to enumerate all eye and light paths in passes can fail: In the illustration, the photons of the second pass would interact with the query locations of the first pass, however, these interactions never become enumerated as pairs. A black image results, although light is transported to the eye. While such illustrative situations can be constructed artificially, they occur in a practical setting as well.

4.1 Algorithm

As derived in the previous section, each query location must interact with all photons, which requires the enumeration of all pairs of query path and light path indices. Therefore $\mathbb{N}_0 \times \mathbb{N}_0$ is partitioned into contiguous blocks of m_b query location indices times n_b light path indices. The ratio $\frac{m_b}{n_b}$ allows for balancing pixel anti-aliasing and photon density. The blocks are enumerated using the index i along the dimension of query paths and j along the dimension of light paths.

Obviously it is most efficient to keep as many query locations and photons as possible in memory. However, as an unavoidable consequence of finite memory, both query locations and photons need to be recomputed. This excess amount of computation depends on the order of how the blocks are enumerated. While the diagonal order in Figure 4a) requires permanent recomputation, the rectangular order in Figure 4b) allows for frequent reuse of either the set of query locations or the set of photons. Such space filling curves are easily implemented, even with direct block access, which allows for parallelization without communication or synchronization [10].

The rigid partition into blocks of equal size can be avoided by generating query locations and photons until a given block of memory is filled. The resulting starting indices m_i and n_j for the query locations and light paths, respectively, are stored in an array each in order to allow for the direct retrieval of the i-th range $m_i, \ldots, m_{i+1} - 1$ of query paths and the j-th range $n_j, \ldots, n_{j+1} - 1$ of light paths.

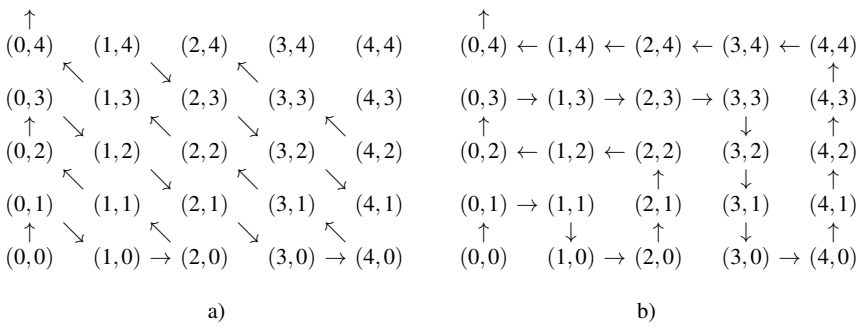

Fig. 4 Enumerating all combinations of integers in the a) classic diagonal order used for enumerating the rational numbers and b) an order that results in much better data coherence and caching.

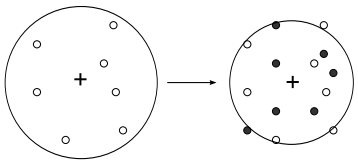

Fig. 5 The radius shrinks with the number of photons blocks enumerated, while at the same time the contribution of the individual photons that were collected with larger radius fades out.

4.2 Consistency of Block-Wise Computation

If the number n of light paths can be fixed in advance, the radius $r(x,n)$ will be used throughout the computation and the sums will be weighted by $\frac{1}{mn}$ as shown in approximation (6).

If the ultimate value of n is unknown, the computation will have to be conducted in a progressive way. The weight for the intermediate results then amounts to the reciprocal of the number of currently processed blocks multiplied by $m_b n_b$, which is the number processed pairs of query points and light paths.

A block with light path block index j is processed using the radius $r(x, j \cdot n_b)$. Note that the algorithm (5) remains consistent, because the weight of each single summand decreases with increasing number of blocks. As j increases, less photons interact with the query locations, since the query radius decreases (see Figure 5). This can be interpreted as slight blur that sharpens with the progress of the computation. As the radius decreases arbitrarily slow (see Section 3.2), this effect is hardly visible, which again emphasizes that the choice of the initial radius is much more important than the overall progression of the radius.

4.3 Deterministic Sampling using Quasi-Monte Carlo Points

Pseudo-random number generators in fact are deterministic algorithms that try to mimic random numbers. However, the approximate independence of pseudo-random numbers is no longer visible once the samples are averaged. More important, the speed of convergence depends on the uniformity of the samples. In that respect deterministic low discrepancy sequences are preferable, as they are more uniform as compared to random samples and therefore improve the speed of convergence [5, 13, 7, 9]. Finally, such deterministic algorithms are simple to parallelize in heterogeneous computing environments and results are exactly reproducible [10].

Interpreting the radiance estimator (4) as an integro-approximation problem allows for applying the results of [9, Theorem 1], which guarantee that generating the photons using deterministic low discrepancy sequences [13] results in a consistent deterministic estimation with all the aforementioned advantages.

With the point-wise convergence of the radiance estimate established, any consistent quadrature rule can be used for sampling the query location paths. Especially, the same low discrepancy sequence as used for photon generation can be applied, which simplifies implementations.

Constructions of low discrepancy sequences are found in [13]. The images in Figure 2 have been computed using the Halton sequence. We also verified the theoretical results using fast implementations of (t,s)-sequences in base b, especially the Sobol' sequence [17, 19], and rank-1 lattices sequences [4].

Note that the point sequences must be dimension extensible in order to account for potentially infinite length transport paths, which in theory would rule out rank-1 lattices and constructions similar to the Faure sequences [1]. However, due to finite precision, path length cannot be infinite on a computer and it is reasonable and acceptable to limit path length by a sufficiently large bound. While in theory this leads to inconsistent results, in practice the resulting bias is not observable in most cases.

