ANNÉE 2013

UNIVERSITÉ DE RENNES 1

ueb

THÈSE / UNIVERSITÉ DE RENNES 1
sous le sceau de l'Université Européenne de Bretagne

pour le grade de

DOCTEUR DE L'UNIVERSITÉ DE RENNES 1

Mention : Informatique

Ecole doctorale Matisse

présentée par

Ricardo Marques

préparée à l'unité de recherche UMR 6074 IRISA
et au centre INRIA - Rennes Bretagne Atlantique
ISTIC

Bayesian and Quasi Monte Carlo Spherical Integration for Global Illumination

Thèse soutenue à Rennes
le 22 octobre 2013

devant le jury composé de :

Victor Ostromoukhov
Professeur, Univ. de Lyon 1, France / rapporteur

Pierre Poulin
Professeur, Univ. de Montréal, Canada / rapporteur

Eric Marchand
Professeur, Univ. de Rennes 1, France / examinateur

António Augusto De Sousa
Professeur, Univ. do Porto, Portugal / examinateur

Christian Bouville
Collaborateur Extérieur, IRISA / examinateur

Luís Paulo Santos
Maître de Conférences, Univ. do Minho,
Portugal / examinateur

Kadi Bouatouch
Professeur, Univ. de Rennes 1 / directeur de thèse

Abstract

The spherical sampling of the incident radiance function entails a high computational cost. Therefore the illumination integral must be evaluated using a limited set of samples. Such a restriction raises the question of how to obtain the most accurate approximation possible with such a limited set of samples. In this thesis, we show that existing Monte Carlo-based approaches can be improved by fully exploiting the information available which is later used for careful samples placement and weighting.

The first contribution of this thesis is a strategy for producing high quality Quasi-Monte Carlo (QMC) sampling patterns for spherical integration by resorting to spherical Fibonacci point sets. We show that these patterns, when applied to the rendering integral, are very simple to generate and consistently outperform existing approaches. Furthermore, we introduce theoretical aspects on QMC spherical integration that, to our knowledge, have never been used in the graphics community, such as spherical cap discrepancy and point set spherical energy. These metrics allow assessing the quality of a spherical point set for a QMC estimate of a spherical integral.

In the next part of the thesis, we propose a new theoretical framework for computing the Bayesian Monte Carlo (BMC) quadrature rule. Our contribution includes a novel method of quadrature computation based on spherical Gaussian functions that can be generalized to a broad class of BRDFs (any BRDF which can be approximated by a sum of one or more spherical Gaussian functions) and potentially to other rendering applications. We account for the BRDF sharpness by using a new computation method for the prior mean function. Lastly, we propose a fast hyperparameters evaluation method that avoids the learning step.

Our last contribution is the application of BMC with an adaptive approach for evaluating the illumination integral. The idea is to compute a first BMC estimate (using a first sample set) and, if the quality criterion is not met, directly inject the result as prior knowledge on a new estimate (using another sample set). The new estimate refines the previous estimate using a new set of samples, and the process is repeated until a satisfying result is achieved.

Acknowledgements

First of all, I would like to gratefully acknowledge the guidance of my supervisor Kadi Bouatouch throughout the three years of my Ph.D. studies. He taught me an enormous amount of things and always pushed me to improve at all levels. His support was not limited to the excellent working conditions I was provided, but also to many aspects of my personal life such as helping me to integrate in a new country and motivating me to run my first half marathon (which we ran together). I am very grateful to him for his perseverance and for having believed in my capacities.

I am also indebted to Christian Bouville for the infinite time and patience he dedicated to me. During these three years Christian was always present, and our long discussions were of central importance for the work described in this manuscript. I feel very privileged to have worked with such a skilled and passionate researcher.

I must express as well my sincere gratitude to Luís Paulo Santos. His contribution gave important pushes to this Ph.D. thesis especially during his visits to Rennes. Nevertheless, I would like to take this opportunity to thank him for more than just the advices and guidance during these three years. Luís Paulo initiated me to computer graphics and motivated me to this field with his captivating courses at Universidade do Minho. He later directed my Master thesis and gave me my first job in computer graphics. Without his support I would not have had the opportunity to pursue this Ph.D. thesis.

I would like to thank my colleagues from the FRVSense team, especially Mickaël Ribardière with whom I shared the office during all this time. I must also thank Adrien Gruson, Billal Merabti and Charly Collin for letting me beat them at ping-pong in the most difficult times.

Finally, a special mention to my girlfriend Florina for unconditional support and love throughout these three hard working years.

Contents

List of Figures

List of Tables

Introduction

1

Contents

1 The global illumination problem

One of the most challenging problems in the field of computer graphics is to synthesize physically-based realistic images given an accurate model of a virtual scene. This model contains geometric information describing the shape of the objects as well as their corresponding material characteristics. It also contains a description of the light sources, and a camera specifying how the scene is 'seen'.

The light emitted by the light sources will then interact with the objects of the scene through multiple inter-reflections. The process of synthesizing a realistic image consists of simulating these interactions and determining how much light enters the virtual camera. It has been mathematically formulated by Kajya [Kaj86] and is given by the following equation:

$$L_o(\boldsymbol{\omega}_o) = L_e(\boldsymbol{\omega}_o) + \int_{\Omega_{2\pi}} L_i(\boldsymbol{\omega}_i)\ \rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)\ (\boldsymbol{\omega}_i \cdot \mathbf{n})\ d\Omega(\boldsymbol{\omega}_i) \tag{1.1}$$

which is referred to as *illumination integral*. For a given point at a surface, Eq. (1.1) gives the outgoing radiance L_o in direction $\boldsymbol{\omega}_o$. L_e is the radiance due to self emission. $L_i(\boldsymbol{\omega}_i)$ is the incident radiance from direction $\boldsymbol{\omega}_i$ and ρ is the *bidirectional reflectance distribution function* (BRDF). It models the fraction of the radiance arriving from the direction $\boldsymbol{\omega}_i$ which is reflected at the surface in direction $\boldsymbol{\omega}_o$. The radiance $L_i(\boldsymbol{\omega}_i)$ arriving at a given surface point can be separated in two main components:

- the **direct component** due to the radiance arriving at the surface point directly from the light sources;
- and the **indirect component** which corresponds to the light arriving at the surface point after one or more bounces.

Due to computing power constraints, the first image synthesizers only evaluated the direct component. Nowadays, all the realistic image synthesis programs allow a full evaluation of the illumination integral. This task of synthesizing an image taking into account both the direct and the indirect light components defines the *global illumination* (GI) problem.

2 Illumination integral evaluation

The illumination integral (Eq. (1.1)) has no analytical solution except for very particular cases in which very strong hypotheses are assumed. Several distinct techniques have been developed to provide a solution to Eq. (1.1). Some approaches are based on a brute force extensive evaluation of the integral by resorting to classical Monte Carlo techniques [Kaj86]. Others are based on the observation that there exists a certain spatial coherence in the illumination and try to exploit this coherence by trading off a higher error for efficiency. But despite their differences, all these approaches have in common requiring a sampling phase.

A good sampling of the image plane is necessary to produce a final image free from aliasing [CPC84]. To simulate effects such as motion blur and depth-of-field one needs to sample the time dimension and the camera lens respectively. The Monte Carlo rendering algorithms which operate in the ray paths space, such as bidirectional path tracing [VG94, LW93], rely on sampling for simulating both the eye paths and the light paths existing in the scene. As shown by Veach [Vea98], the evaluation of the direct incident radiance benefits from resorting to a combination of sampling the light sources and a hemispherical sampling of the outgoing directions leaving the surface point with a probability proportional to the BRDF. This technique of combining two different Monte Carlo estimators is referred to as multiple importance sampling. For evaluating the indirect radiance component, distinct methods such as radiance caching [WRC88], irradiance caching [KGPB05] and photon mapping [Jen96] require at some stage hemispherical sampling around the surface normal at a given surface point.

The quality of the result of these sampling operations is strongly dependent on the samples placement and weighting. Therefore several works have focused on improving the purely random sampling used in classic Monte Carlo techniques [Kel12]. Their approach consists of using deterministic sequences which improve the uniformity of the samples distribution on the domain of integration. The resulting estimator is then called a Quasi-Monte Carlo (QMC) estimator.

In this thesis we will focus on the case of hemispherical sampling for spherical integration. We will show that existing approaches can be improved by fully exploiting the information available which is later used for careful samples placement and weighting.

3 Motivation

The spherical sampling of the incident radiance function entails a high computational cost. Therefore the illumination integral must be evaluated using a limited set of

samples. Such a restriction raises the question of how to obtain the most accurate approximation possible with such a limited set of samples. We need to ensure that sampling produces the highest amount of information possible by carefully placing the limited set of samples. Furthermore we want our integral evaluation to take into account not only the information produced by the sampling but also eventual information available prior to sampling.

The idea of sparse information and the need to fully exploit the few information available will be present throughout this thesis. Indeed, the contributions summarized in the next section improve the state of the art solutions in computer graphics by taking into account information which had so far been underexploited (or even neglected) by the previous approaches. Examples of this type of information are the hemispherical nature of the domain of integration, the relative samples position and the statistical properties of the integrand.

4 Summary of contributions

The contributions presented in this thesis can be summarized as follows:

Spherical Fibonacci Point Sets for Illumination Integrals Quasi-Monte Carlo (QMC) methods exhibit a faster convergence rate than that of classic Monte Carlo methods. This feature has made QMC prevalent in image synthesis, where it is frequently used for approximating the value of spherical integrals (e.g. illumination integral). The common approach for generating QMC sampling patterns for evaluating the illumination integral is to resort to point sets with low unit square discrepancy and map them to the hemisphere [KK02, GHSK06]. But such an approach is suboptimal since these sequences do not account for the spherical topology and their low discrepancy properties on the unit square are impaired by the spherical projection. We present a strategy for producing high quality QMC sampling patterns for spherical integration by resorting to spherical Fibonacci point sets. We show that these patterns, when applied to the rendering integral, are very simple to generate and consistently outperform existing approaches. Furthermore, we introduce theoretical aspects on QMC spherical integration that, to our knowledge, have never been used in the graphics community, such as spherical cap discrepancy and point set spherical energy. These metrics allow assessing the quality of a spherical point set for a QMC estimate of a spherical integral.

A Spherical Gaussian Framework for Bayesian Monte Carlo Bayesian Monte Carlo [RG02] is a non-frequentist integration technique which has recently been successfully applied to the computation of the illumination integral. This pdf-free technique allows a meticulous placement of the samples over the domain of integration. The samples position is chosen so as to minimize the variance of the estimate. The samples value is then averaged according to a quadrature rule where the weight of each sample is determined by resorting to a covariance function. The results obtained by Brouillat et al. [BBL+09] show that Bayesian Monte Carlo can significantly outperform importance sampling Monte Carlo thanks to a more effective use of the prior knowledge and

of the information brought by the samples. But these results are strictly limited to diffuse reflections since their strategy for efficiently computing the quadrature coefficients does not apply to non-diffuse BRDFs. We propose a new theoretical framework that includes a novel method of quadrature computation based on spherical Gaussian functions that can be generalized to a broad class of BRDFs (any BRDF which can be approximated by a sum of one or more spherical Gaussian functions) and potentially to other rendering applications. We account for the BRDF sharpness by using a new computation method for the prior mean function and by introducing a new factor in the method of Brouillat et al. for constructing optimized samples set. Lastly, we propose a fast hyperparameters evaluation method that avoids the learning step.

Adaptive Sampling with Bayesian Monte Carlo Photo-realistic images synthesized with Monte Carlo and Quasi-Monte Carlo techniques are prone to noise. The noise is a consequence of the variance of the estimate (or error of estimate in the case of QMC) which results from a too low sampling rate. In general, the noise is not evenly distributed in all image regions since the smoothness of the signal to be sampled varies across the scene. For the same noise level, regions corresponding to a smoother illumination require fewer samples than regions with a sharper illumination. The rendering techniques which aim at distributing the samples according to the noise level detected in the different image regions are called *adaptive sampling* (e.g., [Mit87, LRU85, XFSS07, XSFS10]). The common approach is to make a first pass using a low sampling rate, detect the image regions for which the noise level is not acceptable and oversample them until a certain quality criterion is met. Our goal is to apply BMC with an adaptive approach for evaluating the illumination integral. The idea is to compute a first BMC estimate (using a first sample set) and, if the quality criterion is not met, directly inject the result as prior knowledge on a new estimate (using another sample set). The new estimate refines the previous estimate using a new set of samples, and the process is repeated until a satisfying result is achieved.

Spherical Fibonacci Point Sets for Illumination Integral

2

Contents

The major contribution presented in this chapter is the introduction of the spherical Fibonacci point sets for Quasi-Monte Carlo (QMC) integration, based on previous works [HN04, SP06] from other research areas. We compare the quality of Fibonacci point sets for estimating the illumination integral with that of state-of-the-art point sets compliant with QMC such as blue noise [dGBOD12], Larcher-Pillichshammer [LP01] and the popular Sobol (0,2)-sequence [Sob67]. We show that the Fibonacci point sets consistently outperform those methods. Furthermore, we introduce theoretical aspects on QMC spherical integration that, to our knowledge, have never been used in the graphics community. In concrete terms, we define worst case integration error (w.c.e.), spherical L_2 discrepancy and an energy metric E_N based on the inter-sample distance which allows assessing the quality of a spherical samples set for spherical integration.

1 Introduction

Among all the methods which have been proposed to speed up Monte Carlo integration for rendering, QMC methods play an important role as they allow improving the convergence rate as well as controlling the error noise perception. The principle is to use more regularly distributed sample sets than the purely random sample sets associated with Monte Carlo integration.

QMC integration is extensively used in computer graphics (see e.g. [SEB08]). In [Kel12], Keller has shown that QMC techniques can be applied in a consistent way to deal with a wide range of problems (anti-aliasing, depth of field, motion blur, spectral rendering, etc.). However, few applications have been reported in the literature specifically addressing hemispherical sampling with a view of computing the illumination integral.

Unlike the unit square sampling case, no explicit construction of point sets with minimal discrepancy for spherical sampling is known. Generally, the point sets for spherical integration are generated by lifting point sets with low discrepancy on the unit square to the unit sphere through an equal-area transform [MFS04, Ost07]. But such an approach is suboptimal as the low discrepancy properties of these point sets on the unit square are impaired by the spherical projection. Recent results from the numerical analysis literature have focused on a point set which is directly generated on the sphere called *spherical Fibonacci point sets* [ABD12, BD11]. They have shown that these point sets are particularly well-suited to sphere sampling compared to the traditional low-discrepancy point sets projected from the unit square. In this chapter, we focus on adapting the spherical Fibonacci point sets for evaluating the illumination integral.

The rest of this chapter is structured as follows. In Section 2, we introduce theoretical concepts regarding spherical integration using QMC. The related work is presented in Section 2.5. Section 3 gives a detailed description of the Fibonacci spherical point sets. In Section 4, we specify how we have implemented BRDF sampling in the context of QMC integration and make explicit the interest in generating high quality spherical distributions for this particular case. Sections 5 and 6 present the benefits of using Fibonacci point sets compared to a Sobol sequence, blue noise point sets and the Larcher-Pillichshammer points. We end the chapter with a conclusion and future work.

2 Background and related work

2.1 QMC on the unit square

Classic Monte Carlo (CMC) methods are frequently used to approximate integrals of the form:

$$I = \int_{[0,1]^2} f(\mathbf{x}) d\mathbf{x}$$

where f is a real valued function defined over $[0,1]^2$. Their approach consists in using a set $P_N = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ of N random points uniformly distributed on $[0,1]^s$ and computing an estimate $\hat{I}$ of I by averaging of the values of f over P_N [L'E09]:

$$\hat{I} = \frac{1}{N} \sum_{j=1}^{N} f(\mathbf{x}_j)$$

Quasi-Monte Carlo can be seen as the deterministic counter-part of CMC, where the point set P_N is constructed deterministically and not at random. The goal of QMC integration is to find sampling patterns that yield a better order of convergence than the $\mathcal{O}(N^{-1/2})$ rate obtained with purely random distributions. In the case of QMC integration over the unit square $[0,1]^2$ the best theoretical rate of convergence of the worst case error is $\mathcal{O}(N^{-1}\sqrt{\log N})$ (see e.g. [BD11]). To find point set constructions that approximate this optimal rate of convergence, the star discrepancy is often used as a criterion to characterize the uniformity of the point distribution. Given a set of boxes B with a corner at the origin of the unit square given by:

$$B = \{[0,x] \times [0,y]\}, (x,y) \in [0,1]^2,$$

the star discrepancy D^* can be defined as:

$$D^*(P_N; B) = \sup_{\mathcal{C} \in B} \left| \frac{\text{Card}\{j : \mathbf{x}_j \in \mathcal{C}\}}{N} - \lambda(\mathcal{C}) \right|$$

where $\lambda(\mathcal{C})$ is the area of a box $\mathcal{C}$. The above equation leverages the fact that $\text{Card}\{j : \mathbf{x}_j \in \mathcal{C}\}/N$ can be used as an estimator of the area of $\mathcal{C}$. The star discrepancy is thus given by the largest error of the estimate of $\lambda(\mathcal{C})$ over all the boxes $\mathcal{C} \in B$. For more details on the star discrepancy we refer the interested reader to [PH10] (Chapter 7, page 360). The connection between star discrepancy and the worst case error

$$\sup_f \left| \frac{1}{N} \sum_{j=1}^{N} f(\mathbf{x}_j) - \int_{[0,1[^2} f(\mathbf{x}) d\mathbf{x} \right|$$

is given by the Koksma-Hlawka inequality [Nie88, BD11]. Each point set construction entails a certain order of convergence of the star discrepancy toward 0. When the convergence rate of the star discrepancy toward 0 is of order $\mathcal{O}(N^{-1}(\log N)^2)$, the point set construction is called *low discrepancy*.

2.2 QMC on the unit sphere

Unlike the unit square case, QMC rules for numerical integration over the unit sphere $\mathbb{S}^2$ in $\mathbb{R}^3$ are less known to the graphics community. Therefore, a brief presentation of important results on this subject are given in this section.

A set of sampling directions $\{\boldsymbol{\omega}_{1,N}, \ldots, \boldsymbol{\omega}_{N,N}\}$ defined as points on the unit sphere $\mathbb{S}^2$ is appropriate for Monte Carlo integration if it is *asymptotically uniformly distributed*, that is if

$$\lim_{N \to \infty} \frac{1}{N} \sum_{j=1}^{N} f(\boldsymbol{\omega}_{j,N}) = \frac{1}{4\pi} \int_{\mathbb{S}^2} f(\boldsymbol{\omega}) d\Omega(\boldsymbol{\omega}) \tag{2.1}$$

is true for every function $f(\boldsymbol{\omega})$ on the sphere $\mathbb{S}^2$, Ω being the surface measure on $\mathbb{S}^2$. Similarly to the unit square case, this property is equivalent to:

$$\lim_{N\to\infty} \frac{\text{Card}\{j : \boldsymbol{\omega}_{j,N} \in \mathcal{C}\}}{N} = \frac{\Omega(\mathcal{C})}{4\pi} \tag{2.2}$$

for every spherical cap $\mathcal{C}$ with area $\Omega(\mathcal{C})$ [KN06]. Informally speaking, Eq. (2.2) means that a spherical cap of any area has its fair share of points as $N \to \infty$. Among all sampling patterns complying with this definition, we are interested in point sets $P_s = \{\boldsymbol{\omega}_{1,N}, \ldots, \boldsymbol{\omega}_{N,N}\} \subseteq \mathbb{S}^2$ such that the worst case integration error (w.c.e.)

$$\text{w.c.e.} := e(P_s) = \sup_f \left| \frac{1}{N} \sum_{j=1}^{N} f(\boldsymbol{\omega}_{j,N}) - \frac{1}{4\pi} \int_{\mathbb{S}^2} f(\boldsymbol{\omega}) d\Omega(\boldsymbol{\omega}) \right|$$

achieves the best rate of convergence as $N \to \infty$. This is equivalent to finding the point sets P_s which minimize the spherical cap L_2 discrepancy [BSSW12]:

$$L_2(P_s) = \left(\int_{-1}^{1} \int_{\mathbb{S}^2} \left| \frac{\text{Card}\{j : \boldsymbol{\omega}_{j,N} \in \mathcal{C}(\boldsymbol{\omega}, t)\}}{N} - \frac{\Omega(\mathcal{C}(\boldsymbol{\omega}, t))}{4\pi} \right|^2 d\Omega(\boldsymbol{\omega}) dt \right)^{\frac{1}{2}}$$

where $\mathcal{C}(\boldsymbol{\omega}, t)$ is a spherical cap centered in $\boldsymbol{\omega}$ and containing all the points $\{\boldsymbol{\omega}_c \in \mathbb{S}^2 : \boldsymbol{\omega}_c \cdot \boldsymbol{\omega} \leq t\}$. The L_2 discrepancy of a point set P_s is connected to the sum of the distances between all the points in P_S through the Stolarsky invariance principle [BD11]. This invariance states that

$$\frac{1}{N^2} \sum_{j=1}^{N} \sum_{i=1}^{N} |\boldsymbol{\omega}_i - \boldsymbol{\omega}_j| + 4\, L_2(P_s) = \frac{4}{3}\,.$$

The above invariance shows that maximizing the sum of distances term $\sum_{j=1}^{N} \sum_{i=1}^{N} |\boldsymbol{\omega}_i - \boldsymbol{\omega}_j|$ is equivalent to minimizing the L_2 discrepancy, with the advantage of being a much simpler operation. We can thus define a criterion E_N to evaluate the quality of the distribution of a spherical point set:

$$E_N(P_s) = \left(\frac{4}{3} - \frac{1}{N^2} \sum_{j=1}^{N} \sum_{i=1}^{N} |\boldsymbol{\omega}_i - \boldsymbol{\omega}_j| \right)^{\frac{1}{2}}, \tag{2.3}$$

which will be referred to as energy.

As stated above, minimizing the L_2 discrepancy is equivalent to minimizing the w.c.e. But both criteria only follow the same $\mathcal{O}(N^{-3/4})$ rate of convergence toward 0 if the integrand f fulfills a specific smoothness criterion. Roughly speaking, f must be at least a C_0 continuous function. In such a case, the w.c.e. is equal to the energy E_N [BD11, BSSW12].

The order of convergence of the w.c.e. can be higher than $\mathcal{O}(N^{-3/4})$ if the order of continuity of the integrand is higher than C_0. But achieving such a convergence rate depends on the points construction algorithm since some are more capable of taking advantage of smooth functions than others [BSSW12].

The order of convergence of the L_2 discrepancy, on the other hand, cannot be better than $\mathcal{O}(N^{-3/4})$. But there surely exist point sets for which the order of convergence of the L_2 discrepancy is better than $\mathcal{O}(N^{-3/4}\sqrt{\log N})$ [Bec84]. In this case these point set constructions are said to be *low-discrepancy sequences*. Note that this order of convergence is lower than the $\mathcal{O}(N^{-1}(\log N)^2)$ rate of low-discrepancy sequences in the $[0,1]^2$ unit square.

In contrast with the unit square case, no explicit direct construction of low discrepancy sequences on the unit sphere is known. That is why QMC sequences on $\mathbb{S}^2$ are generally produced by lifting a $[0,1]^2$ low discrepancy point set to $\mathbb{S}^2$ through an equal-area transform. An alternative to this approach consists in generating the patterns directly on the sphere according to an extremal energy criterion [SK97]. Among the patterns with good E_N properties, spherical Fibonacci point sets are particularly well-suited to QMC integration over the sphere as shown in [HN04], hence our interest in applying them to illumination integral computation.

2.3 QMC point sets

In this section we present some of the most commonly used point sets for QMC. In particular, we describe the characteristics of point sets generated using blue noise [dGBOD12], Larcher-Pillichshammer [LP01] and the Sobol (0,2)-sequence [Sob67]. The Larcher-Pillichshammer points are known for having very good (unit square) star discrepancy properties even when compared to other low discrepancy point sets [Ken13]. The Sobol $(0,2)$-sequence is probably the most used low discrepancy point set and is particularly well suited to adaptive sampling schemes as will be seen later. The blue noise generators allow producing high quality uniform distributions. Stemming from a very active research field, their generation uses a totally different approach to that of the Sobol and Larcher-Pillichshammer points.

In Fig. 2.1, examples of different unit square point sets are shown. They have been generated using a random distribution, blue noise, Larcher-Pillichshammer and a Sobol sequence respectively, from left to right.

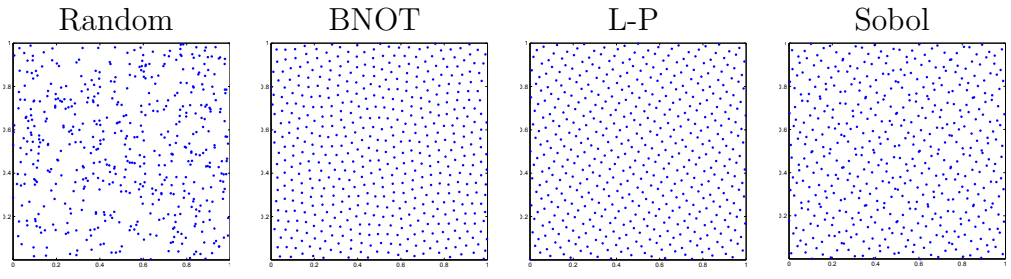


Figure 2.1 – A comparison of a set of 512 points randomly generated, and three other sets of 512 points generated using blue noise (BNOT), the Larcher-Pillichshammer points, and the Sobol $(0,2)$-sequence.

2.3.1 Blue noise

The blue noise point sets are distributions of points designed such that a minimum inter-sample distance criterion is met. Moreover, the distribution should be free from regular patterns [dGBOD12, LD08]. At first sight, it might seem that we would like the minimum inter-sample distance to be as large as possible but, as noted by Lagae and Dutré [LD08], a large minimum distance may result in regular patterns. Thus, the goal is to obtain regularity-free distribution with the largest average inter-sample distance.

To evaluate the quality of the blue noise distributions it is common to resort to a spectral analysis. We have mentioned above that the two features which characterize a blue noise distribution are a large average inter-sample distance and no regularities. In the frequency domain, on the other hand, a blue noise spectrum is characterized by a minimal low frequency energy and by the lack of concentrated energy spikes.

The power spectrum associated with a generator of point sets can be estimated by averaging periodograms of different point sets produced using that generator. As opposed to the random distributions, whose power spectrum is flat, a blue noise pattern produces a high frequency error which is less conspicuous to the human eye. Examples of state of the art blue noise generators can be found in [Fat11, EPM+11, CYC+12, dGBOD12].

2.3.2 Larcher-Pillichshammer points

The Larcher-Pillichshammer points [LP01] are constructions of $(0, m, 2)$-nets in base 2 which are a restriction of the general (t, m, s)-nets. Formally, a (t, m, s)-net in base q is defined as follows [Nie92].

Definition. *A (t, m, s)-net in base q is a set of q^m points in $[0, 1[^s$ enjoying the property that each elementary interval E in base q*

$$E = \prod_{i=1}^{s} \left[\frac{a_i}{q^{b_i}}, \frac{a_i + 1}{q^{b_i}} \right[, \subseteq [0, 1[^s$$

contains exactly q^t points, with $a_i, b_i \in \mathbb{Z}$, $b_i \geq 0$ and $0 \leq a_i < q^{b_i}$. $\prod$ is the cartesian product of sets (or product set). The volume of each elementary interval E is q^{t-m}.

The parameter t works as a kind of quality parameter which controls the number of points per elementary interval. As a $(0, m, 2)$-net in base 2, the Larcher-Pillichshammer's approach allows generating bi-dimensional sets of 2^m points containing a single point per elementary interval. This is a very interesting stratification property. Let us take the example in which $m = 4$. Fig. 2.2 shows a $(0, 4, 2)$-net based on the Larcher-Pillichshammer points, as well as the points distribution over the elementary intervals. It can be seen that the 16 elements fulfill the Latin hypercube stratification condition. This means that if we divide the unit square in 16 horizontal or vertical elementary intervals of size $\left(1, \frac{1}{16}\right)$ and $\left(\frac{1}{16}, 1\right)$ respectively, each line and each column only contain a single sample (Fig. 2.2, right). Furthermore, if we divide

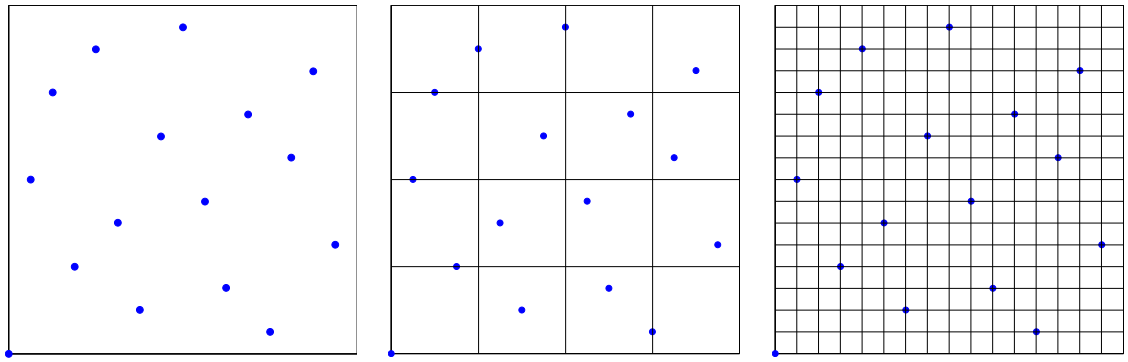


Figure 2.2 – The Larcher-Pillichshammer points.

the unit square in 16 squared elementary intervals of size $\left(\frac{1}{4}, \frac{1}{4}\right)$ there is also a single sample per square (Fig. 2.2, center).

Although these stratification properties are interesting, they do not necessarily guarantee a good distribution of points. Two points in neighboring elementary intervals could be very close to each other, which would increase the discrepancy of the point set. But this problem does not occur when using the Larcher-Pillichshammer points and they were shown to have a comparatively large minimum inter-sample distance [GK08]. For more detail on how to generate these point sets, please refer to [LP01].

2.3.3 Sobol $(0, 2)$-sequence

Introduced by the Russian mathematician I. M. Sobol in [Sob67], the Sobol $(0, 2)$-sequence is a particular case of the general (t, s)-*sequences*. The (t, s)-sequences are defined as follows [Nie92].

Definition. *Let $t \geq 0$ be an integer. A sequence of points $x_0, x_1, \ldots$ in $[0, 1[^s$ is a (t, s)-sequence in base q if, for all integers $k \geq 0$ and $m > t$, the point set composed of the $x_{kq^m}, \ldots, x_{(k+1)q^m-1}$, is a (t, m, s)-net in base q.*

These sequences are specially useful for adaptive sampling schemes. Adaptive sampling consists in generating a first set of samples and, based on some criterion, decide whether or not to add more samples to the initial set. In case the decision is for adding more samples, the samples set resulting from the combination of the first and second samples sets should also be low discrepancy. This roughly means that the second set of samples cannot generate new samples which are too close to the samples of the first set.

The definition of $(0, 2)$-sequences in base 2 (such as the Sobol $(0, 2)$-sequence) implies that each successive set of 2^m points is a $(0, m, 2)$-net. Moreover, the combination of a set of 2^m points $x_0, \ldots, x_{2^m-1}$ with the next set of 2^m points of the sequence forms a $(0, m + 1, 2)$-net. Such a property is interesting because it allows hierarchical or adaptive sampling schemes as described above. The same cannot be done with the point set generators which strictly produce (t, m, s)-nets and not (t, s)-sequences (e.g., the Larcher-Pillichshammer points). Using the Sobol $(0, 2)$-sequence instead of the Larcher-Pillichshammer points, however, is not advantageous in every respect since the latter exhibit lower discrepancy properties.

2.4 Hemispherical projections

QMC spherical point sets are usually generated on the unit square and then projected onto the unit sphere. The used projection must be an equal-area projection so that an area in the unit square corresponds to the same area in the unit sphere. This feature guarantees that an asymptotically uniformly distributed point set on the unit square, will also be asymptotically uniformly distributed on the sphere after the projection. Note that, despite their area-preserving property, these projections cause a distortion and do not preserve the inter-sample distance. The degree and the location of this distortion can vary from one projection to another.

In the following of this section we will present the most common hemispherical projections used in computer graphics: Lambert cylindrical [DHM+01], concentric maps [SC97] and HEALPix [GHB+05].

2.4.1 Lambert cylindrical projection

Probably due to its simplicity, the Lambert cylindrical projection, also called Archimedes projection, is the most common hemispherical projection used in computer graphics [DHM+01]. Using this projection, a given point (x, y) on the unit square is projected onto the hemisphere as follows:

$$\begin{aligned} \phi &= 2\pi x \\ \theta &= \arccos(1 - 2y) \end{aligned}$$

where (θ, ϕ) are the spherical coordinates of the projected point. Although it has no distortion on the equator, the distortion of this projection increases as we approach the pole. At the pole, there is a singularity. All points in the unit square with $y = 1$ are mapped to a single point on the hemisphere (the pole, with $\theta = 0$).

2.4.2 Concentric maps

The concentric maps [SC97] is an equal area projection which allows mapping concentric squares to concentric circles on the hemisphere. The goal of this projection is to provide a lower distortion than the Lambert map by leveraging the fact that our interest is restricted to the hemisphere. Roughly speaking, using this projection close points on the unit square will also tend to be close points on the hemisphere. To achieve this, the unit square points are first mapped onto a unit disk as depicted in Fig. 2.3. It is only after this first projection that the points are mapped to the unit hemisphere.

Given a point with coordinates (x, y) on the unit square corresponding to the region 1 in Fig. 2.3 (left), its hemispherical projection is given by:

$$(x, y) \longrightarrow \begin{array}{l} a = 2x - 1 \\ b = 2y - 1 \end{array} \longrightarrow \begin{array}{lcl} r & = & a \\ \phi_d & = & \frac{\pi a}{4b} \end{array} \longrightarrow \begin{array}{l} \theta = \arcsin\left(r\sqrt{2 - r^2}\right) \\ \phi = \phi_d \end{array}$$

where (a, b) are the coordinates of a point in a unit square centered at $(0, 0)$. (r, ϕ_d) are, respectively, the radius and the rotation angle of a point on the unit disk. Finally

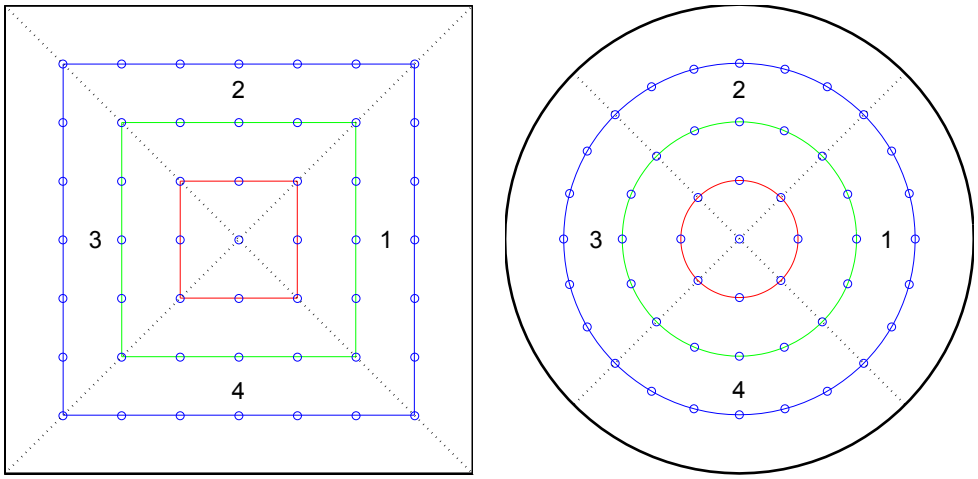


Figure 2.3 – The concentric maps of Shirley and Chiu [SC97].

(θ, ϕ) are the spherical coordinates of the projected points. For the other three regions similar transforms are applied.

2.4.3 HEALPix projection

The HEALPix projection [GHB+05] was originally developed to address processing and analysis needs of huge spherical data sets generated by advanced detectors in modern astronomy. The projection meets several criteria required for the intended application among which the need for an equal-area projection. This makes the HEALPix projection suitable for spherical Monte Carlo integration.

The hemispherical HEALPix projection onto the plane can be appreciated in Fig. 2.4. Since the HEALPix projection has as primary goal to project points from the sphere to the plane (and not the reverse), all the points of the sphere have correspondence on the plane, but there are areas of the plane with no correspondence on the sphere. The same is true for the hemisphere. The zones of the unit square with no correspondence

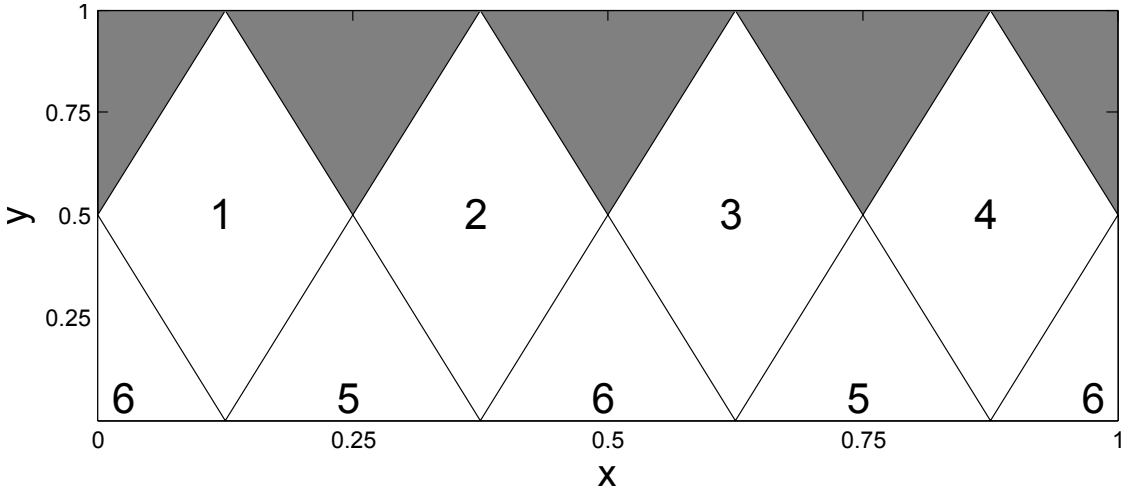


Figure 2.4 – HEALPix hemispherical projection [GHB+05].

on the hemisphere are marked in grey in Fig. 2.4. One of the solutions to project a

set of points onto the hemisphere using HEALPix is to generate six sets of points on the unit square. The six point sets can then be mapped onto the numbered regions in Fig. 2.4. It is also possible to use a single square and place it repeatedly in each of the numbered regions.

An alternative approach is to generate a single set of points on the plane and reject those which are outside the projectable region. We have experimentally found that this solution results in visually less pleasant distributions than using the approach described above.

The mapping of a point (x, y), $y \leq 1/2$ on the unit square to a point (θ, ϕ) on the unit hemisphere is given by:

$$\begin{aligned} \theta &= \arccos\left(\tfrac{4\pi}{3} y\right) \\ \phi &= 2\pi x \end{aligned}$$

while for a point on the unit square with $y > 1/2$, the projection is given by:

$$\begin{aligned} \theta &= \arccos\left(1 - \tfrac{1}{3}(2-2y)^2\right) \\ \phi &= 2\pi x - \tfrac{y-1/2}{y-1}\left(x_t - \tfrac{\pi}{4}\right), \end{aligned}$$

where

$$x_t = 2\pi x \times \mathrm{mod}\ \frac{\pi}{2}\,.$$

Furthermore, for $y > \frac{\pi}{4}$, x_t must satisfy the condition

$$\left| x_t - \frac{\pi}{4} \right| < \frac{\pi}{2}(1-y)$$

so as to guarantee that the point is contained in the projectable area shown in white in Fig. 2.4.

2.5 Summary

The use of low discrepancy sequences is widespread in computer graphics [KPR12]. Their goal is to improve the convergence rate of the QMC integral estimate by using sample sets which minimize a discrepancy criterion. Among the most popular low-discrepancy sequences, the Sobol's (0,2)-sequence [Sob67] guarantees both the minimum distance and the stratification criteria in each successive sets of q^m samples, where q is the base of the sequence. Lower unit square discrepancy values can be obtained using the Larcher-Pillichshammer point sets [LP01], but these points cannot be generated by an (infinite) sequence. Moreover, Kollig and Keller [KK02] showed that both Sobol and Larcher-Pillichshammer point sets can be easily scrambled to decorrelate directions for neighboring pixels, thus avoiding artifacts without sacrificing the discrepancy and stratification properties.

An alternative approach for producing uniform point set distributions on a unit square is to use a blue noise generator [LD08]. This class of point set generators produces high quality uniform (yet unstructured) distributions which try to approach

the spectral characteristics of Poisson disk distributions. The goal is to concentrate the noise in high frequencies where it is less visible. The resulting distributions exhibit better uniformity properties when compared to (0,2)-sequences, but this is achieved at a higher computational cost. Recent works have focused on efficient generation of high quality blue noise patterns [CYC+12, dGBOD12, EPM+11, Fat11]. The state-of-the-art solution is currently given by [dGBOD12].

The unit square-based distributions generated by the methods described above must be lifted to the $\mathbb{S}^2$ sphere using an equal-area projection so as to be used for (hemi)spherical integration. Such projections preserve the property of asymptotic distribution uniformity, but not the samples distance. As discrepancy and w.c.e. directly depend on the distance between samples (see Eq. (2.3)), the resulting sets become suboptimal for (hemi)spherical sampling.

An explicit spherical construction of point sets with small discrepancy has been proposed by [LPS86], but recently a better order of convergence has been reported by using spherical Fibonacci point sets [Nye03, Gon10, ABD12]. Throughout this chapter, we will use a spherical Fibonacci point set based on the work of [SP06]. We will show that constructing these sets is simpler and more efficient than the Sobol (0,2)-sequence [Sob67], the state of the art blue noise [dGBOD12] and the Larcher-Pillichshammer point sets [LP01] when the goal is to evaluate the illumination integral.

3 Spherical Fibonacci point sets

Our goal in this section is to explain how Fibonacci lattices are generated and why such point constructions are well-suited to spherical sampling. A spherical Fibonacci (SF) lattice is defined as follows [Sve94, HN04]:

$$\begin{aligned} \theta_j &= \arccos(1 - 2j/F_m) \\ \phi_j &= 2\pi \left\{ j \frac{F_{m-1}}{F_m} \right\} \end{aligned}$$

where θ_j and ϕ_j are the polar and azimuthal angles respectively of a lattice node $\boldsymbol{\omega}_j$. As the Fibonacci ratio F_m/F_{m-1} quickly approaches the golden ratio $\Phi = (1+\sqrt{5})/2$ as m increases [GKP94], we can write:

$$\lim_{m\to\infty} \phi_j = 2j\pi\Phi^{-1}$$

due to the periodicity of the spherical coordinates. Hence, setting $F_m = N$, the coordinates of an N-point spherical Fibonacci set are given by:

$$\left.\begin{aligned} \theta_j &= \arccos(1 - 2j/N) \\ \phi_j &= 2j\pi\Phi^{-1} \end{aligned}\right\} 0 \le j < N\,.$$

Note that in this case, N needs not be a Fibonacci number anymore, which allows generating point sets with an arbitrary number of points. But the resulting point sets are no longer lattices since Φ is irrational. Therefore, from now on, these point sets

will be called spherical Fibonacci point sets (SF). Letting z_j denote the z coordinate of point j, we have:

$$z_j = \cos\theta_j = 1 - 2j/N$$

which means that the z coordinates of the points are evenly spaced. Such an arrangement divides the sphere into equal-area spherical 'rings' due to the area-preserving property of the Lambert map [Gon10], each 'ring' containing a single lattice node. Swinbank and Pusser [SP06] slightly improved the point set used in [HN04] by introducing an offset of $1/N$ to the z_j coordinates (i.e. half the z coordinate spacing) to achieve a more uniform distribution near the poles. Then we have:

$$\left.\begin{array}{rcl} \theta_j & = & \arccos\left(1 - \frac{2j+1}{N}\right) \\ \phi_j & = & 2j\pi\Phi^{-1} \end{array}\right\} 0 \leq j < N\,. \tag{2.4}$$

As observed by [Gon10], the same point set can be produced using $\Phi^{-2} = (3-\sqrt{5})/2$ instead of Φ^{-1}. The ϕ_j angles will then be multiples of the golden angle $\pi(3-\sqrt{5})$. For more details on the properties of the spherical Fibonacci point set we refer the reader to [SP06, Gon10]. Note that this arrangement of points can also be found in nature (e.g. the packing of seeds on the sunflowers head [Vog79]), a clear indication of its near-optimality w.r.t. the distance based energy metric E_N (Eq. (2.3)). Other theoretical approaches proposed in the literature lead to similar arrangements (e.g. [SK97]).

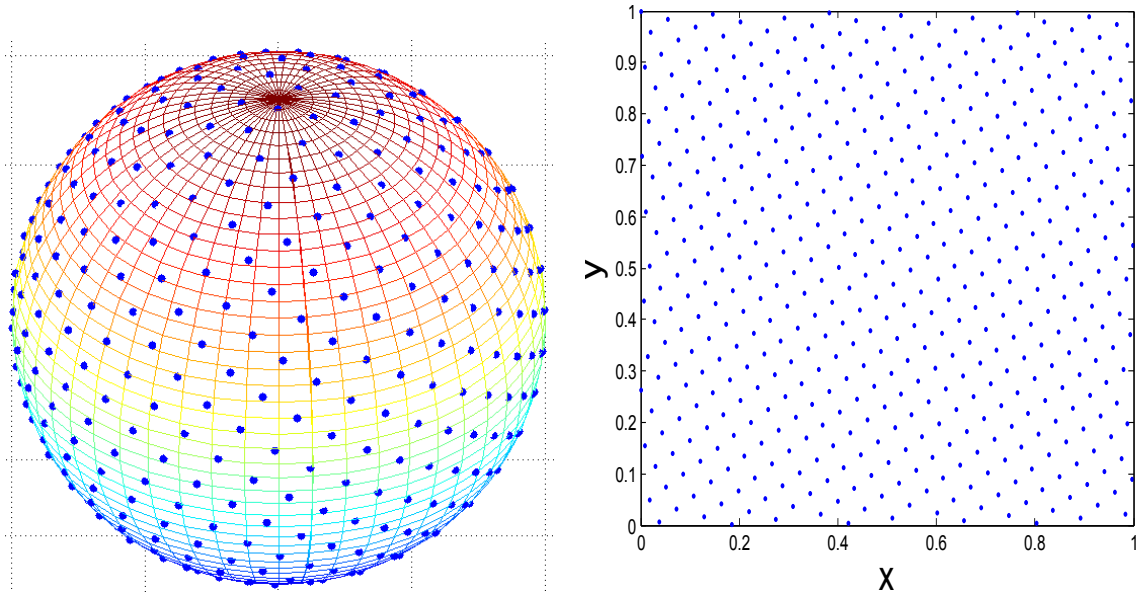
An example of a spherical Fibonacci point set is shown in Fig. 2.5(a). In Fig. 2.5(b), we can see the projection of the same point set onto the unit square $[0,1]^2$, using the Lambert equal-area cylindrical projection. The spirals on the sphere become parallel oblique lines on the unit square projection. This arrangement shows that each point has a different latitude, which allows a fairly good point distribution near the poles compared to other methods as will be seen hereafter. However, no periodicities or near-periodicities appear in this point set because the golden ratio is the most irrational number, that is, the hardest to approximate with a rational number.

In the case of illumination integrals (see Eq. (2.5)), the integration domain is not the sphere, but the hemisphere $\Omega_{2\pi}$, where the vertical axis z is aligned with the surface normal. By modifying Eq. (2.4), an N-point hemispherical SF point set will then be defined as follows:

$$\left.\begin{array}{rcl} \theta_j & = & \arccos(z_j) \\ \phi_j & = & 2j\pi\Phi^{-1} \end{array}\right\} 0 \leq j < N$$

where the $z_j = \left(1 - \frac{2j+1}{2N}\right)$ are the z coordinates of the points on the hemisphere. Such a point set can be very easily generated using the pseudo-code presented in Alg. 1.

Image synthesis involves the computation of many illumination integrals. Using the same point set for computing all illumination integrals results in visible patterns in the rendered images. To avoid this problem the sample sets must be scrambled at each illumination integral evaluation. We used a scrambling strategy of the SF sampling pattern which is made directly in the spherical domain by rotating them about the z axis with a random angle uniformly distributed over $[0, 2\pi]$. This method has the advantage of preserving the inter-sample distances and thus the energy E_N.