For the sake of completeness, we note that the algorithms derived in this article work with any point sequence that is uniform, i.e., has vanishing discrepancy. This includes random, pseudo-random, or randomized point sequences such as for example randomized low discrepancy sequences.

Samples of uniform sequences can be transformed to path space samples using approaches explained in detail in [18]. We therefore only point out that the paths resulting in the query points are generated by sampling the whole image plane or tiles thereof instead of sampling on a pixel-by-pixel basis. While it is possible to simply map the image plane or tiles thereof to the unit square, it may be preferable to directly map pixels to sample indices [8, 9, 14].

5 Results and Discussion

Figure 2 shows a classic test scene, where the algorithm was used to simulate light transport completely and only in parts, especially caustics. The derived method has been proven to unconditionally converge and can be used as an efficient substitute for other photon mapping implementations.

Other than in Equation (5), the derivation of stochastic progressive photon mapping [2] does not allow all query locations to interact with all photons. While it is still possible to argue that stochastic progressive photon mapping is converging as long as random sampling is used, the algorithm cannot be derandomized by just using deterministic samples, because then it is possible to construct scenarios that do not converge (see Figure 3). If for example the camera is used as light source at the same time, query paths and light paths are identical and therefore perfectly correlated. As path space is not uniformly sampled, visible illumination reconstruction artifacts, like for example overmodulation, become visible.

6 Conclusion

We introduced quasi-Monte Carlo progressive photon mapping. Based on the principles of enumerating all pairs of non-negative integers, convergence has been proven for the deterministic case.

The simple derivation and algorithmic principle enable the deterministic and consistent computation of many more linear problems as for example all kinds of (bidirectional) path tracing, in which query and light paths are connected by shadow rays. If path space sampling is extended to consider participating media, the proposed schemes generalize to volume scattering as well [11, Sec.4.2].

Acknowledgements

This work has been dedicated to Jerry Spanier's 80th birthday.

References

1. Faure, H.: Discrépance de suites associées à un système de numération (en dimension *s*). Acta Arith. **41**(4), 337–351 (1982)
2. Hachisuka, T., Jensen, H.: Stochastic progressive photon mapping. In: SIGGRAPH Asia '09: ACM SIGGRAPH Asia 2009 papers, pp. 1–8. ACM, New York, NY, USA (2009)
3. Hachisuka, T., Ogaki, S., Jensen, H.: Progressive photon mapping. ACM Transactions on Graphics **27**(5), 130:1–130:8 (2008)

4. Hickernell, F., Hong, H., L'Ecuyer, P., Lemieux, C.: Extensible lattice sequences for quasi-Monte Carlo quadrature. SIAM J. Sci. Comput. **22**, 1117–1138 (2001)
5. Hlawka, E., Mück, R.: Über eine Transformation von gleichverteilten Folgen II. Computing **9**, 127–138 (1972)
6. Jensen, H.: Realistic Image Synthesis Using Photon Mapping. AK Peters (2001)
7. Keller, A.: Quasi-Monte Carlo Methods for Photorealistic Image Synthesis. Ph.D. thesis, University of Kaiserslautern, Germany (1998)
8. Keller, A.: Strictly Deterministic Sampling Methods in Computer Graphics. SIGGRAPH 2003 Course Notes, Course #44: Monte Carlo Ray Tracing (2003)
9. Keller, A.: Myths of computer graphics. In: H. Niederreiter (ed.) Monte Carlo and Quasi-Monte Carlo Methods 2004, pp. 217–243. Springer (2006)
10. Keller, A., Grünschloß, L.: Parallel quasi-Monte Carlo methods. In: L. Plaskota, H. Woźniakowski (eds.) Monte Carlo and Quasi-Monte Carlo Methods 2010, p. to appear. Springer (2011)
11. Knaus, C., Zwicker, M.: Progressive photon mapping: A probabilistic approach. ACM Transactions on Graphics (TOG) **30**(3) (2011)
12. Kollig, T., Keller, A.: Efficient bidirectional path tracing by randomized quasi-Monte Carlo integration. In: H. Niederreiter, K. Fang, F. Hickernell (eds.) Monte Carlo and Quasi-Monte Carlo Methods 2000, pp. 290–305. Springer (2002)
13. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. SIAM, Philadelphia (1992)
14. Raab, M., Grünschloß, L., Keller, A.: Enumerating certain quasi-Monte Carlo sequences in voxels. In: H. Wozniakowski, L. Plaskota (eds.) Monte Carlo and Quasi-Monte Carlo Methods 2010, p. in this volume. Springer (2011)
15. Shirley, P.: Realistic Ray Tracing. AK Peters, Ltd. (2000)
16. Silverman, B.: Density Estimation for Statistics and Data Analysis. Chapman & Hall/CRC (1986)
17. Sobol', I.: Uniformly Distributed Sequences with an additional Uniform Property. Zh. vychisl. Mat. mat. Fiz. **16**(5), 1332–1337 (1976)
18. Veach, E.: Robust Monte Carlo Methods for Light Transport Simulation. Ph.D. thesis, Stanford University (1997)
19. Wächter, C.: Quasi-Monte Carlo Light Transport Simulation by Efficient Ray Tracing. Ph.D. thesis, Universität Ulm (2008)
20. Woźniakowski, H.: Average case complexity of multivariate integration. Bull. Amer. Math. Soc. **24**, 185–194 (1991)