(a) (b)

Figure 2.5 – Example of Fibonacci lattice ($N = 512$).

Algorithm 1 The spherical Fibonacci point set algorithm.

$\Delta\phi \leftarrow \pi(3 - \sqrt{5})$ ▷ Golden angle (step on ϕ)
$\phi \leftarrow 0$ ▷ Initialize ϕ
$\Delta z \leftarrow 1/n$ ▷ Compute the step on z
$z \leftarrow 1 - \Delta z/2$ ▷ Initialize z with offset
for all $j \leftarrow [1 : n]$ **do**
  $z_j \leftarrow z$
  $\theta_j \leftarrow \arccos(z_j)$
  $\phi_j \leftarrow \text{mod } (\phi, 2\pi)$ ▷ Modulo of ϕ
  $z \leftarrow z - \Delta z$ ▷ Give a step on z
  $\phi \leftarrow \phi + \Delta\phi$ ▷ Give a step on ϕ
end for

4 QMC for illumination integrals

To render an image of a scene, the illumination integral must be computed at each point of visible surfaces. Let us recall that, as shown in Chapter 1, this integral gives the reflected radiance $L_o(\boldsymbol{\omega}_o)$ at a given visible point. For a non self-emitting surface it can be expressed as follows:

$$L_o(\boldsymbol{\omega}_o) = \int_{\Omega_{2\pi}} L_i(\boldsymbol{\omega}_i)\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)(\boldsymbol{\omega}_i \cdot \mathbf{n})\, d\Omega(\boldsymbol{\omega}_i)\,. \tag{2.5}$$

A straightforward application of Eq. (2.1) would consist in computing an estimate of $L_o(\boldsymbol{\omega}_o)$ by averaging samples of the integrand of Eq. (2.5) with uniformly-distributed sampling points on $\Omega_{2\pi}$. Such an approach would be quite inefficient since the product

$\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)(\boldsymbol{\omega}_i \cdot \mathbf{n})$ is generally close to zero in a large part of the integration domain. In classic Monte Carlo methods, a common solution is to distribute the samples according to a pdf proportional to $\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)(\boldsymbol{\omega}_i \cdot \mathbf{n})$. In the QMC deterministic context, as probabilistic distributions cannot be used, instead this function is moved into the integration variables through an appropriate variable substitution. In the following, we will show how to reformulate the problem of optimally sampling $\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)(\boldsymbol{\omega}_i \cdot \mathbf{n})$ in the context of QMC integration, starting from a uniform point set distribution.

Eq. (2.5) can be developed as follows:

$$L_o = \int_0^{2\pi} \int_0^{\pi/2} L_i(\theta, \phi)\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) \cos\theta \sin\theta \, d\theta \, d\phi \tag{2.6}$$

where θ and ϕ are the spherical coordinates of the incident direction $\boldsymbol{\omega}_i$ w.r.t. the z axis.

The Phong glossy BRDF is given by:

$$\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) = k\frac{(\max[0, (\boldsymbol{\omega}_i \cdot \boldsymbol{\omega}_r)])^n}{\boldsymbol{\omega}_i \cdot \mathbf{n}},$$

where $\boldsymbol{\omega}_r = 2(\boldsymbol{\omega}_o \cdot \mathbf{n}) - \boldsymbol{\omega}_o$ is the perfect mirror incident direction. A diffuse BRDF can be seen as a special case for which $\boldsymbol{\omega}_r = \mathbf{n}$ and $n = 1$ (its albedo is then πk).

Considering that the incident radiance function is zero for incident directions below the tangent plane (i.e. $L_i(\boldsymbol{\omega}_i) = 0$ if $(\boldsymbol{\omega}_i \cdot \mathbf{n}) < 0$), we can take the hemisphere $\Omega_{2\pi}^{(r)}$ centered about $\boldsymbol{\omega}_r$ as the integration domain. Our coordinate frame will then be rotated such that its z axis is aligned with $\boldsymbol{\omega}_r$ and therefore, the azimuthal polar θ of a point $\boldsymbol{\omega}$ on $\Omega_{2\pi}^{(r)}$ will be defined by $\theta = \arccos(z)$ with $z = (\boldsymbol{\omega} \cdot \boldsymbol{\omega}_r)$. Consequently, by making the variable substitution $z = \cos\theta$, Eq. (2.6) can be written as follows:

$$L_o(\boldsymbol{\omega}_o) = k \int_0^{2\pi} \int_0^1 L_i(z, \phi) z^n \, dz \, d\phi \, .$$

Making the substitution $z' = z^{n+1}$, we have:

$$L_o(\boldsymbol{\omega}_o) = \frac{k}{n+1} \int_0^{2\pi} \int_0^1 L_i'(z', \phi) \, dz' \, d\phi$$

where $L_i'(z', \phi) = L_i\left((z')^{1/(n+1)}, \phi\right)$. Since the integral bounds still define a hemispherical integration domain, an estimate of $L_o(\boldsymbol{\omega}_o)$ is obtained using Eq. (2.1):

$$\tilde{L}_o(\boldsymbol{\omega}_o) = \frac{2\pi k}{N(n+1)} \sum_{j=1}^{N} L_i(z_j^{1/(n+1)}, \phi_j) \tag{2.7}$$

where (z_j, ϕ_j) are the coordinates of a uniformly-distributed samples set P_N on $\Omega_{2\pi}^{(r)}$. Eq. (2.7) means that the incident radiance function L_i is sampled with a sampling pattern obtained by morphing the z coordinates of the samples of the uniformly-distributed set P_N with the function $f(z) = z^{1/(n+1)}$.

To sum up, the above derivations show how to use a spherical uniform point set to compute an approximation of the illumination integral while taking into account the BRDF shape. Although the original samples set P_N undergoes a morphing operation, the w.c.e. of the estimate given by Eq. (2.7) is still strongly dependent on the characteristics of P_N (and in particular on the energy E_N), as will be seen in the following sections.

5 Tested point sets

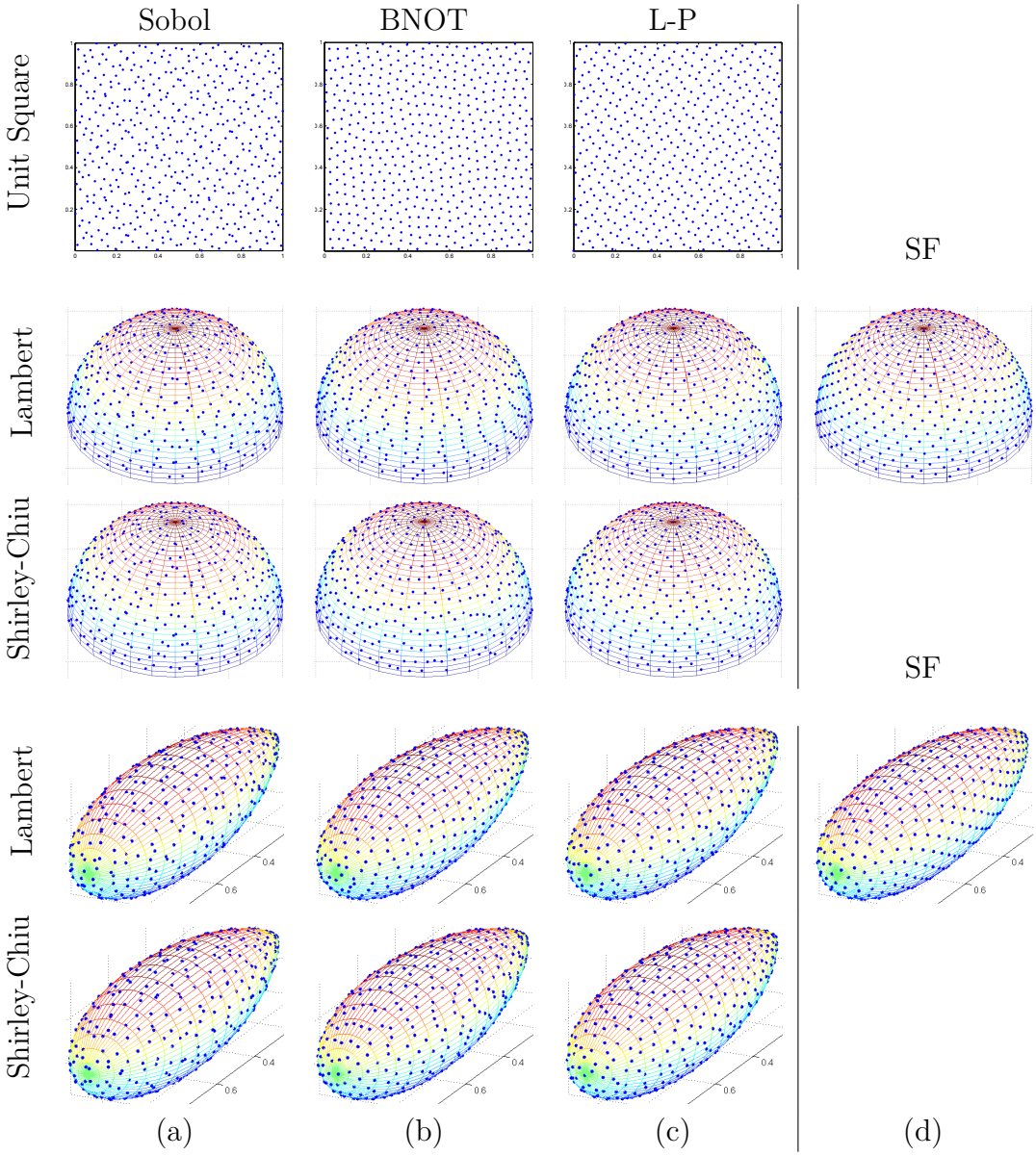


Figure 2.6 – Examples of point sets of size 512 produced by different algorithms. Top row: unit square projection. Second row: Lambert cylindrical projection. Third row: Shirley-Chiu [SC97] concentric maps projection and, on the right, the Fibonacci point set generated directly in the spherical domain. The last two rows show the spherical point sets after morphing according to a Phong lobe with a shininess value of 20.

In this section, our goal is to compare the properties of the presented spherical Fibonacci point sets with those of the sample sets produced by the following algorithms:

- Sobol $(0, 2)$-sequence with random digit scrambling as described by [KK02];
- Periodic blue noise, generated with the state of the art algorithm of de Goes et al. [dGBOD12];

- Larcher-Pillichshammer points [LP01] with random digit scrambling as described by [KK02].

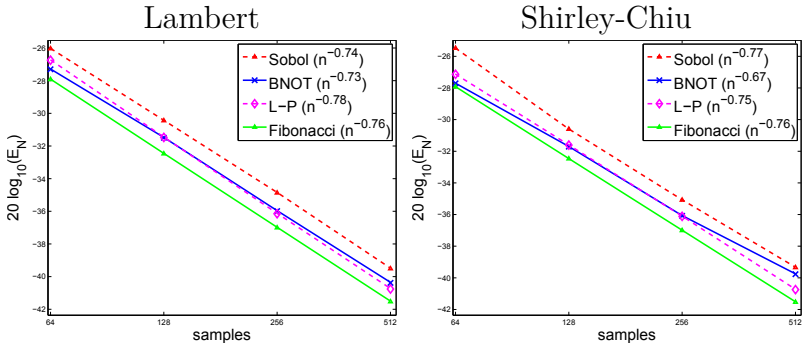
Henceforth, we will refer to these three algorithms as Sobol, BNOT and L-P respectively. Fig. 2.6 shows different projections of sets of 512 samples generated using Sobol, BNOT and L-P, as well as an example of a spherical Fibonacci point set. We used two different techniques for projecting the unit square point sets to the unit hemisphere: the Lambert cylindrical projection (e.g., see [ABD12]) and the concentric maps of Shirley and Chiu [SC97] presented in section 2. In this section we do not consider the HEALPix projection because experiments we have made showed that it provides results of lower quality compared to the other projections. Note that these projections do not apply to Fibonacci point sets since these latter are generated directly on the sphere. The pattern generated by the Sobol sequence is apparently non-optimal in terms of discrepancy, since the distance of a sample to its closest neighbour is quite variable. This can be observed both on the unit square and on the unit hemisphere projections. On the other hand, the BNOT and L-P sampling patterns (Fig. 2.6(b) and (c) respectively) seem to be more uniformly distributed than the Sobol sequence. As for the spherical Fibonacci point set (Fig. 2.6(d)), it exhibits superior uniformity properties when compared to all the other point sets projected on the hemisphere.

A quantitative analysis of these visual impressions can be made by comparing the different sampling patterns in terms of the energy metric defined in Eq. (2.3). Let us recall that, as stated in section 2, the w.c.e. is proportional to energy under a C_0 continuity assumption for the integrand. Fig. 2.7(a) clearly illustrates that the Fibonacci point set exhibits a lower energy (Eq. (2.3)) than the other tested algorithms and is thus expected to yield a lower w.c.e. value. In the same line of results, Fig. 2.7(b) shows that the minimum inter-sample distance is consistently larger for the Fibonacci point sets, which is an indication of better uniformity properties. All the tested point sets (except for BNOT using the Shirley-Chiu projection) yield approximately the same $\mathcal{O}(N^{-3/4})$ rate of decay for E_N, which corresponds to the optimum rate of convergence for the w.c.e., as explained in section 2. Recall that this convergence rate is obtained under a C_0 continuity assumption of the integrand, which is in general not fulfilled for illumination integrals. Nevertheless, as will be seen in the next section, these inconsistencies have marginal effects. In particular, we will show experimentally that the accuracy of the estimates given by Eq. (2.7) strongly depends on the energy E_N of the uniformly-distributed samples set P_N.

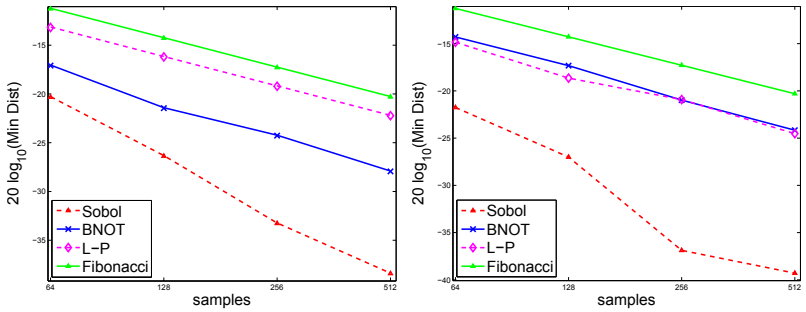
6 Results

6.1 General considerations

The results presented in this section have been generated with the Mitsuba raytracer [Jak10] on a 64-bit machine equipped with a 2GHz Intel Core i7 processor and a 8-Gb RAM. Three different scenes have been used: Cornell Box (185K triangles), Room (540K triangles) and Cars (1500K triangles). The illumination integral computation



(a) Energy.



(b) Minimum inter-sample distance.

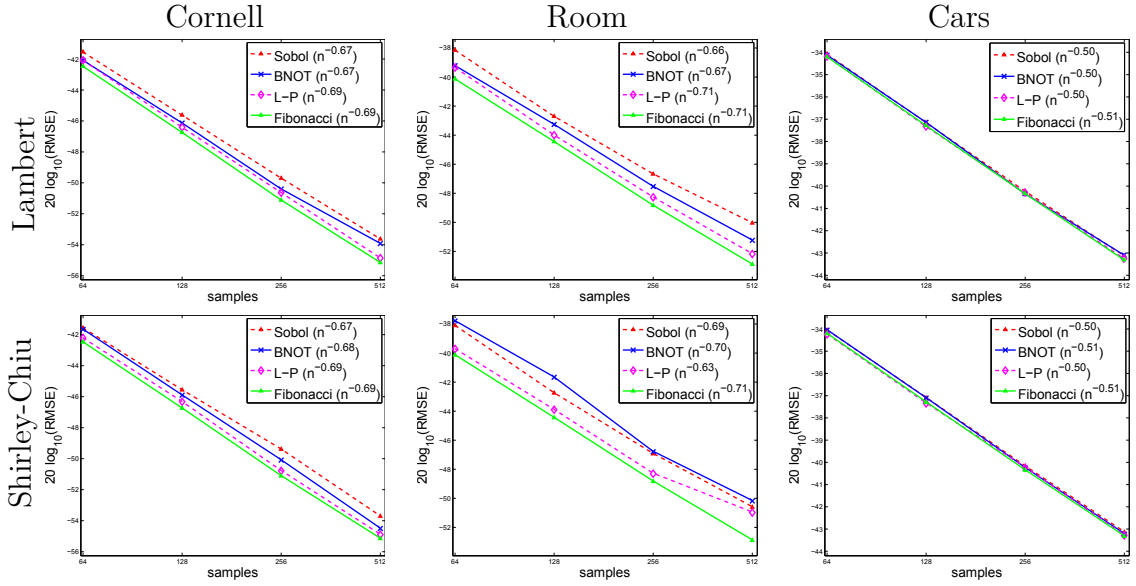
Figure 2.7 – Properties of the tested point sets. For each metrics (energy and minimum distance), the same point set was projected using the Lambert (left) and Shirley-Chiu [SC97] (right) projections, except for the Fibonacci point set which is generated directly in the spherical domain.

has been performed in the context of final gathering for photon mapping, using the estimator given in section 4, Eq. (2.7). We have compared the results produced using the different point set construction strategies which have been presented in section 5. A reference image has been computed using a sampling pattern produced by a Sobol sequence and a large number of samples until convergence was achieved. This reference image was then used to evaluate the Root Mean Square Error (RMSE) of the images produced with the different point sets. For SF and BNOT scrambling is performed on the sphere (as described in section 3), but for L-P and Sobol sampling patterns it is made on the plane according to the random digit scrambling method proposed by [KK02].

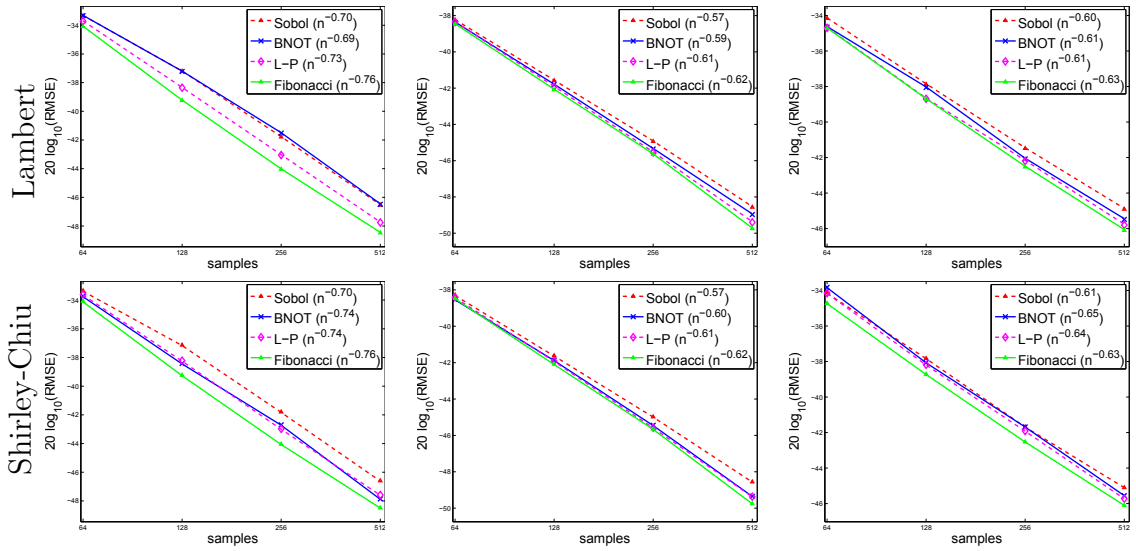
6.2 RMSE analysis and convergence slope

Fig. 2.8 shows that for the same number of samples, the spherical Fibonacci point sets yield consistently smaller RMSE values than the other tested methods. Indeed, we have not registered any case where the Fibonacci lattices have been outperformed in terms of RMSE value. The convergence slope of QMC methods depends on the smoothness properties of the integrands. Therefore, it is not guaranteed that the theoretical convergence rate for the w.c.e. ($\mathcal{O}(N^{-3/4})$ for C_0 continuous functions) can be obtained for highly discontinuous integrands, such as those commonly met in illumi-

nation integrals. Nevertheless, in the Cornell Box scene and in the diffuse component of the Room scene, it was possible to report convergence rates close to the theoretical $\mathcal{O}(N^{-3/4})$, which means that the integrand for that scene fulfills the C_0 smoothness condition most of the times. A comparison between the convergence rates in Fig. 2.8



(a) Glossy component.



(b) Diffuse component.

Figure 2.8 – RMSE plots for the three test scenes as a function of the number of samples. The slopes of the RMSE line fits are displayed in-between brackets.

shows that the convergence slope of the SF point set is in general as good or better than those of the other tested point sets. Note that when the convergence slope is steeper for all methods (e.g., diffuse component of the Cornell Box scene) SF point sets clearly outperform the other tested point sets. This can be explained by the fact that spherical Fibonacci point sets are more able to take advantage of smooth integrands. According to [BD11], a convergence rate as high as $\mathcal{O}(N^{-2})$ is possible with SF point sets in the

case of very smooth integrands. On the other hand, when the rate of decay is close to $\mathcal{O}(N^{-1/2})$ (e.g., the glossy component of the Cars scene in Fig. 2.8), all the point sets yield similar performances since QMC in general is inefficient for very discontinuous integrands.

		Glossy Component				Diffuse Component			
		Lambert		Shirley-Chiu		Lambert		Shirley-Chiu	
Scene	Point set	RMSE	Same quality rays needed	RMSE	Same quality rays needed	RMSE	Same quality rays needed	RMSE	Same quality rays needed
Cornell Box	Sobol	+19.2%	658 (+28.5%)	+18.1%	667 (+30.2%)	+24.9%	762 (+48.8%)	+24.2%	760 (+48.5%)
	BNOT	+15.2%	610 (+19.2%)	+7.8%	601 (+17.4%)	+25.9%	785 (+53.4%)	+7.4%	605 (+18.1%)
	L-P	+3.4%	544 (+6.3%)	+3.0%	541 (+5.6%)	+8.7%	603 (+17.8%)	+10.9%	614 (+19.9%)
Room	Sobol	+38.7%	799 (+56.0%)	+30.0%	718 (+40.3%)	+14.6%	661 (+29.1%)	+14.8%	662 (+29.2%)
	BNOT	+20.9%	665 (+29.9%)	+36.6%	763 (+49.1%)	+9.4%	601 (+17.5%)	+4.8%	569 (+11.2%)
	L-P	+8.5%	557 (+8.8%)	+24.6%	665 (+29.8%)	+4.1%	555 (+8.5%)	+4.5%	558 (+9.0%)
Cars	Sobol	+2.2%	528 (+3.1%)	+2.1%	531 (+3.7%)	+14.7%	634 (+23.9%)	+12.2%	607 (+18.5%)
	BNOT	+2.5%	526 (+2.7%)	+1.2%	520 (+1.6%)	+7.3%	569 (+11.0%)	+6.4%	559 (+9.3%)
	L-P	+0.6%	514 (+0.4%)	+1.1%	518 (+0.5%)	+3.5%	535 (+4.5%)	+4.4%	544 (+6.3%)

Table 2.1 – Comparison of the results obtained using a Sobol sequence, blue noise and the Larcher-Pillichshammer points, relative to those obtained using spherical Fibonacci point sets. The glossy and diffuse components are presented separately, as well as the used projection. For each projection, the first column shows the relative RMSE w.r.t. that of spherical Fibonacci, using 512 sample rays for all methods. The second column shows the number of rays required to achieve the same RMSE as spherical Fibonacci with 512 rays. In-between brackets is the corresponding percentage.

6.3 Efficiency and image quality

The benefit of using spherical Fibonacci point sets is thoroughly assessed in Tab. 2.1. The results show that for 512 samples per shading point, the RMSE of L-P, BNOT and Sobol point sets w.r.t. to that of SF can be up to +8.7%, +36.6% and +38.7% respectively. Note that this results in an even higher percentage of saved rays. As an example, for the same cases pointed out above, L-P needs +17.8% sample rays, BNOT +49.1% and Sobol +56% to achieve the same RMSE as spherical Fibonacci with 512 sample rays. These results have been computed by extrapolating the RMSE curves from Fig. 2.8.

The number of rays needed to close the gap between the RMSE of SF with 512 rays and that of the other methods depends on the rate of convergence for the given configuration: scene, sampling method, spherical projection and radiance component. This can be clearly seen in Tab. 2.1 for the Room scene using the L-P points and a Lambert projection. In this case, the relative RMSE of the glossy component (+8.5%) is more than twice that of the diffuse component (+4.1%), using 512 samples. But both components require approximately the same number of samples (around 556) to achieve the same RMSE than SF with 512 samples. The reason for this is that the glossy component converges faster than the diffuse component (see caption of Fig. 2.8).

The improvement brought by the use of spherical Fibonacci point sets can be appreciated on the close-up views of Fig. 2.9 which show that SF yields less visual noise compared to the other methods. As for the Room scene in Fig. 2.10, the error images indicate that SF performs better in critical areas such as the specular highlights. In the Cars scene (Fig. 2.11) on the other hand, the high discontinuity of the incident radiance makes the performance of all methods be roughly similar (as shown in Tab. 2.1). Nevertheless, it is still possible to identify image regions where the incident radiance is smoother, which favors SF point sets as shown on the top row of the close-up views of Fig. (2.11).

Fig. (2.12) shows images computed with the SF point sets and L-P with a Lambert projection. We have compared SF with L-P since they both provide the smallest RMSE in the other test cases. The scene is made up of four plates, each one having a different shininess coefficient. It contains seven light sources of variable size and variable radiance producing an incident radiance along the plates of variable frequency. Note that this frequency increases as we approach the center of the plates. The objective is to show, through this scene, how SF and L-P point sets behave for different frequencies of the incident radiance and different shininess coefficients. Both the images generated with SF and L-P exhibit the same general features: for the same illumination conditions the noise increases as n decreases, and for the same shininess coefficient n the noise is higher for sharp variations of the incident radiance. We can notice that the image generated using SF point sets provides results with a lower noise when compared to that generated using the L-P points. Furthermore, despite the regularity of the SF point sets, no regular patterns can be seen thanks to the used spherical scrambling method. The relative RMSE of the image generated with the L-P points w.r.t. that of the image corresponding to SF point sets is 7.55% higher.

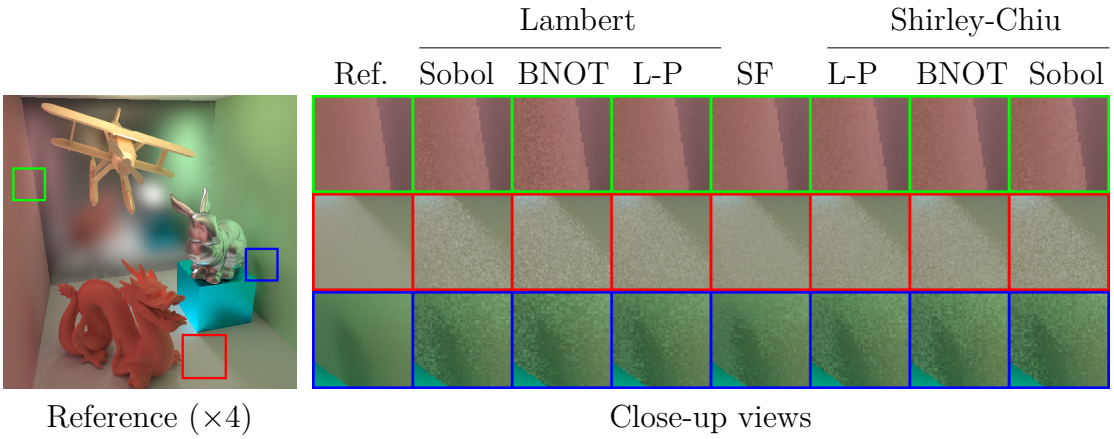


Figure 2.9 – Cornell Box scene (indirect radiance component only). The rabbit, the blue box and the back wall material contain a glossy BRDF, while the rest of the objects have a perfectly diffuse BRDF. Left: reference image multiplied by a factor of 4. Right: close-up views for all the used methods with 128 and 256 sample rays for the glossy and diffuse components respectively.

7 Conclusions

In this chapter, we have presented an algorithm for efficient generation of high quality spherical QMC sequences for approximating illumination integrals. The advantages of our approach can be summarized as follows:

Simplicity: The SF point sets algorithm is simpler to implement than the other tested QMC spherical sample sets.

Compactness: A single sequence is needed to synthesize an image. This is achieved by exploiting the axial symmetry of the BRDF lobes, which allows scrambling the point sets directly on the spherical domain using just a random axial rotation. This feature makes SF point sets particularly well-suited to GPU implementations.

Efficiency: SF point sets outperform L-P, Sobol and blue noise-based QMC in all the test cases, allowing to save a significant amount of sampling rays for the same image quality.

The main reason for the improvement brought by spherical Fibonacci point sets is that they better suit the spherical geometry. The other methods, in contrast, by focusing on the unit square distribution, neglect the fact that the discrepancy properties of the samples set on the unit square are unpaired by the spherical projection.

8 Future work

An obvious research line is to develop adaptive sampling schemes while keeping the high quality of the energy criterion exhibited by the spherical Fibonacci point sets.

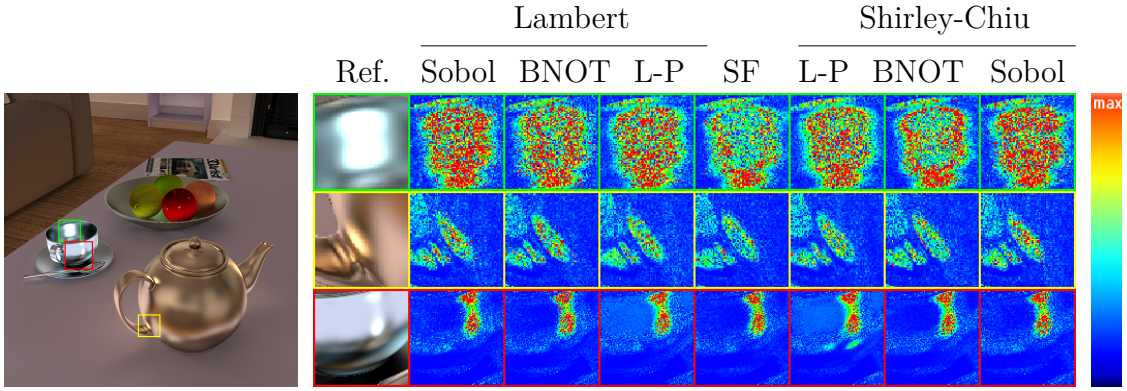


Figure 2.10 – Room scene (indirect radiance component only). The teapot, the teacup and the fruit-dish materials contain a glossy BRDF, while the rest of the objects have a perfectly diffuse BRDF. Left: reference image multiplied by a factor of 4. Right: close-up views comparison of the error images for all the used methods using 32 and 128 sample rays for the glossy and diffuse components respectively. The color encodes the error magnitude.

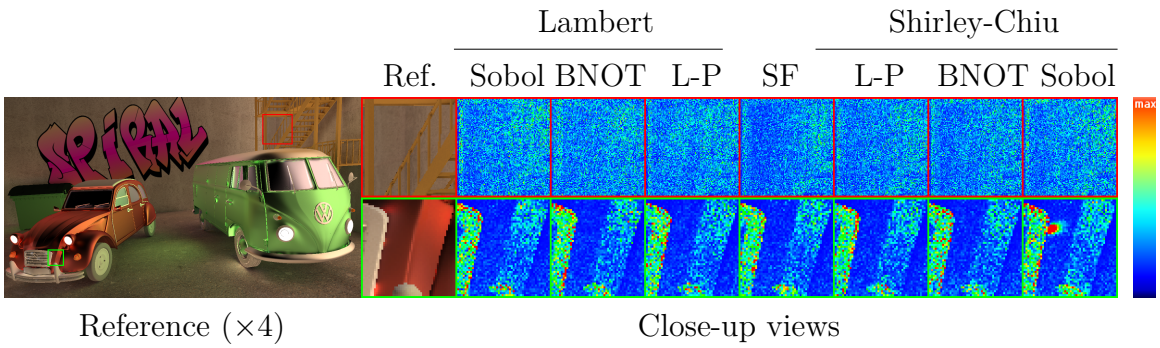


Figure 2.11 – Cars scene. The materials associated with the glasses and the body-works of both cars contain a glossy BRDF, while the rest of the objects have a perfectly diffuse BRDF. Left: reference image multiplied by a factor of 4. Right: close-up views comparison of the error images for all the used methods with 512 sample rays. The color encodes the error magnitude.

As for increasing the accuracy of integral estimate, one could resort to non-frequentist approaches, as we will see in the next chapter. They allow adapting the sampling patterns according to a global covariance function of the incident radiance samples. Another possible research line is the reduction of the perceived error by introducing some correlation between the random rotation angles assigned to sample sets used in illumination integrals of neighbour pixels.

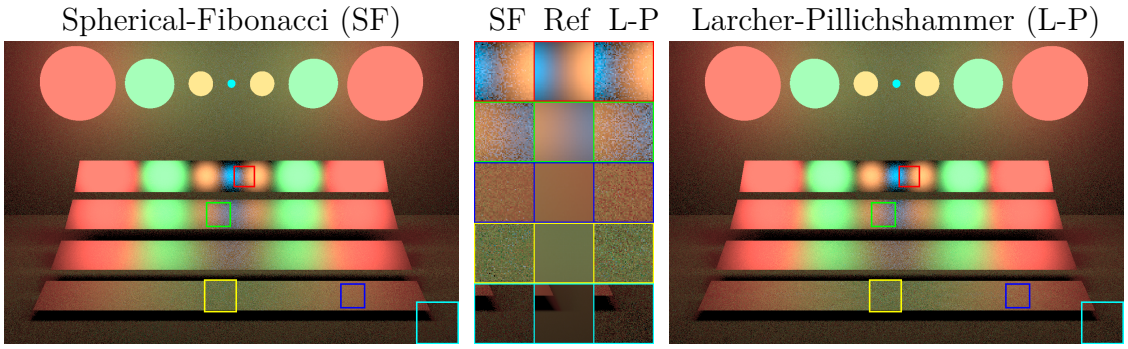


Figure 2.12 – This figure shows how spherical Fibonacci (SF) point sets behave for an incident radiance function covering a wide range of frequencies and materials of different glossinesses. Direct lighting of different plates with light sources of varying size using SF (left) and Larcher-Pillichshammer (L-P, right) point sets. The L-P point sets have been projected using the Lambert cylindrical projection. The Phong shininess coefficient n of each of the plates is 10, 50, 80 and 200 from bottom to top respectively, while the background is perfectly diffuse. The RMSE of the image rendered using the L-P points is 7.55% higher. The maximum quadratic error per pixel is 0.39 for SF and 0.71 for L-P.

A Spherical Gaussian Framework for Bayesian Monte Carlo Rendering

3

Contents

The Monte Carlo method has proved to be very powerful to cope with global illumination problems but it remains costly in terms of sampling operations. In various applications, previous work has shown that Bayesian Monte Carlo can significantly

outperform importance sampling Monte Carlo thanks to a more effective use of the prior knowledge and of the information brought by the samples set. These good results have been confirmed in the context of global illumination but strictly limited to the perfect diffuse case. Our main goal in this chapter is to propose a more general Bayesian Monte Carlo solution that allows dealing with non-diffuse BRDFs thanks to a spherical Gaussian-based framework. We also propose a fast hyperparameters determination method which avoids learning the hyperparameters for each BRDF. These contributions represent two major steps towards generalizing Bayesian Monte Carlo for global illumination rendering. We show that we achieve quality improvements over state of the art spherical QMC integration presented in the previous chapter (spherical Fibonacci) at comparable computational cost.

1 Introduction

In global illumination rendering, the main issue lies in the computation of multi-dimensional integrals involving intensive ray-traced sampling. Although the Monte Carlo method has proved to be very powerful when coping with this problem, it remains costly in terms of sampling operations. As the computational cost of sampling is very high compared to the sheer quadrature cost, we can ask ourselves if Monte Carlo methods make an efficient use of information brought by the samples. This question has been raised by O'Hagan which led him to propose the "Bayes-Hermite quadrature" [O'H91], a new form of quadrature which is referred to as "Bayesian Monte Carlo" (BMC) by other authors. While keeping the fundamental property of data dimension independence of Monte Carlo methods, it considerably broadens the set of theoretical tools that can be used to exploit the information produced by sampling. In particular, BMC uses the information regarding the samples location, which is ignored in the classic Monte Carlo method (CMC). Moreover, BMC offers much more flexibility in the exploitation of the prior knowledge compared to CMC which mainly relies on sampling strategies. However, all these advantages are obtained at the expense of the quadrature complexity and additional preprocessing. Brouillat et al. [BBL+09] have proposed efficient solutions that make the overhead of computing the BMC quadrature negligible compared to CMC. Moreover, their results show that BMC can significantly outperform CMC methods, even when including the preprocessing step. Nevertheless, their work only considers diffuse reflections and their strategy for efficiently computing the quadrature coefficients does not apply to non-diffuse BRDFs. A direct application of the method to view dependent BRDFs would require massive precomputations of multidimensional tables so as to allow an acceptable rendering time. The prior model construction strategy presented in Brouillat et al. [BBL+09] would also be inappropriate to the sharpness features of glossy BRDFs. It would require performing a learning phase for each BRDF present in the scene, which would represent a high computational cost. Furthermore, the proposed samples set optimization method is not suited to the highly non-uniform samples distributions needed to efficiently compute glossy reflections. Applied to glossy BRDFs, their method converges towards local minima which are very far from the optimal solution.

In this chapter we propose a new theoretical framework that includes a novel method of quadrature computation based on spherical Gaussian functions that can be generalized to a broad class of BRDFs (any BRDF which can be approximated by a sum of one or more spherical Gaussian functions) and potentially to other rendering applications. We account for the BRDF sharpness by using a new computation method for the prior mean function and by introducing a new factor in the method of Brouillat et al. [BBL+09] for constructing optimized sample sets. Lastly, we propose a fast hyperparameters evaluation method that avoids the learning step.

In the following, after a presentation of related work, we introduce the theoretical aspects of BMC. Then we present the application of BMC to the illumination integral and develop our theoretical framework. It is followed by a description of the full rendering algorithm and a presentation of comparative results. Then, we suggest various research directions in a discussion section. We conclude by summarizing what was learned from our implementation and mention new perspectives of applications of BMC.

2 Related work

BMC is a fundamental reconsideration of Monte Carlo integration. Our research work has thus few connections with previous work on CMC and on computer graphics. However, it is interesting to note that BMC implicitly includes some of the improvements that have been brought to CMC. This is the case for example of the integrand prefiltering method proposed in [KC08], which is equivalently obtained in BMC through Bayesian regression as discussed in Section (3.2).

The fundamental difference between BMC and CMC lies in the use of a prior stochastic model of the function $f(\mathbf{x})$ to be integrated. Note that in the Bayesian framework the variable $\mathbf{x}$ is considered deterministic, as opposed to CMC methods. Therefore, the randomness is not introduced by random sampling as in CMC but by the uncertainty (in a Bayesian sense) we put on the function to be integrated before any samples are drawn. Several approaches have been proposed in the literature regarding how to model this uncertainty. Our approach is based on the Bayes-Hermite quadrature of O'Hagan [O'H91] which uses a Gaussian Process (GP) for the prior model. We have found this method more appropriate for the computation of global illumination integrals because it easily leads to closed-form solutions. Rasmussen and Ghahramani [RG02] have shown that GP-based BMC can significantly outperform Monte Carlo importance sampling (MCIS). Several applications of the Bayes-Hermite quadrature have then been proposed in the literature and among them, the work of Pfingsten et al. [PHR06] is particularly interesting in the way they derive a closed-form solution. However, it is not suited to the particularities of spherical functions we are faced with in global illumination problems. Brouillat et al. [BBL+09] have proposed solutions in the context of final gathering for photon mapping and have clearly shown the benefit of BMC over MCIS. However, their method can only deal with diffuse reflections. The main goal of this chapter is thus to propose a BMC framework which can efficiently deal with the problems raised by glossy reflections.

3 Background and related work

In this section, we introduce the theoretical basis of BMC, and make a short description of the previous application of BMC to the diffuse BRDF case. Detailed information can be found in [O'H91, BBL+09, RG02]. The notations used throughout the Chapter are described in Tab. 3.1.

$\mathbf{x}$	Point in $\mathbb{R}^D$
$p(\mathbf{x})$	Analytically known function
$f(\mathbf{x})$	Unknown function interpreted as a random quantity and modeled through a Gaussian Process (GP)
$\bar{f}(\mathbf{x})$	Mean function $\mathrm{E}[f(\mathbf{x})]$
$\bar{\mathbf{F}}$	Vector of mean function values
$k(\mathbf{x}, \mathbf{x}')$	General non-stationary covariance function
$k(\mathbf{x} - \mathbf{x}')$	Stationary covariance function
$k'(\mathbf{x} - \mathbf{x})$	Stationary correlation function
$\mathbf{k}(\mathbf{x})$	Vector of covariance values between a position $\mathbf{x}$ and a set of sample locations $\mathbf{x}_1, \ldots, \mathbf{x}_n$
K	Noise-free covariance matrix
Q	Covariance matrix
Y_i	Noisy observation of f at location $\mathbf{x}_i$ $(f(\mathbf{x}_i) + \epsilon_i)$
$\mathbf{Y}$	Vector of observations $(Y_1, \ldots, Y_n)$
ϵ_i	Sample of an i.i.d. Gaussian noise $(\mathrm{E}[\epsilon_i] = 0)$
σ_n^2	Variance of ϵ_i
σ_f^2	Variance of the GP
σ_y^2	Variance of the observations $(\sigma_n^2 + \sigma_f^2)$
$L_i(\boldsymbol{\omega})$	Incident radiance from direction $\boldsymbol{\omega}$
l	Lengthscale of the GP which models L_i
$G()$	Spherical Gaussian function
$\rho()$	BRDF
m	Shininess coefficient of the BRDF
w	Gaussian BRDF lobe width $w = 1/\sqrt{m}$

Table 3.1 – Notations description.

3.1 The Gaussian process prior

As in any Bayesian method, we need to state our knowledge prior to performing observations. This is usually defined as a probability distribution often simply called the prior. In BMC, the prior is modeled by a Gaussian process as explained in the following.

Let us consider the computation of the integral:

$$I = \int f(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \,, \qquad \text{with } \mathbf{x} \in \mathbb{R}^D \tag{3.1}$$

where $p(\mathbf{x})$ is analytically known and $f(\mathbf{x})$ can only be determined through physical observations or, as in our case, through numerical evaluations. Given the high cost of these evaluations, it is infeasible to have a detailed knowledge about $f(\mathbf{x})$ in all its domain. Our knowledge about $f(\mathbf{x})$ is restricted to a limited set of samples while anywhere else in the domain, we are not sure about the value of $f(\mathbf{x})$. This uncertainty about $f(\mathbf{x})$ leads us to interpret it as a random quantity. The Bayesian reasoning states that all forms of uncertainty can be modeled by probability. Consequently, $f(\mathbf{x})$ can be considered as random because its value is unknown and thus uncertain. To model our uncertainty about $f(\mathbf{x})$, we will use a stochastic model called Gaussian process (GP). For a more practical knowledge about GP, the reader may refer to [Gei07]. Formally, a GP is a collection of random variables, any finite number of which has a joint Gaussian distribution. A GP is completely defined by its mean function $\bar{f}(\mathbf{x})$ and its covariance function $k(\mathbf{x}, \mathbf{x}')$, which must be positive definite [RW06]:

$$\bar{f}(\mathbf{x}) = \mathrm{E}[f(\mathbf{x})] \tag{3.2}$$

$$k(\mathbf{x}, \mathbf{x}') = \mathrm{E}[(f(\mathbf{x}) - \bar{f}(\mathbf{x}))(f(\mathbf{x}') - \bar{f}(\mathbf{x}'))] \tag{3.3}$$

and will be denoted as:

$$f(\mathbf{x}) \sim \mathcal{GP}[\bar{f}(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')] \,.$$

We use the GP formalized by Eqs. (3.2) and (3.3) as our prior model. $\bar{f}(\mathbf{x})$ is our expectation of $f(\mathbf{x})$ before any observation is made. The covariance function characterizes the belief we have on the smoothness of $f(\mathbf{x})$, i.e., how correlated are nearby samples. A strong correlation between the samples implies that $f(\mathbf{x})$ is very smooth. The covariance function of the prior GP is often assumed stationary. A covariance function k is stationary when $k(\mathbf{x}, \mathbf{x}') := k(\mathbf{x} - \mathbf{x}')$, $\forall(\mathbf{x}, \mathbf{x}')$, in which case the variance of $f(\mathbf{x})$, i.e., $k(\mathbf{x}, \mathbf{x}) = k(\mathbf{0})$, is constant. The prior covariance function k is parametrized by a set of *hyperparameters* which need to be determined in a preprocessing step. This problem will be addressed in sections 4.2.2 and 5.2. For now, we assume that the hyperparameters are known.

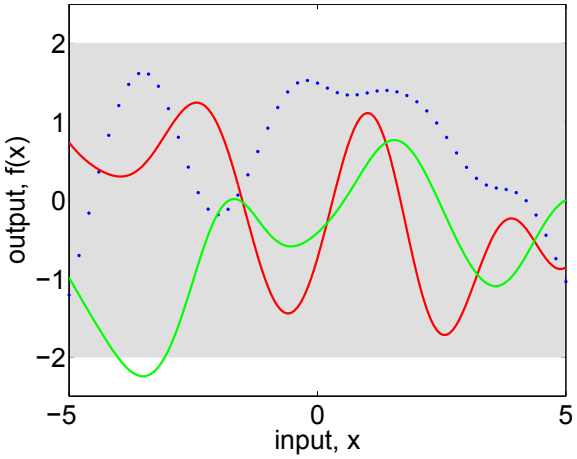


Figure 3.1 – Examples of different functions drawn at random from a GP prior with a constant mean $\bar{f} = 0$ and a stationary covariance function k. Figure from [RW06].

Fig. 3.1 shows an example of a prior GP with a constant mean $\bar{f} = 0$ and a stationary covariance function k. The grey area represents the interval of confidence of

95% around the mean value for each input $\mathbf{x}$. The red, green and blue lines are different realizations of the GP. The largest the covariance (induced by k), the smoother the realizations of the GP.

3.2 The posterior Gaussian process

Once the prior is defined, we can collect observations of $f(\mathbf{x})$ so as to refine our model by leveraging the prior GP. The resulting process is also a GP and is called posterior GP. In this section, we describe how to compute the posterior GP given the prior and a set of observations.

Let us suppose that we are provided with a set $\mathcal{D}$ of noisy samples of $f(\mathbf{x})$:

$$\mathcal{D} = \{(\mathbf{x}_i, Y_i) \,|\, i = 1, \ldots, n\} \quad \text{with} \quad Y_i = f(\mathbf{x}_i) + \varepsilon_i$$

the ε_i being samples of an independent, identically distributed (i.i.d.) Gaussian noise with zero mean and variance σ_n^2, $\mathbf{x}_i$ being a sample location (the input) and Y_i its corresponding sample value (the output). Given this additive noise assumption, the covariance of the observations becomes:

$$\operatorname{cov}(Y_p, Y_q) = k(\mathbf{x}_p, \mathbf{x}_q) + \sigma_n^2 \delta_{pq} \tag{3.4}$$

where δ_{pq} is the Kronecker symbol. σ_n^2 is thus a hyperparameter of the prior model and represents the variance which is unexplained by the GP. It is important to understand that this noise component is not only due to measurement errors and computation inaccuracies. In a very broad sense, the noise term allows to accommodate highly discontinuous data that do not fit the smoothness assumption implied by the prior GP. Indeed, illumination functions are generally discontinuous and cannot be modeled accurately with smooth basis functions. The noise term provides the necessary flexibility so that when building the posterior process, the most plausible smooth model can be fitted to discontinuous data. This data fitting has the same effect as the pre-filtering method proposed by Křivánek and Colbert [KC08] which aims at low-pass filtering the observed samples to reduce aliasing. However GP model fitting does not assume any band-limited functions hypothesis. In conclusion, BMC is applicable whatever the smoothness or the bandwidth of the integrand but of course, it performs better when the noise level σ_n^2 is low, that is when the prior fits well the data.

The posterior process results from applying the Bayes' rule to incorporate the information brought by the samples. It can be shown that the obtained posterior process is also a GP with mean and covariance functions given by [RW06]:

$$\begin{aligned} \mathrm{E}[f(\mathbf{x})|\mathcal{D}] &= \bar{f}(\mathbf{x}) + \mathbf{k}(\mathbf{x})^t Q^{-1}(\mathbf{Y} - \bar{\mathbf{F}}) \\ \operatorname{cov}[f(\mathbf{x}), f(\mathbf{x}')|\mathcal{D}] &= k(\mathbf{x}, \mathbf{x}') - \mathbf{k}(\mathbf{x})^t Q^{-1} \mathbf{k}(\mathbf{x}') \end{aligned} \tag{3.5}$$

with:

$$\begin{aligned} \mathbf{k}(\mathbf{x}) &= (k(\mathbf{x}_1, \mathbf{x}), \ldots, k(\mathbf{x}_n, \mathbf{x}))^t \\ K_{i,j} &= k(\mathbf{x}_i, \mathbf{x}_j) \qquad \text{with } (i,j) \in [1,n]^2 \\ Q &= (K + \sigma_n^2 I_n) \\ \mathbf{Y} &= (Y_1, \ldots, Y_n)^t \\ \bar{\mathbf{F}} &= \left(\bar{f}(\mathbf{x}_1), \ldots, \bar{f}(\mathbf{x}_n)\right)^t \end{aligned} \tag{3.6}$$

and I_n being the $n \times n$ identity matrix. $\mathbf{Y}$ is the vector of observed samples value while $\bar{\mathbf{F}}$ is the vector of prior GP mean values at the sampling locations. Q contains the covariance between the samples and is called covariance matrix. Eq. (3.5) gives an estimate of $f(\mathbf{x})$ for an unobserved input $\mathbf{x}$ given the observed data set $\mathcal{D}$. This particular form of regression is called *Bayesian regression*.

Fig. 3.2 shows an example of a posterior GP produced by combining five samples (marked with a +) with the prior GP from Fig. 3.1. Note that the grey area denoting the 95% confidence region has been drastically reduced, especially around the sampled values where the uncertainty regarding the value of $f(\mathbf{x})$ is smaller. Consequently, the functions drawn from the posterior GP (red, green and blue lines in the plot) are much closer to each other than the ones drawn from the prior GP.

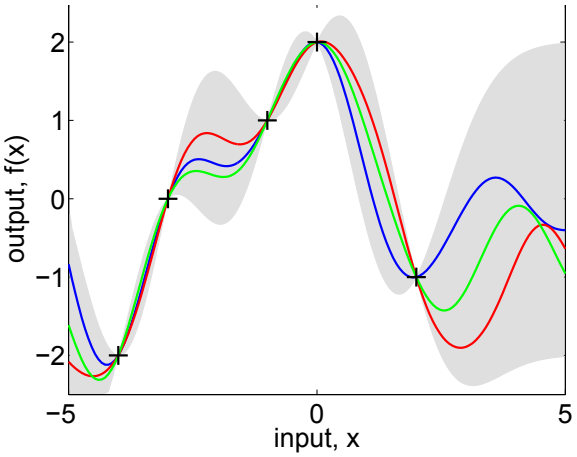


Figure 3.2 – Examples of different functions drawn at random from a posterior GP. Figure from [RW06].

3.3 Bayesian quadrature

Eq. (3.5) gives a posterior estimate of $f(\mathbf{x})$ given the observed samples. Then, from Eq. (3.1), a posterior estimate of I, that is $\hat{I} = \mathrm{E}(I|\mathcal{D})$, is obtained by integrating both terms of Eq. (3.5):

$$\hat{I} = \bar{I} + \mathbf{z}^t Q^{-1} (\mathbf{Y} - \bar{\mathbf{F}}) \tag{3.7}$$

with:

$$\bar{I} = \int \bar{f}(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \tag{3.8}$$

$$\mathbf{z} = \int \mathbf{k}(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \tag{3.9}$$

It can be seen that the posterior estimate $\hat{I}$ results from adding to the prior expectation $\bar{I}$ a corrective term that represents the effect of the observed samples $\mathcal{D}$. This term includes three factors: $(\mathbf{Y} - \bar{\mathbf{F}})$, which measures how wrong was our prior expectation for each observed sample value, Q^{-1}, the inverse covariance matrix which accounts for the relative positions of the observed samples, and the $\mathbf{z}$ vector, which captures the influence of each sample on the deterministic part of the integrand given the covariance function.

3.4 Variance analysis

3.4.1 Variance of the integral estimate and optimal sampling pattern

The posterior estimate of I given by Eq. (3.7) has a Gaussian distribution of mean $\hat{I}$ and a variance given by:

$$\mathrm{Var}(I|\mathcal{D}) = \bar{V} - \mathbf{z}^t Q^{-1} \mathbf{z} \tag{3.10}$$

with:

$$\bar{V} = \iint k(\mathbf{x}, \mathbf{x}') p(\mathbf{x}) p(\mathbf{x}') \; d\mathbf{x} d\mathbf{x}' \,. \tag{3.11}$$

Note that the estimate of the posterior variance in Eq. (3.10) does not depend on the observed values $\mathbf{Y}$. It only depends on the location $\mathbf{x}$ of the samples. This might seem not plausible at first sight but actually, the covariance function of our prior model (Eq. (3.4)) already comprises the statistical information necessary to estimate $\mathrm{Var}(I|\mathcal{D})$. This variance is due to the implicit variability of our observations given our prior. It can be interpreted as a measure of the confidence we may attribute to our integral estimate. Note also that BMC does not require drawing the samples randomly according to a PDF. However, Eq. (3.10) shows that the $\mathrm{Var}(I|\mathcal{D})$ strongly depends on the choice of the samples set position $\{\mathbf{x}_i\}$. An optimal choice consists in selecting the $\{\mathbf{x}_i\}$ that minimizes $\mathrm{Var}(I|\mathcal{D})$. This optimal set will reflect both the influence of the deterministic function $p(\mathbf{x})$ (through the $\mathbf{z}$ vector) and that of the prior knowledge on $f(\mathbf{x})$ (through the covariance function).

3.4.2 Bias considerations

As opposed to CMC, in BMC the observed data are considered as known and thus deterministic whereas the function $f(\mathbf{x})$ (Eq. (3.1)) is considered as uncertain. The BMC integral estimate is obtained by computing the integral of the most probable function among all possible realizations of the posterior GP. The estimator of Eq. (3.7) is thus unbiased (in the Bayesian sense) as its value coincides with the expected value of the posterior probability distribution. However this unbiasedness has nothing to do with the usual interpretation of this term in a classic Monte Carlo framework in which sampling is produced by a random process. That is why unbiasedness considerations can be misleading when comparing classic and Bayesian Monte Carlo, since the randomness in each of the methods is of a very different nature.

Specifically, in a frequentist approach such as CMC, the estimate can be refined by averaging estimates with respect to multiple sample sets whereas a Bayesian method

uses averaging with respect to the posterior distribution instead. In [Bis06] (Chapter 3, Section 2), Bishop applies a frequentist bias-variance analysis to regression and he shows on a parametric regression example how the regularization parameter (which, in BMC, corresponds to the noise ratio introduced in the next section) can be used to control the frequentist bias. However, Bishop points out that this is of limited practical interest since averaging w.r.t. multiple sample sets would affect the efficiency of a Bayesian approach. Indeed, as noted by Bishop, to fully benefit from the Bayesian approach, a direct application of BMC would combine multiple sample sets into one large set rather than considering each set individually. Therefore, in our BMC method, we will consider only a single samples set and apply a full-fledged Bayesian method. Furthermore, the Bayesian framework offers much more efficient methods to implement progressive refinement than averaging with respect to multiple sample sets.

3.5 The case of stationary covariance functions and the noise ratio hyperparameter

In the absence of accurate information on $f(\mathbf{x})$ variations, it is sensible to assume a stationary covariance function for the prior GP. As mentioned in Section 3.1, the variance of the prior GP is constant in the case of stationary covariance functions and we have:

$$k(\mathbf{x}-\mathbf{x}') = \sigma_f^2 k'(\mathbf{x}-\mathbf{x}') \tag{3.12}$$

where $\sigma_f^2 = k(0)$ is the prior GP variance and $k'(\mathbf{x}-\mathbf{x}')$ is its correlation function. σ_f is thus an hyperparameter of our GP model. It represents the confidence we have on the prior mean value. As the additive noise ε is assumed independent from the prior GP, the variance σ_y^2 of the observed samples value can be expressed as follows:

$$\sigma_y^2 = \sigma_f^2 + \sigma_n^2 \,.$$

The above equations reveal the σ_f and σ_n hyperparameters as totally independent variables. However, in global illumination problems, many integrals of the same type but with different data have to be computed to render a full image. The observed variance σ_y^2 can strongly vary from one integral to the other, but in practice, we can expect that the values of both hyperparameters will roughly have the same order of magnitude as σ_y^2. Therefore, the noise contribution is better characterized by the ratio σ_n/σ_f rather than σ_n alone. Henceforth, we will thus use the noise ratio hyperparameter $\sigma'_n = \sigma_n/\sigma_f$ instead of σ_n to represent the noise contribution.

In the following of this section, we show that once σ'_n has been determined, the value of σ_f is not required to compute the integral estimate $\hat{I}$.

Given Eq. (3.12), the Q matrix defined in Eq. (3.6) can be expressed as follows:

$$\begin{aligned} Q &= \sigma_f^2 Q' \\ \text{with } Q' &= K' + \sigma_n'^2 I_n \end{aligned} \tag{3.13}$$

where K' is obtained by replacing $k(\mathbf{x}-\mathbf{x}')$ by $k'(\mathbf{x}-\mathbf{x}')$:

$$K'_{i,j} = k'(\mathbf{x}_i - \mathbf{x}_j) \qquad \text{with } (i,j) \in [1,n]^2$$

We can now rewrite Eq. (3.7) with terms that depend on $\sigma_n'^2$ instead of σ_f^2 and σ_n^2:

$$\hat{I} = \bar{I} + \mathbf{z}'^t Q'^{-1}(\mathbf{Y} - \bar{\mathbf{F}}) \tag{3.14}$$

where the $\mathbf{z}'$ vector is obtained by replacing $k(\mathbf{x} - \mathbf{x}')$ by $k'(\mathbf{x} - \mathbf{x}')$ in Eq. (3.9). Each element of this vector can then be expressed by the following convolution integral:

$$z_i' = \int k'(\mathbf{x} - \mathbf{x}_i)p(\mathbf{x})d\mathbf{x} \tag{3.15}$$

where $\mathbf{x}_i$ is a sample location. Assuming that the σ_n' hyperparameter is known, the integral estimate given by Eq. (3.14) does not require determining σ_f^2 and σ_n^2. The noise ratio σ_n' can be interpreted as the level of confidence we have in the GP model fitting. The higher σ_n', the lower the proportion of the observed samples variance that can be explained by the GP.

3.6 BMC algorithm overview

To summarize, the BMC method consists of the following steps:

1. Build the GP prior by choosing a covariance function and a mean function.
2. Learn the hyperparameters associated with the covariance function.
3. Select the set of sampling positions $\{\mathbf{x}_i\}$ which minimize the variance of the BMC estimate and compute the inverted covariance matrix Q'^{-1}.
4. Compute the $\mathbf{z}'$ vector and the vector of quadrature coefficients $\mathbf{c} = \mathbf{z}'Q'^{-1}$.
5. Collect the observed samples value $\mathbf{Y}_i$ for each sampling position $\{\mathbf{x}_i\}$.
6. Compute the prior mean value vector $\bar{\mathbf{F}}$.
7. Compute the posterior estimate with Eq. (3.14).

The diffuse BRDF case. As mentioned above, global illumination entails the computation of many integrals of the same type with different data and of course, all these steps need not be repeated for each integral evaluation. As a matter of fact, in implementing a BMC algorithm, we will try to transfer the most computer intensive steps to a preprocessing stage. Brouillat et al . [BBL+09] proposed a solution in which the same set of hyperparameters is used for all integral evaluations. Since the hyperparemeters do not vary locally, the optimal samples set (obtained by minimizing $\mathrm{Var}(I|\mathcal{D})$ in Eq. (3.10)) is also the same for all integrations. This means that sample locations are known before evaluating the integral. We can thus precompute the covariance matrix Q' which only depends on the hyperparameters and on the locations of the samples. Furthermore, in the case of a purely diffuse BRDF, also the coefficients z_i' of the vector $\mathbf{z}'$ only depend on the sample locations. This feature allows precomputing the vector of quadrature coefficients $\mathbf{c} = \mathbf{z}'Q'^{-1}$. During rendering, only steps 5, 6 and 7 of the above algorithm are performed.

The glossy BRDF case. The approach taken in [BBL+09] is not applicable to the case of glossy BRDFs because of the dependency on the viewing direction. The deterministic function $p(\mathbf{x})$ that essentially contains the BRDF will then change at each integral evaluation, and steps 2 and 3 can no longer be preprocessed as we will see in the following. In the next sections, we will propose a more general approach based on a spherical Gaussian framework that can be applied to a large class of problems.

4 A spherical Gaussian-based BMC framework

We will now consider the computation of the illumination integral at a given shading point:

$$L_o(\boldsymbol{\omega}_o) = \int_{\Omega_{2\pi}} L_i(\boldsymbol{\omega}_i)\ \rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)\ (\boldsymbol{\omega}_i \cdot \mathbf{n})\ d\Omega(\boldsymbol{\omega}_i)\,. \tag{3.16}$$

In this integral, the analytical part, which corresponds to $p(\mathbf{x})$ in Eq. (3.1) is naturally the BRDF factor $\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o)(\boldsymbol{\omega}_i \cdot \mathbf{n})$. The unknown function $f(\mathbf{x})$ in Eq. (3.1), modeled with a Gaussian Process, is the incident radiance $L_i(\boldsymbol{\omega}_i)$ at the shading point in Eq. (3.16). Our goal is to compute an estimate of $L_o(\boldsymbol{\omega}_o)$ from a set of samples value $\{L_i(\boldsymbol{\omega}_j), j \in [1, n]\}$, where $\boldsymbol{\omega}_j$ are the samples location (called $\mathbf{x}_i$ in Section 3.2).

Throughout this section, we describe our approach to the different problems involved in computing the Bayesian quadrature. We start by describing a spherical Gaussian-based framework for BMC which greatly simplifies the computation of the integrals for the $\mathbf{z}'$ vector. Then we show how the prior GP covariance and mean functions are built, and how optimal sampling patterns are determined. Finally, we elaborate on our rendering algorithm.

4.1 Our theoretical approach

In this section, we essentially address step 4 of the BMC algorithm presented in Section 3.6 and more particularly the problem of making the computation of the $\mathbf{z}'$ vector tractable knowing that it is not possible to precompute the z'_j coefficients given their dependency on ω_o. Our approach is to model both the BRDF and the correlation function with spherical Gaussian functions (SGF), a choice which allows us to reduce the $\mathbf{z}'$ computation to a simple query to a scene-independent 2-entry look up table.

An interesting property of SGFs is that the product of two SGFs yields a SGF. This property will be very useful for the computation of the Eq. (3.15) integral. As far as $p(\mathbf{x})$ can be modeled by a SGFs product or mixture, the method described in this section can be applied to compute the quadrature of Eq. (3.14) given its linearity properties.

A SGF results from the restriction of a Gaussian RBF (Radial Basis Function) to the unit sphere S^2. Consequently, for $x, y \in S^2$, we have:

$$|x - y|^2 = 2(1 - x \cdot y) \tag{3.17}$$

since $|x| = |y| = 1$. A SGF can thus be expressed as follows:

$$G(x - y; \mu, \lambda) := \mu \exp\left(\frac{x \cdot y - 1}{\lambda^2}\right).$$

Since each point $x \in S^2$ corresponds to a direction $\boldsymbol{\omega}$, we model the correlation function k' by a SGF $G_{k'}$ defined as follows:

$$k'(\boldsymbol{\omega} - \boldsymbol{\omega}') := G_{k'}(\boldsymbol{\omega} - \boldsymbol{\omega}'; 1, l) = \exp\left(\frac{\boldsymbol{\omega} \cdot \boldsymbol{\omega}' - 1}{l^2}\right) \tag{3.18}$$

where l is the lengthscale hyperparameter which controls the smoothness of the prior GP. We model the BRDF ρ by a SGF G_ρ defined as:

$$\rho(\boldsymbol{\omega}_i, \boldsymbol{\omega}_o) := G_\rho(\boldsymbol{\omega}_r - \boldsymbol{\omega}_i; k_s, w) = k_s \exp\left(\frac{\boldsymbol{\omega}_r \cdot \boldsymbol{\omega}_i - 1}{w^2}\right) \tag{3.19}$$

where k_s is the specular coefficient. G_ρ has an axially symmetric lobe whose axis is aligned with $\boldsymbol{\omega}_r$ which is itself function of the outgoing direction $\boldsymbol{\omega}_o$. w characterizes the lobe sharpness, i.e., the "width" of the lobe. It is important to note that this choice does not restrict the generality of our analysis as several works have shown that most BRDFs can be approximated by a mixture of SGFs (e.g. in [WRG+09]). In particular, a single SGF is sufficient to model the glossy term of a Phong's BRDF. In this case, the lobe axis direction $\boldsymbol{\omega}_r$ is the perfect mirror incident direction:

$$\boldsymbol{\omega}_r = 2(\boldsymbol{\omega}_o \cdot \mathbf{n})\mathbf{n} - \boldsymbol{\omega}_o$$

and using the following approximation:

$$\cos^m \theta = e^{m \ln(\cos\theta)} \approx e^{m(\cos\theta - 1)} \tag{3.20}$$

with $\cos\theta = (\boldsymbol{\omega}_r \cdot \boldsymbol{\omega}_i)$, we have $w = 1/\sqrt{m}$, m being the Phong shininess parameter. The resulting approximation RMSE is below 10^{-3} when $m > 10$. Note also that modeling BRDFs with SGFs (which are naturally isotropic) does not restrict our framework to the isotropic BRDF case since anisotropic BRDFs can be modeled using a weighted sum of (isotropic) SGFs as mentioned above. Moreover, we have found that the Blinn-Phong BRDF can be modeled with only three SGFs with a RMSE below 0.05. More details regarding the potential application of our approach to multiple-lobe BRDFs are given in Section 6.2 of this chapter.

Given Eqs. (3.15), (3.18) and (3.19), each coefficient of the $\mathbf{z}'$ vector can be expressed as follows:

$$z'_j = \int_{\Omega_{2\pi}} G_{k'}(\boldsymbol{\omega}_j - \boldsymbol{\omega}'; 1, l)\ G_\rho(\boldsymbol{\omega}_r - \boldsymbol{\omega}'; k_s, w)\ d\Omega(\boldsymbol{\omega}')$$

where $\boldsymbol{\omega}_j$ is the direction vector of sample j. As the product of two SGFs is also a SGF, we have:

$$z'_j = \int_{\Omega_{2\pi}} G(\boldsymbol{\omega}_m - \boldsymbol{\omega}'; c_m, l_m)\ d\Omega(\boldsymbol{\omega}') \tag{3.21}$$

with:

$$\begin{aligned} \frac{1}{l_m^2} &= \left| \frac{\boldsymbol{\omega}_j}{l^2} + \frac{\boldsymbol{\omega}_r}{w^2} \right| \\ \boldsymbol{\omega}_m &= l_m^2 \left(\frac{\boldsymbol{\omega}_j}{l^2} + \frac{\boldsymbol{\omega}_r}{w^2} \right) \\ c_m &= k_s \exp\left(\frac{1}{l_m^2} - \frac{1}{l^2} - \frac{1}{w^2} \right) . \end{aligned}$$

By developing Eq. (3.21), we have:

$$z'_j = c_m \, S_g(\boldsymbol{\omega}_m, l_m) \, .$$

where $S_g(\boldsymbol{\omega}, l)$ is the spherical Gaussian integral (SGI):

$$S_g(\boldsymbol{\omega}, l) = \int_{\Omega_{2\pi}} \exp\left(\frac{\boldsymbol{\omega} \cdot \boldsymbol{\omega}' - 1}{l^2} \right) d\Omega(\boldsymbol{\omega}') \tag{3.22}$$

The computation of the $\mathbf{z}'$ coefficients is then reduced to evaluations of the SGI of Eq. (3.22) which, given its smoothness, can easily be tabulated for quick evaluations. Only the elevation angle θ between the direction $\boldsymbol{\omega}$ and the normal $\mathbf{n}$ is necessary to specify the input direction $\boldsymbol{\omega}$ in Eq. (3.22) given the axial symmetry of the SGI about the normal $\mathbf{n}$. Therefore, only a single 2-entry table with (θ, l) as inputs is necessary for SGI evaluations. This table is independent of the scene and the used BRDFs. It is computed just once and used for any BMC integration within this framework.

Fig. 3.3 illustrates the shape of the SGI $S_g(\theta, l)$ for $(\theta, l) \in [0, \ldots, \pi] \times [0, \ldots, 1]$. Note that, as the value of θ increases, the value of the SGI decreases. This is because the part of the lobe which remains inside the domain of integration gets smaller and smaller. Such an effect is more visible for larger lobes.

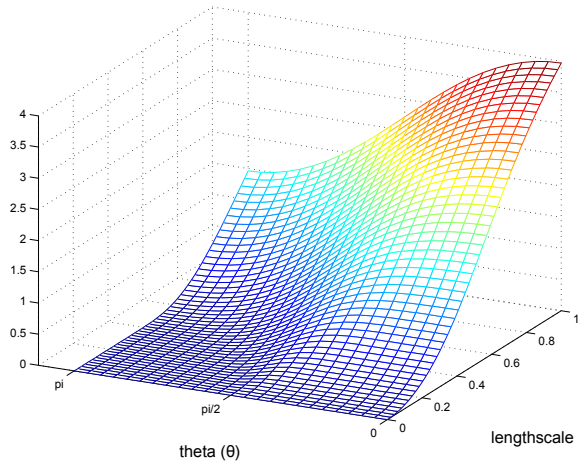


Figure 3.3 – Plot of the SGI 2D table values. In this example 25 values have been used for the θ dimension, and 50 values have been used for the lengthscale l. In our work we have used 200 values for each dimension. In such a case, the table is of size 160 KB using floating point notation.

4.2 Constructing the prior GP model

4.2.1 Determining the mean function

To define the GP, a mean function $\bar{L}_i(\boldsymbol{\omega}_i)$ corresponding to $\bar{f}(\mathbf{x})$ in Section 3.1 must be specified. Our approach to this question is to assume a locally constant mean function $\bar{L}_i$ that will be determined from the observed samples value $L_i(\boldsymbol{\omega}_j)$. A simple average of the samples value would provide a strongly biased $\bar{L}_i$ estimate since the samples distribution is not at all uniform as explained in Section 4.3. Instead, we leverage the prior GP as described in Chapter 2, Section 7 of [RW06]. The method consists in inferring a mean function from the observed samples value through an explicit basis function model. In our case, we only use one basis function ($h(\boldsymbol{\omega}) = 1$) and its associated weight is thus the desired constant mean value, which yields:

$$\bar{L}_i = \frac{HQ'^{-1}Y}{HQ'^{-1}H^t} \tag{3.23}$$

where $H = [h(\boldsymbol{\omega}_1), \ldots, h(\boldsymbol{\omega}_n)] = [1, \ldots, 1]$ and $Y = [L_i(\boldsymbol{\omega}_1), \ldots, L_i(\boldsymbol{\omega}_n)]^t$ is the vector of observations. As shown in [RW06], this method slightly increases the estimates variance but this can be neglected in practice. Then, from Eq. (3.8) and Eq. (3.19) we have:

$$\bar{I} = \bar{L}_i \times \mu(\boldsymbol{\omega}_r) \tag{3.24}$$

where $\mu(\boldsymbol{\omega}_r)$ is the SGI:

$$\mu(\boldsymbol{\omega}_r) = \int_{\Omega_{2\pi}} G_\rho(\boldsymbol{\omega}_r - \boldsymbol{\omega}; k_s, w)\ d\Omega(\boldsymbol{\omega})\,.$$

To evaluate this integral, the same 2-entry table as the one required for evaluating the SGI of Eq. (3.22) can be used. Moreover, since $H = [1, \ldots, 1]$, Eq. (3.23) can be expressed as a weighted sum of the observed samples value:

$$\bar{L}_i = \frac{1}{\Gamma}\sum_j \gamma_j Y_j$$

where the weight γ_j is equal to the sum of the coefficients of the j^{th} column of the Q'^{-1} matrix and $\Gamma = \sum_j \gamma_j$ is the sum of all coefficients of the Q'^{-1} matrix. All the γ_j/Γ weights can be precomputed once the sampling pattern and the hyperparameters have been determined.

4.2.2 Hyperparameters selection

In the preceding sections, we have shown that only two hyperparameters are required to compute the integral estimate from Eq. (3.14): the noise ratio σ'_n and the lengthscale l of the correlation function (Eq. (3.18)). Our strategy for determining appropriate hyperparameter values consists in using the same set of hyperparameters for all illumination integrals that involve the same BRDF. We shall see in Section 5 that this choice

is perfectly acceptable since the hyperparameters values mainly depend on the BRDF shininess. This simplification allows us to learn the hyperparameters per BRDF at scene level, and then precompute the inverse covariance matrix Q^{-1} and the optimal sampling pattern (i.e., steps 2 and 3 of the algorithm of Section 3.6).

To learn the hyperparameters at a scene level for a given BRDF, we use the same principle as Brouillat et al. [BBL+09]. It consists in numerically estimating the covariance between the samples, and then fitting it to our covariance function model given by $\mathrm{cov}(\boldsymbol{\omega}_p, \boldsymbol{\omega}_q) = \sigma_f^2 \exp\left(\frac{(\boldsymbol{\omega}_p \cdot \boldsymbol{\omega}_q)-1}{l^2}\right) + \sigma_n^2 \delta_{pq}$. Nevertheless, the application of this principle to the glossy BRDF case implies designing a learning algorithm which is different from that of Brouillat et al.

The conditions in which the covariance is estimated should be as close as possible to the ones which will be found when evaluating the integral. In particular, the samples used to compute the expected value for the integration procedure (Eqs. (3.2) and (3.23)) are distributed according to the lobe of the actual BRDF. Therefore, we have decided to compute the expected value for the covariance estimation procedure using the same samples distribution. This decision implies that the covariance is evaluated separately for each BRDF shininess. Furthermore, to simplify the learning procedure, we simulated the incident radiance using an environment map.

We propose a learning method in which the training set is built as a superset of incident radiance $\{L_i(\boldsymbol{\omega}_j)\}$. Each of these sets is designed so as to be nearly optimal for the quadrature equation. As the integral of Eq. (3.16) is strongly dependent on the BRDF, each $\{\boldsymbol{\omega}_j\}$ set will be more or less concentrated around the $\boldsymbol{\omega}_r$ direction specified for the BRDF. To take this into account for the determination of the global hyperparmeters of the $L_i(\boldsymbol{\omega})$ GP model, our method proceeds as follows.

1. Generate a set of n_r uniformly-distributed direction vectors $\{\boldsymbol{\omega}_r\}$ over the S^2 sphere (the basic spiral point algorithm proposed in [SK97] can be used for this). The resulting distribution is depicted in Fig. 3.4, center.

2. For each $\boldsymbol{\omega}_r$:

(a) Generate a set $\{\boldsymbol{\omega}_j\}$ of n sampling directions (Fig. 3.4, left) using the spiral points of Algorithm 2 without the polynomial distribution unknown at this stage. Center $\{\boldsymbol{\omega}_j\}$ around the actual $\boldsymbol{\omega}_r$ (see Fig. 3.4, right). Compute the corresponding $L_i(\boldsymbol{\omega}_j)$ values.

(b) Compute the Monte Carlo estimate of the mean incident radiance value:

$$\bar{L} = \frac{1}{n} \sum_j \frac{L_i(\boldsymbol{\omega}_j)}{\rho(\boldsymbol{\omega}_j, \boldsymbol{\omega}_r)}$$

(c) Build a covariance matrix $A_{\boldsymbol{\omega}_r}$, the coefficients of which are the products:

$$a_{ij} = a_{ji} = (L_i(\boldsymbol{\omega}_i) - \bar{L})(L_i(\boldsymbol{\omega}_j) - \bar{L})$$

3. Compute $\bar{A}$, the arithmetic mean of the $A_{\boldsymbol{\omega}_r}$ matrices.

4. Perform a linear regression on all the $\bar{A}$ matrix coefficients except the diagonal. For this, we take the square of the Euclidean distance between pairs of sampling directions (see explanations below) as input variable, and the natural logarithm of the $\bar{A}$ coefficients value as output variable (see Eq. (3.25)). The slope gives $1/2l^2$ and the intercept $2\ln(\sigma_f)$.

5. Subtract σ_f^2 from the mean of the diagonal coefficients to obtain σ_n^2.

The goal of steps 1-3 is to build an average covariance matrix $\bar{A}$ by analyzing the whole range of incidence directions while keeping a point distribution that fits the BRDF. $\bar{A}$ can then be considered as an observed sample of matrix Q (Eq. (3.6)). As all A_{ω_r} elements are produced with the same point distribution, the Euclidean distance $|\boldsymbol{\omega}_i - \boldsymbol{\omega}_j|$ between pairs of sampling directions $\boldsymbol{\omega}_i$ and $\boldsymbol{\omega}_j$ that are used to compute a_{ij} is the same for all A_{ω_r} matrices and thus $\bar{A}$. Therefore:

$$|\boldsymbol{\omega}_i - \boldsymbol{\omega}_j| = r_{ij} \, .$$

These distances can then be computed once for all and used in the linear regression step. Given Eq. (3.17) and Eq. (3.18), we have:

$$\ln[k(\boldsymbol{\omega} - \boldsymbol{\omega}')] = 2\ln(\sigma_f) - \frac{|\boldsymbol{\omega} - \boldsymbol{\omega}'|^2}{2l^2}, \tag{3.25}$$

hence the result of the linear regression. We only consider the range of r_{ij} values which correspond to significant values of covariance. In our implementation, we have obtained good results with $n_r = 4K$ sample sets, each one containing $n = 80$ samples, yielding a training set with $320K$ samples.

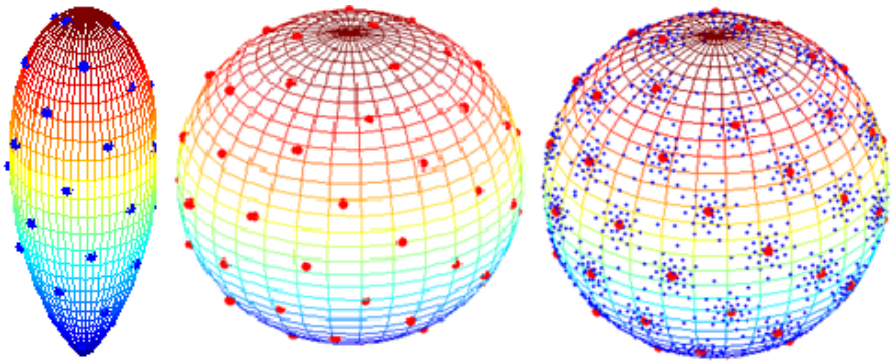


Figure 3.4 – Samples set layout for covariance estimation. On the left: samples set $\{\boldsymbol{\omega}_j\}$ uniformly distributed on a SGF BRDF lobe with a given shininess. Center: set of $\boldsymbol{\omega}_r$ directions uniformly distributed on the sphere S^2. Right: $\{\boldsymbol{\omega}_j\}$ centered on each $\boldsymbol{\omega}_r$.

Using the procedure described above, we sample the incoming radiance with a distribution concentrated within the BRDF lobe. In this way, the fitting of the covariance function is adapted for the range of interest of the inter-samples distance, that is, a narrow range of distances for high shininess and a wide one for low shininess. But this approach has a limitation regarding scenes with several BRDFs since it implies learning the hyperparameters separately for each shininess parameter used in the scene.

Although this operation takes just a few seconds per shininess parameter (typically around 1 to 4 seconds depending on the scene complexity), it would make the BMC approach inefficient for scenes with different shininess parameters. To cope with this problem, a fast method to approximate the hyperparameters is proposed in Section 5.4.

4.3 Optimal sampling pattern

The choice of the sample directions $\{\boldsymbol{\omega}_j\}$ is crucial to the quality of the rendering integral estimate, especially in the case of glossy BRDFs. Nevertheless, not only the BRDF but also the characteristics of the incident radiance function L_i should be taken into account for this choice. These characteristics are represented in a stochastic manner through the hyperparameters of the prior GP.

The variance estimate in Eq. (3.10) factors in the influence of the BRDF (through $\mathbf{z}'$), the prior GP hyperparameters, and the sampling pattern defined by the $\{\boldsymbol{\omega}_j\}$ set. Moreover, since the *global hyperparameters* σ_n' and l are known at this stage (see Section 3.6), the variance only depends on the $\{\boldsymbol{\omega}_j\}$ set. This is a strong feature of the BMC framework since we can optimize the samples direction in order to minimize $\mathrm{Var}(I|\mathcal{D})$.

Optimizing the directions for a large set of samples can become cumbersome. A naive approach would be to consider that the variance is a function of n variables which are the samples directions. But minimizing such a function would become intractable even for medium size sample sets (40 samples or more). Our solution to this problem consists in modifying the spherical Fibonacci point sets algorithm presented in the previous chapter as shown in Alg. 2.

Algorithm 2 The modified spiral points algorithm.

1: $\Delta\phi \leftarrow \pi(3-\sqrt{5})$ ▷ Compute the step on ϕ
2: $\phi \leftarrow 0$ ▷ Initialize ϕ for the first sample
3: $\Delta z \leftarrow 1/n$ ▷ Compute the step on z
4: $z \leftarrow 1-\Delta z/2$ ▷ Initialize z for the first sample
5: **for all** $k \leftarrow [1:n]$ **do**
6: $\quad z_p \leftarrow \mathrm{polyval}(\mathit{coeff}, z)$ ▷ Covariance effect
7: $\quad z_v \leftarrow \frac{1}{m}\ln[1+z_p(e^m-1)]$ ▷ BRDF effect
8: $\quad \theta_k \leftarrow \arccos(\min(z_v, 1))$ ▷ Compute θ angle
9: $\quad \phi_k \leftarrow \mathrm{mod}(\phi, 2\pi)$ ▷ Compute ϕ angle
10: $\quad z \leftarrow z-\Delta z$ ▷ Give a step on z
11: $\quad \phi \leftarrow \phi+\Delta\phi$ ▷ Give a step on ϕ
12: **end for**

The original algorithm generates a discrete spiral on a sphere about the up axis z with a constant pitch (Δz). In our application, the z axis is aligned with the surface normal $\mathbf{n}$. The z coordinates of the original spiral points are thus uniformly distributed. We replace the uniform distribution by a polynomial distribution (whose coefficients are called *coeff* in Alg. 2, line 6) to account for the covariance effect (line 6). In addition, we change this algorithm to produce a discrete spiral on the BRDF lobe (line

7) rather than on the sphere. This can be obtained by warping the z coordinates with the following function:

$$z_v = g(z) = \frac{1}{m} \ln[1 + z(e^m - 1)]$$

where m is the Phong shininess parameter. Details regarding the derivation of this warping function will be given in Section 4.4.

The optimizer will then take the polynomial coefficients *coeff* as input arguments to minimize $\mathrm{Var}(I|\mathcal{D})$ instead of taking as input all the sampling directions. The number of parameters to optimize is thus reduced to the number of coefficients of the polynomial (a polynomial of degree 3 is sufficient). Actually, only the term $-\mathbf{z}^t Q^{-1}\mathbf{z}$ of Eq. (3.10) needs to be considered since $\bar{V}$ does not depend on the sampling pattern. The convergence is very fast (a few seconds) with the BFGS quasi-Newton method provided by Matlab and the result is not very sensitive to the initial values even if the initial polynomial is of degree 0.

The variance of the BMC estimate strongly depends on the correlation function which is, in its turn, parametrized by the lengthscale hyperparameter. We can thus expect the optimal sample set to depend on the lengthscale value. A very smooth L_i function implies a large lengthscale value. Consequently, a high concentration of samples around the BRDF apex would be redundant as the samples are highly correlated. This explains the sparsity of the samples on the lobe (Fig. 3.5(b)). Conversely, if L_i has a low lengthscale value the samples will tend to be concentrated around the BRDF lobe apex (Fig. 3.5(a)). In a preprocessing step, we compute optimal sampling patterns and store the sample directions. The sample sets are computed for a discrete set of pairs (l, m), l and m being the lengthscale and the Phong shininess parameter respectively. With respect to the noise ratio σ'_n, we have experimentally found that using $\sigma'_n = 0.25$ yields the best results. These optimal sampling patterns are independent of the scene to be processed. When rendering, we select the appropriate sampling patterns from the precomputed ones.

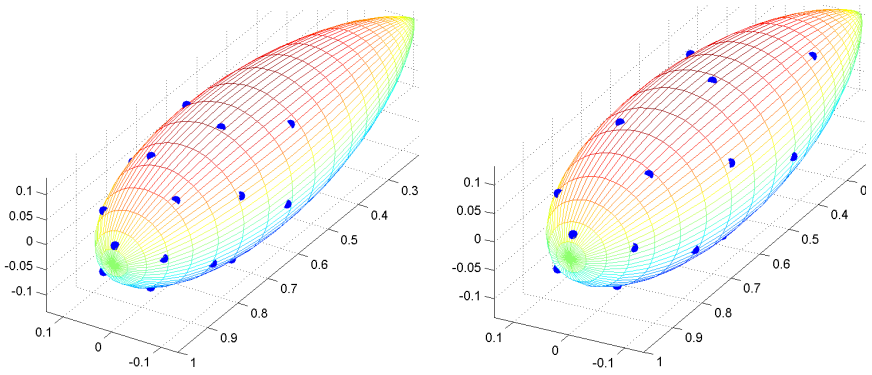
Note that the use of an optimized sampling pattern is beneficial to fully exploit the BMC method. It allows accounting for the prior information about the unknown function L_i on the samples distribution. This aspect is detailed in Section 5.

4.4 The z warping function for Gaussian lobe

Our approach to this problem is to exploit the property of every function f on S^2 that:

$$\lim_{n\to\infty} \frac{1}{n} \sum_{j=1}^{n} f(\boldsymbol{\omega}_{j,n}) = \frac{1}{4\pi} \int_{S^2} f(\boldsymbol{\omega}) d\Omega(\boldsymbol{\omega})$$

for configurations of asymptotically uniformly distributed sampling directions $\{\boldsymbol{\omega}_{1,n}, \ldots, \boldsymbol{\omega}_{n,n}\}$ [SK97]. Therefore, at the beginning of the optimization algorithm, if the initial sampling pattern is such a configuration, all samples can contribute equally to the integral estimate, which can be considered as a sensible choice for a starting point. Our goal in this section is to show how such configurations can be built with the spiral



(a) $l = 0.05$ (b) $l = 0.17$

Figure 3.5 – Optimal sampling patterns of size 20 computed for two different length-scale values and a purely glossy Phong BRDF with a shininess $m = 50$. Each blue point on the lobe corresponds to a sample direction.

points algorithm. In Section 4.3, we have assumed that the lobe central axis is aligned with the surface normal ($\boldsymbol{\omega}_r = \mathbf{n}$) and consequently, using the spherical Gaussian BRDF expression given by Eq. (3.19), the illumination integral of Eq. (3.16) becomes:

$$L_o(\boldsymbol{\omega}_o) = k_s \int_{\Omega_{2\pi}} L_i(\boldsymbol{\omega}_i) \exp[m(\mathbf{n} \cdot \boldsymbol{\omega}_i - 1)] \, d\Omega(\boldsymbol{\omega}_i) \, .$$

This equation can be further developed as follows:

$$L_o(\boldsymbol{\omega}_o) = k_s \int_{\phi=0}^{2\pi} \int_{\theta=0}^{\frac{\pi}{2}} L_i(\theta, \phi) e^{m(\cos\theta - 1)} \, \sin\theta \, d\theta d\phi$$

where (θ, ϕ) are the polar coordinates of the $\boldsymbol{\omega}_i$. If we make the substitution $z = \cos\theta$, we obtain:

$$L_o(\boldsymbol{\omega}_o) = k_s \int_{\phi=0}^{2\pi} \int_{z=0}^{1} L_i(z, \phi) e^{m(z-1)} \, dz d\phi$$

If we directly sample the function $L(z, \phi)e^{m(z-1)}$, the distribution over the hemisphere $\Omega_{2\pi}$ must be uniform as stated above, which is precisely what the original spiral points algorithm does. However, if we sample the function $L(z, \phi)$ only, another substitution is necessary to include the BRDF factor into new integration variables. This is possible by making the substitution:

$$z' = g^{-1}(z) = \frac{e^{mz} - 1}{e^m - 1} \tag{3.26}$$

and then:

$$dz' = \frac{m}{1 - e^{-m}} e^{m(z-1)} dz$$

which leads to:

$$L_o(\boldsymbol{\omega}_o) = \frac{k_s(1 - e^{-m})}{m} \int_{\phi=0}^{2\pi} \int_{z'=0}^{1} L'_i(z', \phi) \, dz' d\phi,$$

where $L'_i(z', \phi) = L_i(g(z'), \phi)$. Note that the integration domain remains the hemisphere $\Omega_{2\pi}$. This domain is then uniformly sampled by (z', ϕ). Nevertheless we still

need to revert to the original z coordinate so that we can sample the original $L_i()$ function. By inverting Eq. (3.26), we have:

$$z = g(z') = \frac{1}{m} \ln[1 + z'(e^m - 1)]$$

which gives the warping function we need for our spiral point algorithm. Its effect can be interpreted as a warping of the distance measure so that inter-samples distances are measured between points on the Gaussian lobe rather that on the unit sphere.

4.5 The rendering algorithm

The BMC integrator described in Alg. 3 is applied to every shading point. The BMC function (line 1) has two parameters: the outgoing direction ω_o and the surface normal $\mathbf{n}$. It returns the BMC estimate of $L_o(\omega_o)$ as defined in Eq. (3.16). All global data for the whole scene including the hyperparameters (l, σ'_n), the inverse covariance matrix Q'^{-1}, the optimal sampling pattern SS and the BRDF parameters (w, k_s) are assumed to belong to the program global scope and are thus not explicitly declared.

Algorithm 3 The Bayesian Monte Carlo integrator.

1: **function** BMC($\omega_o, \mathbf{n}$)
2: $SS_{rotated} \leftarrow$ rotateRandom$\phi(SS)$
3: $\omega_r \leftarrow$ reflect$(\omega_o, \mathbf{n})$
4: $SS_{rotated} \leftarrow$ centerAroundR$(SS_{rotated}, \omega_r)$
5: **for all** $\omega_j \in SS_{rotated}$ **do**
6: $\mathbf{Y}[j] \leftarrow L_i(\omega_j)$
7: **end for**
8: $\bar{L}_i \leftarrow$ computeWeightedMean$(\mathbf{Y})$
9: $\bar{I} \leftarrow \bar{L}_i \times \mu(\omega_r)$
10: $\bar{\mathbf{F}} \leftarrow \bar{L}_i \times \mathbf{H}$
11: $\mathbf{z}' \leftarrow$ computeZVector$(\omega_r, w, SS_{rotated})$
12: $L_o \leftarrow \bar{I} + \mathbf{z}'^t \times Q'^{-1} \times (\mathbf{Y} - \bar{\mathbf{F}})$
13: **return** L_o
14: **end function**

The optimal sampling pattern, initially centered around the surface normal $\mathbf{n}$, is rotated around $\mathbf{n}$ (line 2) by a random ϕ value. The idea behind this random rotation is to introduce a scrambling effect that allows masking the visibility of the regular sampling pattern, hence avoiding the introduction of artifacts in the final image. Note that since the covariance function is a SGF dependent only on the inter-samples distance, it is invariant to rotations of the whole sampling pattern. This property allows generating different optimal sampling patterns by randomly rotating the original optimal sampling pattern while keeping the same matrix Q'^{-1}. The mirror incident direction $\boldsymbol{\omega}_r$ is computed at line 3. An additional rotation aligns the central axis of the sampling pattern with $\boldsymbol{\omega}_r$ (line 4). The loop beginning at line 5 numerically evaluates the samples value for each sampling direction $\boldsymbol{\omega}_j$. $\bar{L}_i$ is computed at line 8 according to Eq. (3.23) and $\bar{I}$

is computed at line 9 according to Eq. (3.24). At line 11, $\mathbf{z}'$ is computed as described in Section 4.1 and Eq. (3.21) through a look up in a scene-independent precomputed table of SGI values. Finally, the Bayesian quadrature (Eq. (3.14)) is applied to estimate L_o, which amounts to a vector-matrix and a vector-vector product.

As the sampling pattern SS is centered around $\boldsymbol{\omega}_r$, a part of the sampled BRDF lobe may be located under the plane tangent to the object surface at the shading point; in other words, some of the directions of the sampling pattern may be occluded by object surfaces. One way of reconstructing the signal would be to only use the samples within the integration domain. This would imply recomputing and inverting the covariance matrix each time one or more samples of the samples set lie outside the domain of integration. To avoid this computational overhead, we use all the samples (even those outside the integration domain), which raises the problem of assigning a value to the occluded samples.

The values of the occluded samples could be naively set to zero, but such a method could potentially introduce wrong information with an important effect on the estimated integral value (Fig. 3.6 (a)). The reason for this is that although the considered sampling direction $\boldsymbol{\omega}_j$ is outside the integration domain, i.e., $\boldsymbol{\omega}_j \notin \Omega_{2\pi}$, it may still contribute to the reconstruction of the function $L_i(\boldsymbol{\omega}_i)$ within the integration domain. Setting the value of such samples to zero would introduce wrong information and force the reconstructed signal to tend towards zero when approaching the surface tangent plane. Indeed, there is no reason why an artificial discontinuity of the incident radiance should be introduced at the tangent plane since occlusion due to the surface is already accounted for through the choice of the integration domain. We have experimentally found that an efficient solution to this problem is to assign to such samples the value of the nearest sample that lies in the unoccluded part of the lobe (Fig. 3.6 (b)). In particular, this solution performs better than assigning the occluded samples the mean value of the samples within the integration domain.

5 Results

5.1 Experimental setup

The goal of this section is to assess the effectiveness of the BMC method with optimized sample sets. Its results are compared with those of BRDF-based low-discrepancy QMC using a Sobol $(0, 2)$-sequence [Sob67] and also using spherical Fibonacci point sets (SF) as described in the previous chapter. We refer to these two methods as QMC Sobol and QMC SF respectively. Moreover, we have also evaluated BMC using a Sobol $(0, 2)$-sequence (BMC Sobol) and BMC using spherical Fibonacci point sets (BMC SF). The efficiency of each method is computed in terms of root mean square error (RMSE). The reference images are computed using QMC SF with 10000 samples per pixel. We have chosen to test our framework using Phong shininess coefficients m varying from 10 to 200, a range of values which we consider illustrative of the common usage of glossy BRDFs. Six different scenes have been used, the details of which are presented in Tab. 3.2. In some scenes (i.e., Dragon, Buddha and Horse), for the sake

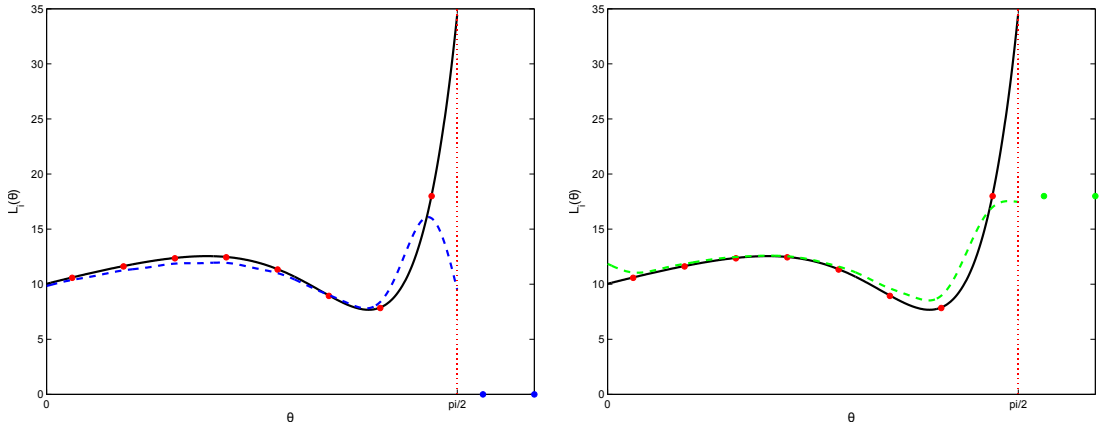


Figure 3.6 – Two examples of a Bayesian regression using occluded samples. On (a) the value of the occluded samples is set to zero, while on (b) their value is set to the value of the nearest unoccluded sample. The red dashed line at $\pi/2$ marks the limit of the integration domain. The samples at the right of this line are thus occluded samples (marked in blue in (a) and green in (b)). The real signal is marked by a black line. The resulting reconstructed signal is represented by a dashed blue line on the left, and a dashed green line on the right.

Scene	triangles	m	l_l	σ'_{nl}
Dragon	100K	20	0.29	0.96
Buddha	170K	50	0.17	0.53
Horse	110K	80	0.13	0.67
VW	440K	20, 50, 80	nd	nd
Room	540K	20, 50, 80	nd	nd
Cornell Box	110K	80, 200	nd	nd

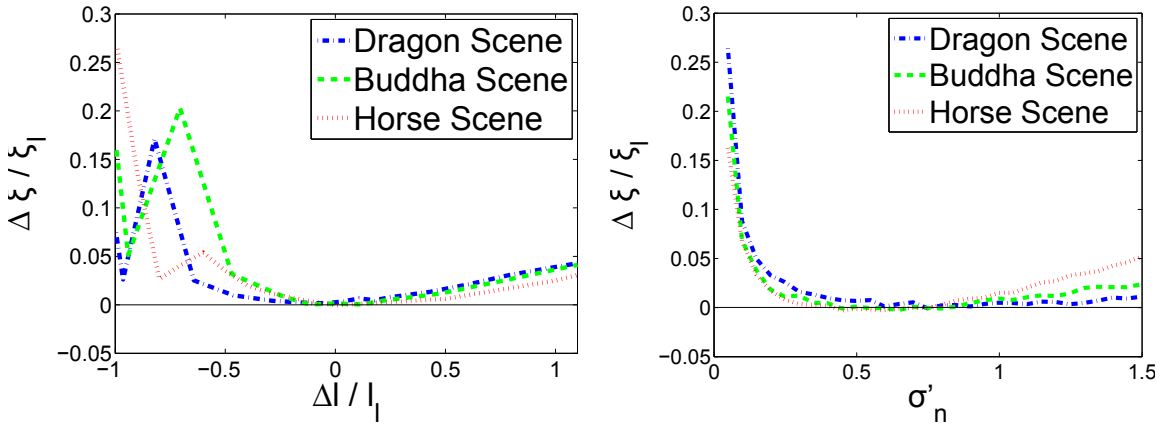
Table 3.2 – Test profiles. Columns list the scene characteristics, the Phong shininess parameter m and the learned hyperparameters l_l, σ'_{nl}. Values of l_l and σ'_{nl} tagged as 'nd' indicate that the approximate method of hyperparameters is used, as shown in Section 5.4.

of computation speed, incident illumination is simulated by an environment map, a simplification that does not limit the significance of our results and conclusions since our goal is to show that we can obtain better estimates with less samples and a negligible computing overhead whatever the used global illumination method. In other scenes (i.e., VW, Room and Cornell Box), the results are generated using final gathering for photon mapping. The results were generated with the Mitsuba raytracer [Jak10] at a resolution of 1024 × 1024 pixels for all the scenes, using a 64 bit machine equipped with a 2GHz Intel Core i7 processor and 8GB of RAM.

5.2 Hyperparameters learning

The validity of our prior model depends on the good choice of the hyperparameters. Tab. 3.2 lists the hyperparameter values obtained by learning for the scenes Dragon, Buddha and Horse. The learning method provides us with the hyperparameters that

allow the best global adaptation of the GP model. But it is not guaranteed that these hyperparameters yield the best integral estimate. To validate our hyperparameters learning method, we have measured the RMSE for different values of the hyperparameters (see Fig. 3.7).



(a) Lengthscale l (b) Noise ratio σ'_n

Figure 3.7 – Relative RMSE difference when varying one of the hyperparameters. $\Delta\xi = \xi - \xi_l$, ξ being the RMSE of an image computed when varying one hyperparameter, and ξ_l being the RMSE of an image computed with the learned hyperparameters (listed in Tab. 3.2). In (a) σ'_n is fixed to its learned value, while in (b) the lengthscale l is assigned its learned value. The abscissa axis of the lengthscale plot (a) is the relative difference between the tested lengthscale and the learned value lengthscale l_l.

Let l_l and σ'_{nl} be the learned hyperparameters listed in Tab. 3.2. To generate the RMSE plots of Fig. 3.7(a), we have fixed the value of σ'_n to σ'_{nl} and computed the RMSE for different values of l. We have repeated this process for the plot of Fig. 3.7(b) by varying the value of σ'_n while keeping $l = l_l$. Fig. 3.7(a) shows that the RMSE is minimum around $\Delta l = l_l - l = 0$ for the three scenes, which means that the learned lengthscale value l_l is appropriate for integration. Furthermore, Fig. 3.7(b) shows that we can draw the same conclusion for σ'_{nl}. Note that the RMSE is more sensitive to negative than to positive deviations from the optimal value. This low sensitivity to a variation of the hyperparameter values around their optimal values might give room for skipping the learning step by using approximated hyperparameters instead of optimal values. A more detailed analysis of this interesting feature will be presented in Section 5.4.

5.3 Comparison with QMC

Fig. 3.8 illustrates the results obtained with QMC SF and BMC (using optimized sample sets) for three scenes with a varying number of samples. The rendering times of both methods are similar. This is because BMC with a single BRDF entails a negligible overhead compared to QMC. The close-up views demonstrate that BMC achieves a reduction of high frequency noise as compared to QMC SF for the same number of samples. The RMSE is also reduced.

Fig. 3.9 shows the RMSE plots as a function of the number of samples for QMC Sobol, BMC Sobol, QMC SF, BMC SF and BMC (with optimal sample sets), for the scenes shown in Fig. 3.8. It can be observed that the RMSE of BMC is consistently lower than that of the other methods. This result confirms the advantage of BMC compared to QMC SF observed in Fig. 3.8. The impact of the BMC samples weighting can be appreciated by comparing the results achieved by BMC and QMC when using exactly the same sample sets. This impact is particularly visible when comparing the results obtained with QMC Sobol and BMC Sobol for the Dragon scene, as well as when comparing QMC SF and BMC SF for the other two scenes. As for the benefits of using optimized sample sets, it can be assessed by comparing the curves corresponding to BMC SF with those of BMC Opt. This result shows the importance of optimized point sets for BMC. It is only through an appropriate samples distribution that the information brought by the prior covariance function can be efficiently exploited.

A thoroughly analysis of the results can be made by resorting to Tab. 3.3. The values in the RMSE columns presented in this table are computed relative to the RMSE of BMC with 80 optimally distributed samples. For a method M, its relative RMSE is given by:

$$\frac{|\mathrm{RMSE}_{BMC} - \mathrm{RMSE}_{M}|}{\mathrm{RMSE}_{BMC}} \times 100\,.$$

The number of rays needed by each of the methods to achieve the same RMSE quality as BMC is also presented relatively to the 80 samples used by BMC. The absolute number of rays needed for the other methods to achieve the same RMSE as BMC with 80 optimal samples has been obtained by extrapolating the RMSE curves in Fig. 3.9. The results show that the current state-of-the-art method for QMC spherical integration (QMC SF, see previous chapter) required from 7.1% up to 9.8% more rays to achieve the same RMSE as BMC using optimal sample sets. The relative RMSE of SF w.r.t. that of BMC ranges from +5.1% to +7.0% (even better results will be presented in the next section). The improvement registered from BMC SF w.r.t. QMC SF is due exclusively to better samples weighting, since the used sample sets are exactly the same. The relative RMSE penalty observed with BMC SF is due to the use of suboptimal sample sets. These sample sets do not account for the covariance of the incident radiance.

5.4 Skipping the learning step

A detailed analysis of the sensitivity of the method suggests that BMC can still achieve good performances using approximated values for l and σ'_n, which would allow skipping the learning step. Indeed, Fig. 3.7(b) shows that for our test cases, we can use any value of σ'_n in the interval $[0.4, 1.1]$ without incurring a significant RMSE increase. However, we have observed that when increasing σ'_n, the error tends to increase faster in the areas of high variance of the incident radiance and is thus more conspicuous. We have found experimentally that $\sigma'_n = 0.5$ is a good trade-off.

Fig. 3.7(a) shows that the estimation of the integral value is not very sensitive to positive variations of l. Moreover, Fig. 3.10 demonstrates that the learned values of

	Dragon		Buddha		Horse	
	RMSE	Same quality rays needed	RMSE	Same quality rays needed	RMSE	Same quality rays needed
QMC Sobol	+38.4%	+61.6%(≈ 129)	+30.9%	+47.2%(≈ 118)	+31.8%	+58.5%(≈ 127)
BMC Sobol	+30.0%	+48.6%(≈ 119)	+26.7%	+42.8%(≈ 114)	+28.9%	+49.9%(≈ 120)
QMC SF	+7.0%	+9.8%(≈ 88)	+5.1%	+8.4%(≈ 87)	+5.2%	+7.1%(≈ 86)
BMC SF	+5.3%	+7.2%(≈ 86)	+2.9%	+3.9%(≈ 83)	+2.9%	+4.0%(≈ 83)

Table 3.3 – Comparison of the results obtained using the different tested methods. For each scene, the first column gives the relative RMSE obtained with a given method w.r.t. the RMSE obtained using BMC with 80 optimal samples. The second column shows the relative number of rays required by a given method to achieve the same RMSE as BMC with 80 sample rays. In-between parentheses is the corresponding absolute number of rays.

lengthscale exhibit a dependence on the shininess parameter m, such that the learned lengthscale $l_l \approx \alpha/\sqrt{m} = \alpha \times w$ where $m = 1/w^2$ as defined in Section 4.1.

A linear regression on the learned lengthscale values yields $\alpha \approx 1.25$ (see Fig. 3.10). To test this fast hyperparameter derivation method, we have compared the performances of BMC without hyperparameters learning with those of QMC SF and QMC Sobol. We have used three different scenes: VW, Room and Cornell Box. The results are shown in Figs. 3.11, 3.12 and 3.13. The relative RMSE of QMC Sobol and QMC SF w.r.t. that of BMC is of 0.1%, 13.5% and 18.9% for Figs. 3.11, 3.12 and 3.13 respectively. It is interesting to note that even when the RMSE is roughly the same for BMC and QMC SF (Fig. 3.12), the image produced by BMC has a noise distribution which is visually more pleasant. This can be explained by the fact that in BMC the GP acts as a low-pass filter which smooths out high frequencies from the integrand without introducing low frequency artifacts.

6 Discussion

6.1 Possible improvements

The results we have obtained demonstrate the soundness of our approach. However, the implementation of BMC presented in this chapter is far from exploiting the full potential of the Bayesian approach. Indeed the main advantage of BMC over other methods lies in its ability to incorporate the prior knowledge, which can be obtained in two complementary ways. First by specifying a more accurate mean function for the prior GP with, for example, a rough approximation of incoming radiance using spherical harmonics or a SGF mixture (to this regard, there are some similarities with the control covariate method [LW95]). Second by improving the covariance function adaptation. In particular, the hyperparameters can be adapted locally to better fit the local radiance function characteristics. For example, different classes of pixels can be distinguished based on the samples variance. In this way, different hyperparameters

can be chosen for each class. Indeed, the pixels with high variance have generally more impact on the visual quality and should be assigned proper hyperparameters value whereas the hyperparameters choice is much less critical for low variance pixels. Since these latter pixels represent the great majority, this explains the low RMSE sensitivity to hyperparameters variation observed in Fig. 3.7.

6.2 BMC with many complex BRDFs

In this chapter, we have focused our attention to single spherical Gaussian BRDF since it is the basic component in our approach to illumination integral computation. In Section 4.1 we have given some indications on how to generalize our approach to any type of BRDF, either analytical or measured, but further research work is necessary to develop an efficient implementation. In the following, we try to be more specific on the different possible approaches. If a BRDF is approximated by a weighted sum of n SGFs, it is easy to see that the illumination integral can be broken down into a weighted sum of n integrals of the same type as Eq. (3.16). Our method can then be applied separately to each term of the sum and each integral will be assigned a samples set of size proportional to its weight. However, this straightforward method cannot be optimal in terms of sampling efficiency since each integral estimate will not benefit from the information brought by the samples used in the other integrals. An alternative to this method is then to use a single samples set for all the BRDF lobes. In this case, only one global covariance matrix is necessary for computing the integral estimate with Eq. (3.14). In Eq. (3.15), since the function $p(\mathbf{x})$ that represents the BRDF is expressed as a weighted sum of SGFs $\sum_j \alpha_j G_j()$, the $\mathbf{z}'$ vector in Eq. (3.14) will also be expressed as a weighted sum of vectors $\sum_j \alpha_j \mathbf{z}'_j$ and each $\mathbf{z}'_j$ can be computed with the method described in Section 4.1. As for samples set optimization, the morphing function can be built from an approximation of the BRDF leading to a simpler analytical expression.

As mentioned above, one of the main problems to solve in BMC lies in the computation of the $\mathbf{z}'$ vector and the complexity of this computation depends on the choice of the deterministic function $p(\mathbf{x})$ in Eq. (3.1). So far, we have considered in a rigid manner that this function corresponds exactly to the BRDF but actually this choice can be made more flexible since $p(\mathbf{x})$ does not need to be an accurate approximation of the BRDF. If a function $g(\mathbf{x})$ (e.g. a weighted sum of SGFs) is a quite raw approximation of $p(\mathbf{x})$, it is possible to consider that the unknown function is $f(\mathbf{x})p(\mathbf{x})/g(\mathbf{x})$ and substitute $g(\mathbf{x})$ to $p(\mathbf{x})$ in Eq. (3.15). This will not change very much the behavior of the unknown function since the factor $p(\mathbf{x})/g(\mathbf{x})$ is generally a very smooth function but it could greatly simplify the computation of the $\mathbf{z}'$ vector for complex BRDFs.

With respect to the application of BMC for scenes with a large number of different BRDFs, the single problem that could arise is the need for precomputing an optimized samples set for each shininess coefficient. This operation is much lighter than learning the hyperparameters and is scene independent. But still, if one wants to avoid performing this operation for each different shininess coefficient, the same strategy as the one applied for the hyperparameters could be used: optimize the sample sets for a subset of

shininess values (which results in a polynomial of degree 3 as described in Section 4.3) and interpolate the values of these polynomials for the intermediate values.

6.3 Limitations

As explained in this chapter, the main difficulties that we are faced with in implementing BMC lie in the computation of the inverted covariance matrix and the $\mathbf{z}'$ vector. Our proposal in this chapter is to solve this problem by precomputing optimized points sets and using spherical Gaussian mixtures. As far as this solution is applicable, there is no reason why BMC should not perform better that CMC and QMC methods.

Directly applying our framework to the case of a very large samples set would increase the memory footprint and make costly the preprocessing step because of the inversion of the covariance matrix and the samples set optimization. One possible solution is to resort to a progressive approach which would refine the integral estimate by using successive small sample sets. At each iteration, the prior GP for the current estimate (for the current samples set) would be replaced by the posterior GP of the previous estimate (for the previous samples set). In this way the estimate is progressively refined.

The extension of BMC to larger dimensionality, such as illumination integral calculations requiring image plane sampling, lens sampling and multiple reflections, is not obvious. Indeed, finding an appropriate prior model (definition of the known and uncertain functions, determination of the hyperparameters and the mean function) while keeping a good trade off between efficiency and performance is a difficult task.

7 Conclusion

In this chapter, important steps are given towards the goal of reaching a general Bayesian Monte Carlo approach for solving global illumination problems, which was until now limited to the diffuse BRDF case. Modeling the BRDF and the covariance function with the same family of spherical Gaussian functions is a key point, since it allows making the computation of the Bayesian quadrature coefficients tractable for glossy BRDFs. The results confirm that Bayesian Monte Carlo outperforms state-of-the-art Quasi-Monte Carlo methods. In a context where rendering times are dominated by the cost of sampling, Bayesian Monte Carlo has a clear advantage over low discrepancy Monte Carlo importance sampling since it exhibits less noise and lower RMSE for the same number of samples, while having a negligible overhead. Finally, we demonstrated that Bayesian Monte Carlo outperforms Quasi-Monte Carlo importance sampling even with a suboptimal hyperparameters setting that avoids the learning step. Future work will aim at improving the efficiency of Bayesian Monte Carlo as explained above and also at extending its use to other problems of larger dimensionality, e.g., measured BRDFs, multiple reflections, depth blur, multi-view rendering, among others.

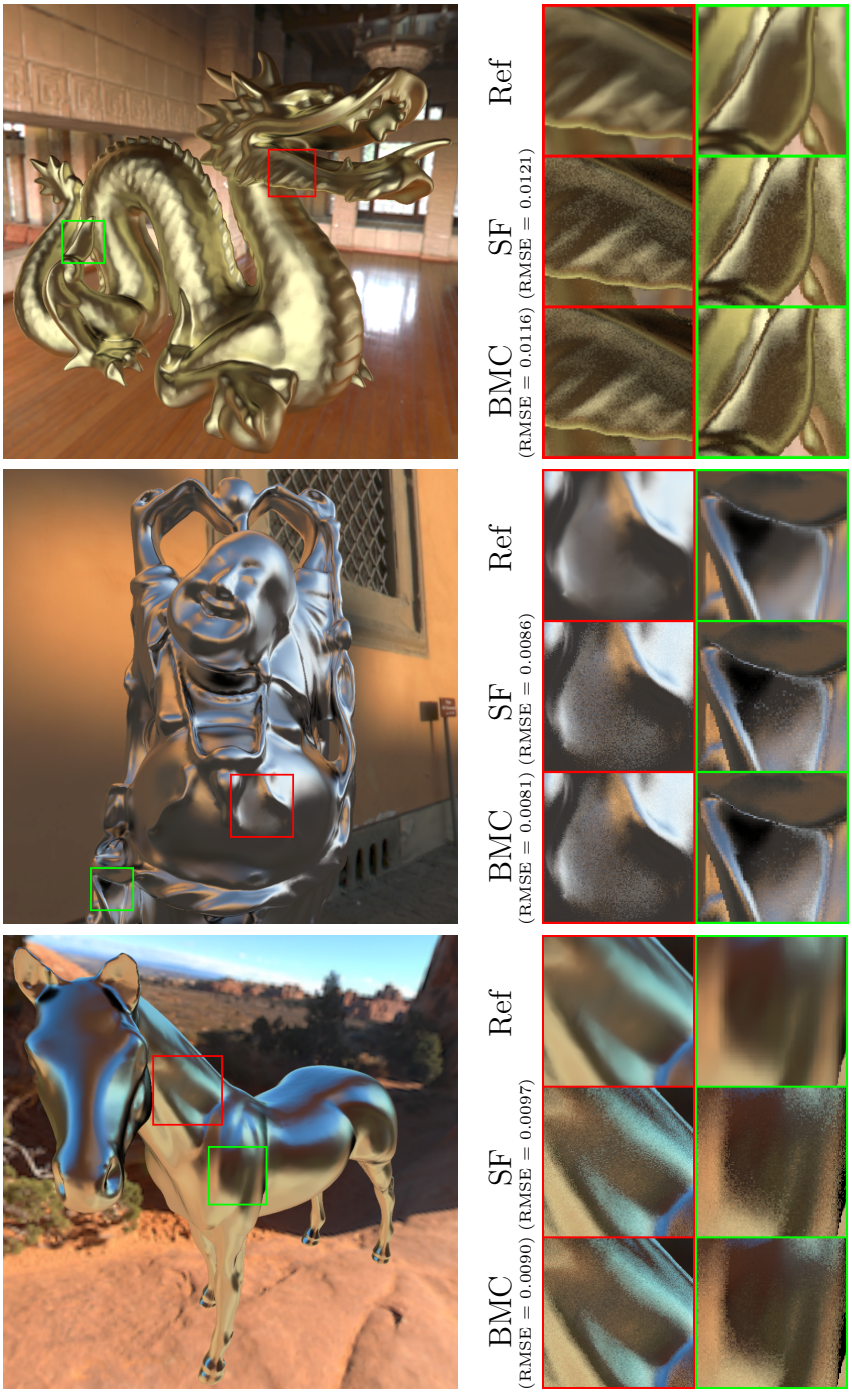


Figure 3.8 – Close-up views for three scenes rendered with Bayesian Monte Carlo (BMC) and low discrepancy importance sampling (LDIS). The number of ray samples per visible point is 40 for the Dragon and the Buddha, and 20 for the Horse. Each of these objects has a purely glossy Phong BRDF with a shininess parameter of 20, 50 and 80 respectively. The figure also shows the RMSE values for each method.

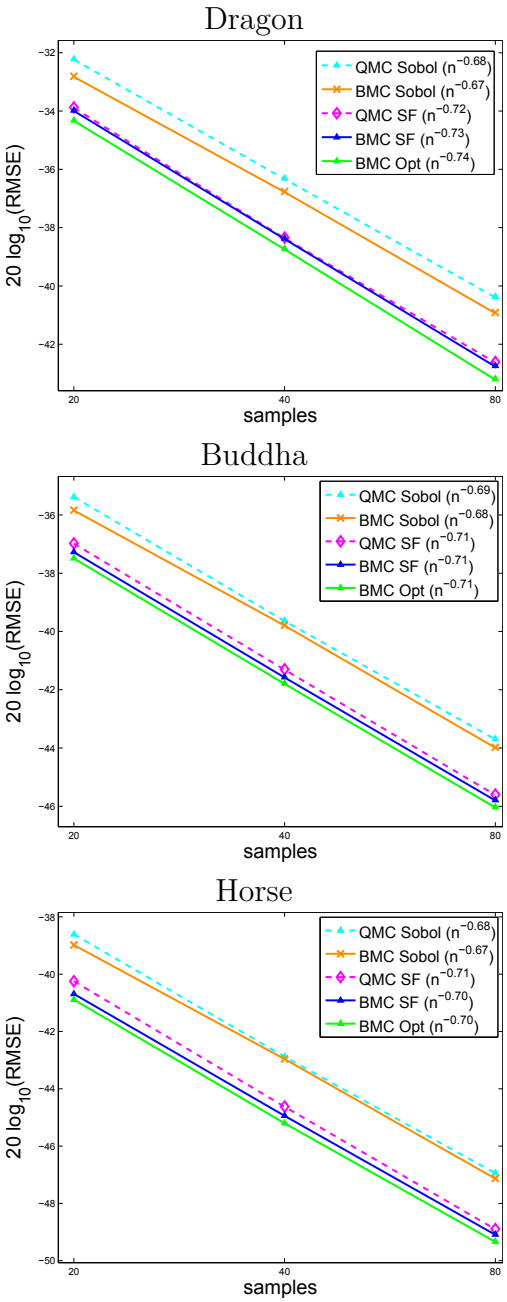


Figure 3.9 – RMSE plots as a function of the number of samples n. The slopes of the line fit for the RMSE curves of each method are given in the legend in-between parentheses.

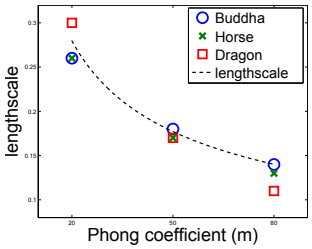


Figure 3.10 – Fitting of the learned lengthscales as a function of the shininess parameter m.

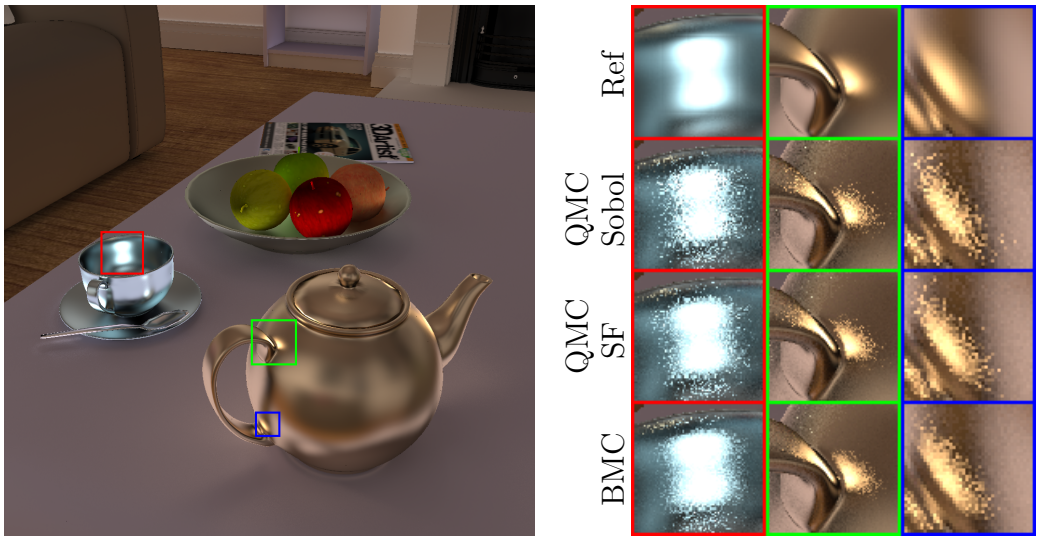


Figure 3.11 – Indirect radiance component for the Room scene rendered with BMC, QMC SF and QMC Sobol. The RMSE of BMC is 0.026. The relative RMSE of QMC Sobol and QMC SF w.r.t. that of BMC is +50.0% and +13.5% respectively. The rendering times are 2m20s for BMC, and 2m18s for the QMC-based methods. The images have been multiplied by a factor of 4. The RMSE and the time are computed by considering only the glossy component. 16 ray samples per visible point were used for the materials with a glossy BRDF (teapot, tea cup and fruit-dish), while 10000 samples were used for the diffuse BRDFs to ensure convergence since these are not computed using our spherical Gaussian framework.

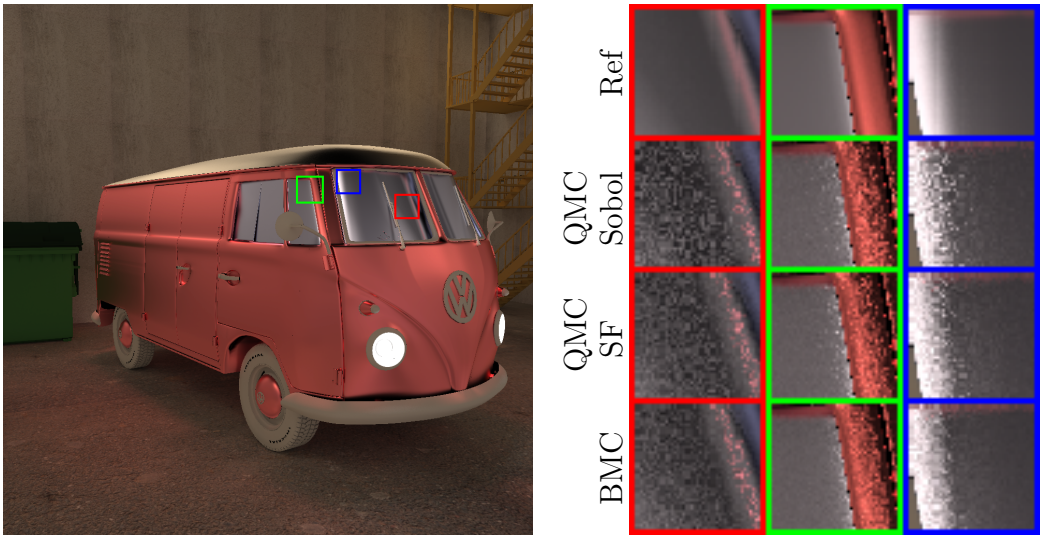


Figure 3.12 – Indirect radiance component for the VW scene rendered with BMC, QMC SF and QMC Sobol. The RMSE of BMC is 0.020. The relative RMSE of QMC Sobol and QMC SF w.r.t. that of BMC is +17.7% and +0.1% respectively. The rendering times are 4m11s for BMC, and 4m09s for the QMC-based methods. The shown images have been multiplied by a factor of 4. The RMSE and the time are computed by considering only the glossy component. 16 ray samples per visible point were used for the materials with a glossy BRDF (VW glass, VW bodywork and VW roof), while 10000 samples were used for the diffuse BRDFs to ensure convergence since these are not computed using our spherical Gaussian framework.

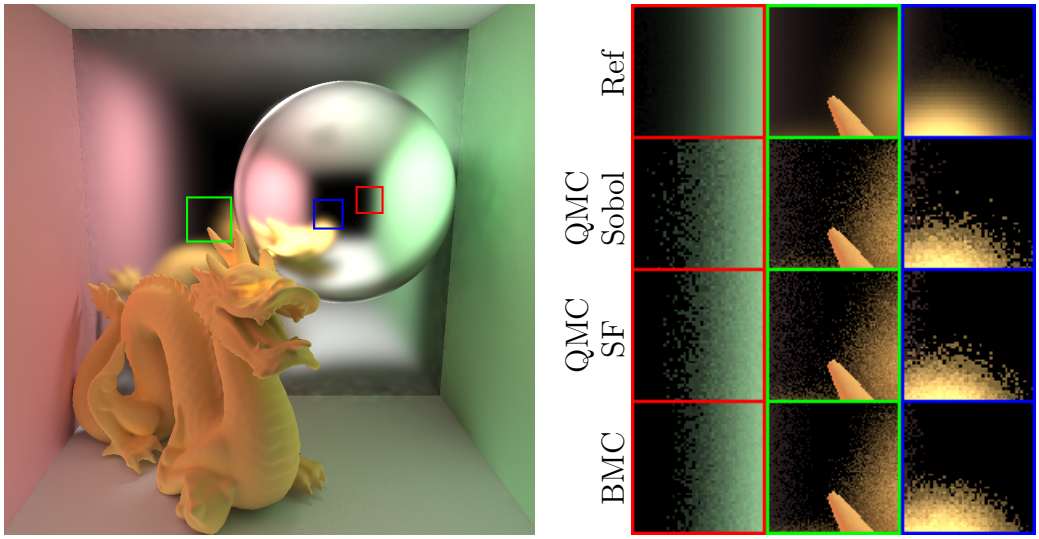


Figure 3.13 – Indirect radiance component for the Cornell Box scene rendered with BMC, QMC SF and QMC Sobol. The RMSE of BMC is 0.022. The relative RMSE of QMC Sobol and QMC SF w.r.t. that of BMC is +71.7% and +18.9% respectively. The rendering times are 3m38s for BMC, and 3m36s for the QMC-based methods. The shown images have been multiplied by a factor of 4. The RMSE and the time are computed by considering only the glossy component. 16 ray samples per visible point were used for the materials with a glossy BRDF (sphere and back wall), while 10000 samples were used for the diffuse BRDFs to ensure convergence since these are not computed using our spherical Gaussian framework.

Adaptive Sampling with Bayesian Monte Carlo

4

Contents

1 Introduction

In global illumination, rendering an image is a very computationally demanding task, whose execution time depends on the overall number of samples used to estimate the illumination integrals. A direct approach to render an image consists in assigning a fixed number of samples to each integral evaluation. Such a choice is not optimal since different image regions may correspond to different types of illumination and, therefore, require different numbers of samples to achieve the same quality.

A common approach to solve this problem is to use adaptive sampling. When using adaptive sampling, a first estimate with few samples is computed. Then, if some quality criterion of the estimate is not met, a new set of samples is drawn and used to refine the previous estimate. Consequently, regions of the image which correspond to sharper variations of the sampled signal will tend to be more densely sampled.

Our goal in this chapter is to apply BMC with an adaptive approach for evaluating the illumination integral. The idea is to compute a first BMC estimate based on a first samples set and, if the quality criterion is not met, directly use the result as a prior knowledge for a new estimate. The new estimate refines the previous one using a new set of samples, and the process is repeated until a satisfying result is achieved.

2 Related work

Most of the work regarding adaptive sampling and refinement criteria has been developed for image plane sampling (possibly in the spatio-temporal domain such as MDAS [HJW+08]). The main goal of these works is to detect regions of the image plane for which the sampling rate does not allow an adequate signal reconstruction. Once these regions are detected, their sampling rate is increased to remove aliasing. Finally, an image is produced by combining the samples, either using a simple average or a filter whose support can be adapted to the resolution at which the image will be displayed.

Lee et al. [LRU85] and later Purgathofer [Pur87] have developed adaptive sampling algorithms which determine whether or not local oversampling is necessary based on the variance of the estimate of the pixel values. The samples are normally distributed around a pixel center, and if the variance of the estimate does not meet the required quality criterion the local sampling rate is increased.

In [Mit87], Mitchell has proposed a 2-level sampling method. To generate the appropriate sample sets, he uses a point-diffusion algorithm which generates point sets having spectral characteristics similar to those of blue noise point sets. As regards the refinement criterion, Mitchell [Mit87] proposed a contrast metrics which operates per RGB color channel. This contrast function leverages the work about human visual perception of Caelli [Cae81], and is given by:

$$C = \frac{I_{\max} - I_{\min}}{I_{\max} + I_{\min}}$$

If the contrast, measured on a color channel, is larger than a given threshold then the estimate is refined. Mitchell suggests to use thresholds adapted to the human visual system for each of the color channels: 0.4, 0.3 and 0.6 for the red, green and blue respectively.

The above cited works present complete solutions to the adaptive sampling problem. Several other works focus only on the refinement criterion. In particular, in [RFS02, RFS03], the *Shannon entropy* is used as a criterion for oversampling. Entropy can be viewed as a measure of the uncertainty associated with a random variable. Given a discrete random variable X and a set of realizations $\{x_0, \ldots, x_n\}$, its entropy is defined as [Sha48]:

$$H(X) = -\sum_{i=0}^{n} p_i \log_2(p_i)$$

where $p_i = p(X = x_i)$ and, by convention, $0 \log_2(0) = 0$. Rigau et al. [RFS03] use a combination between the samples color entropy and the samples geometric entropy to

derive a contrast metrics which is used as a refinement criterion. Later, a different type of entropy, called *Tsallis entropy*, has also been used as criterion [XBZ+05, XFSS07, XSFS10]. These methods using these criteria rely on the sampled values to compute the uncertainty of the estimate and use it to decide if the estimate should be refined.

In global illumination, the goal of sampling is not to accurately reconstruct the incident radiance functions, which would require a huge number of samples. Generally, much less samples are required to reach an acceptable accuracy when it comes to illumination integral estimation. Despite these differences, there are obvious similitudes with adaptive sampling for signal reconstruction and adaptive sampling for integral evaluation. Both methods require a refinement criterion which, based on the information brought by the samples already taken, determines if a higher sampling rate is needed. Moreover, an efficient samples distribution, which guarantees that at each step new samples are not too close to previous ones, is also required.

Our approach is different from the above presented related work in two points:

- the oversampling does not concern the image plane, but the incident directions space (to compute the incoming radiance at a surface point as seen from a pixel);
- the posterior of iteration $n-1$ is used as prior for iteration n (where the adaptive process consists of several iterations).

3 Overview

The steps required for our approach can be summarized as follows:

1. Draw a set of samples for the first estimate.
2. Apply BMC with a constant mean function and a stationary covariance function as if it were a single step method.
3. While the quality criterion is not met and the maximum number of iterations is not reached:
   (a) Draw an additional set of samples.
   (b) Take as prior the posterior of the previous estimate.
   (c) Apply BMC.

Throughout this chapter we will formalize the mechanism of adaptive sampling using BMC and propose potential answers to the questions raised by the application of BMC for an adaptive approach. More specifically, we propose solutions to the distribution of the second level samples set, as well as the computation method of the z coefficients that reuses the z coefficients value of the first set.

4 A two-level BMC estimate

In this section we will show how to use the result of a BMC estimate as prior knowledge for a subsequent estimate. For the sake of simplicity, a two-level estimation method will be considered, but the same approach could be extended to the the multi-level case.

The objective is the same as in the previous chapter, that is, the approximation of the value of the integral:

$$I = \int f(\mathbf{x})p(\mathbf{x})d\mathbf{x}\,, \qquad \text{with } \mathbf{x} \in \mathbb{R}^D, \tag{4.1}$$

where $p(\mathbf{x})$ is the analytically known part of the integrand. n_1 and n_2 samples are used for the first and second steps respectively. Let us suppose we are provided with two sets $\mathcal{D}_1$ and $\mathcal{D}_2$ of noisy samples of $f(\mathbf{x})$:

$$\begin{aligned} \mathcal{D}_1 &= \{(\mathbf{x}_1^i, Y_1^i)\,|\,i = 1, \ldots, n_1\}, \\ \mathcal{D}_2 &= \{(\mathbf{x}_2^i, Y_2^i)\,|\,i = 1, \ldots, n_2\}, \quad \text{with} \quad Y_L^i = f(\mathbf{x}_L^i) + \varepsilon_L^i, \end{aligned}$$

ε_L^i being samples of an independent, identically distributed Gaussian noise with zero mean and variance σ_n^2, $\mathbf{x}_L^i$ being a sample location (the input) of the sample set of level L and Y_L^i its corresponding sample value (the output).

The first level estimate $\hat{I}_1$ of I is computed using the samples set $\mathcal{D}_1$ as described in the previous chapter. To compute the second level estimate $\hat{I}_2$, the posterior GP resulting from the first estimate will be used as prior. The second level GP prior ($\mathcal{GP}_2$) which models the unknown function $f(\mathbf{x})$ is thus defined as:

$$f(\mathbf{x}) \sim \mathcal{GP}_2[\bar{f}_2(\mathbf{x}), k_2(\mathbf{x}, \mathbf{x}')],$$

where

$$\begin{aligned} \bar{f}_2(\mathbf{x}) &= \mathrm{E}[f(\mathbf{x})|\mathcal{D}_1] \\ k_2(\mathbf{x}, \mathbf{x}') &= \mathrm{cov}[f(\mathbf{x}), f(\mathbf{x}')|\mathcal{D}_1], \end{aligned}$$

$\mathrm{E}[f(\mathbf{x})|\mathcal{D}_1]$ and $\mathrm{cov}[f(\mathbf{x}), f(\mathbf{x}')|\mathcal{D}_1]$ being given by Eq. (3.5). Note that compared to the first step, the second step prior is refined by the information brought by the first samples set. Similarly to the first level estimate, the BMC estimate for the second level is given by:

$$\hat{I}_2 = \bar{I}_2 + \mathbf{z}_2^t Q_2^{-1}(\mathbf{Y}_2 - \bar{\mathbf{F}}_2) \tag{4.2}$$

where

$$\begin{aligned} \bar{I}_2 &= \hat{I}_1 \\ \mathbf{z}_2 &= \int \mathbf{k}_2(\mathbf{x})p(\mathbf{x})d\mathbf{x} \\ \mathbf{k}_2(\mathbf{x}) &= (k_2(\mathbf{x}_2^1, \mathbf{x}), \ldots, k_2(\mathbf{x}_2^{n_2}, \mathbf{x}))^t \\ Q_2 &= (K_2 + \sigma_n^2 I_n) \\ K_2^{(i,j)} &= k_2(\mathbf{x}_2^i, \mathbf{x}_2^j) \quad \text{with } (i,j) \in [1, n_2]^2 \\ \bar{\mathbf{F}}_2 &= (\bar{f}_2(\mathbf{x}_2^1), \ldots, \bar{f}_2(\mathbf{x}_2^{n_2})) \end{aligned}$$

While the terms of the Bayesian quadrature equation of level two in Eq. (4.2) have roughly the same role as presented in the previous chapter, the covariance function k_2 is no longer stationary and has a more complex expression as will be seen in Eq. (4.3). $\bar{I}_2$ is the prior integral estimate for the second level. Naturally, its value is set to the posterior estimate $\hat{I}_1$ of the first level. $\mathbf{z}_2$ is a vector of integrals which accounts for the influence of each sample of the second level on the deterministic part of the integrand given the second level covariance function k_2. Q_2^{-1} is the inverse of the covariance matrix between the level two samples.

5 Making adaptive BMC practical

In this section we tackle the most challenging tasks when applying BMC to an adaptive sampling case, that is, the distribution of the second samples set as well as the computation method of the $\mathbf{z}_2$ coefficients.

5.1 Selecting the samples set

As in any Monte Carlo method, a high quality samples distribution is critical for obtaining a good integral estimate using BMC. As seen in the previous chapter, using optimal sample sets (which minimize the variance of the estimate) is beneficial for BMC. Nevertheless, since we are taking an adaptive approach, it is also advantageous that the samples of the current level samples are not too close to the samples taken previously. The intuition is obvious: since the prior model already contains the information brought by the samples used for the previous estimates, sampling again the same positions would bring no extra information. Moreover, the new samples position should be selected so as to minimize the variance criterion that characterizes the confidence we have in the integral estimate.

Sobol $(\mathbf{0}, \mathbf{2})$-sequence An obvious candidate for generating samples for the different levels which are not too close to each other are the $(0, 2)$-sequences presented in Chapter 2. These sequences allow incrementing the number of samples while maintaining the quality of the distribution. Fig. 4.1 shows two examples of two level sample sets using a Sobol $(0, 2)$-sequence. Using a $(0, 2)$-sequence for generating the sample sets guarantees that new samples can be added without superposing to the existent ones. Nevertheless, this implies a suboptimal use of BMC, since such sample sets do not minimize the variance of the estimate, as has been shown in the previous chapter.

Independent optimal sample sets An alternative to the Sobol $(0, 2)$-sequence is to use sample sets which are optimized independently for each level. To this end, we can use the sample sets precomputed in the previous chapter. Such sample sets already account for the information regarding the covariance of the incident radiance function. Nevertheless, since the samples set for the second level is computed independently from the samples set for the first level, this approach has the disadvantage of possibly resampling previously sampled locations.

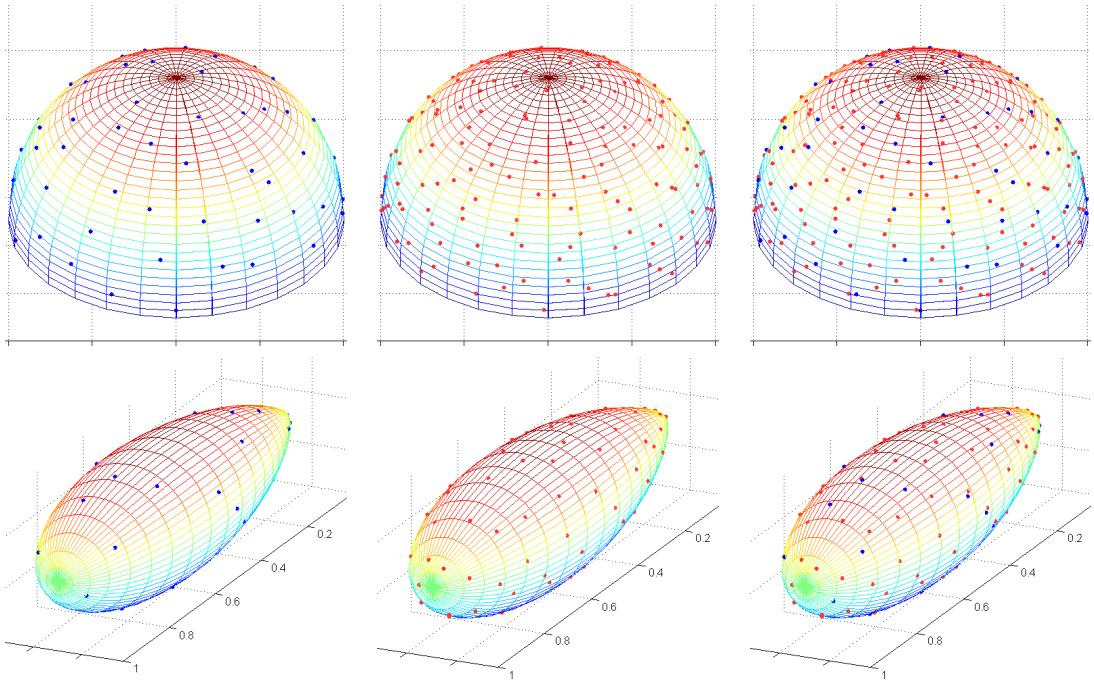


Figure 4.1 – Sample sets for adaptive sampling using a Sobol $(0, 2)$-sequence. The top row shows a unit hemisphere while the bottom row shows a Phong lobe with a shininess factor of 20. The left column shows the first level sample sets, where each sample corresponds to a blue point. In the middle column, the samples of the second level can be seen. At the rightmost column the combination of the sample sets for both levels is depicted.

Incremental optimal sample sets Comparing the two sampling strategies described above can be interesting to determine which of the two following characteristics has a stronger impact on the quality of an adaptive BMC estimate:

- Samples of different levels are not too close to each other.
- An optimal samples distribution in each of the levels.

However, the ideal samples set should combine the two characteristics stated above. Huszár and Duvenaud [HD12] propose a sampling scheme in which the samples position is chosen in a sequential fashion. Each new sample is placed in the position of the domain where it will decrease the most the variance of the estimate (see Eq. (3.10)). We suggest using an approach inspired by that of Huszár and Duvenaud. The idea is to start from the optimal samples set of the first level and add the new samples one by one until we reach the number of samples desired for the second level. Such an approach would not have the problem of superposing samples, and would also respond to a criterion of minimal variance of the estimate. Furthermore, it would allow for local sampling adaptation, but at the cost of extra computations.

5.2 The vector of integrals $\mathbf{z}_2$

Once the sampling locations for both levels are determined, Q_1^{-1} and Q_2^{-1} can be precomputed to accelerate the Bayesian quadrature computation when rendering. Nevertheless, this optimization is not sufficient for an efficient implementation since calculating the vectors of integrals $\mathbf{z}_1$ and $\mathbf{z}_2$ are the most computationally intensive operations. In the previous chapter, we have proposed a solution for calculating $\mathbf{z}_1$ by resorting to a spherical Gaussian-based framework. In the following of this section, we will show how $\mathbf{z}_2$ can be computed using the same general approach.

Let us assume that, similarly to Chapter 3, $k_2(\mathbf{x}, \mathbf{x}')$ and the known part of the integrand $p(\mathbf{x})$ are given by spherical Gaussian functions (see Eqs. (3.18) and (3.19)). The second level covariance $k_2(\mathbf{x}, \mathbf{x}')$ is then given by:

$$k_2(\mathbf{x}, \mathbf{x}') = k_1(\mathbf{x}, \mathbf{x}') - \mathbf{k}_1^t(\mathbf{x}) Q_1^{-1} \mathbf{k}_1(\mathbf{x}') \,. \tag{4.3}$$

Consequently, the i^{th} element of the $\mathbf{z}_2$ vector from Eq. (4.2) can be expressed as:

$$z_2^i = \underbrace{\int k_1(\mathbf{x}_i^2, \mathbf{x}')\, p(\mathbf{x}')\, d\mathbf{x}'}_{\text{SGI}} - \overbrace{\int \underbrace{\mathbf{k}_1^t(\mathbf{x}_i^2)\, Q_1^{-1}}_{\text{constant}}\, \mathbf{k}_1(\mathbf{x}')\, p(\mathbf{x}')\, d\mathbf{x}'}^{T_2^i} \,. \tag{4.4}$$

Note that the leftmost integral of Eq. (4.4) is a spherical Gaussian integral (SGI, Eq. (3.22)). Its value can thus be precomputed using the same strategy presented in the previous chapter, that is, by resorting to a compact precomputed 2-entry table. Recall that this table is only precomputed once and for all, so there is no need to precompute it again for the adaptive BMC case. In the rightmost integral of Eq. (4.4) T_2^i, two terms are constant since they are not dependent on the integration variable $\mathbf{x}'$. These terms can thus be taken out of the integral and T_2^i can be rewritten as:

$$T_2^i = \mathbf{k}_1^t(\mathbf{x}_i^2)\, Q_1^{-1} \underbrace{\int \mathbf{k}_1(\mathbf{x}')\, p(\mathbf{x}')\, d\mathbf{x}'}_{\mathbf{z}_1} = \mathbf{k}_1^t(\mathbf{x}_i^2)\, Q_1^{-1}\, \mathbf{z}_1 \,.$$

The term $\mathbf{k}_1(\mathbf{x}_i^2)$ is a vector containing the covariance $(k_1(\mathbf{x}_i^2, \mathbf{x}_1^1), \ldots, k_1(\mathbf{x}_i^2, \mathbf{x}_{n_1}^1))$ of the new sample $\mathbf{x}_i^2$ with each of the samples $\mathbf{x}_j^1, j \in \{1, \ldots, n_1\}$ of the first level. Since we have assumed that the locations of the level one and level two samples are known, then this vector can be precomputed. The term Q_1^{-1} is the inverse covariance matrix of the first level. This term has been previously precomputed for the first level, so it can just be reused for the second level estimate. Similarly, the $\mathbf{z}_1$ has also been computed for the previous level and can be reused for the second level estimate.

6 Experimental results

This section has two main objectives. The first one is to verify the effectiveness of the adaptive approach. To this end, we compare adaptive BMC (ABMC) with non-adaptive BMC (NABMC) as described in the previous chapter, where both methods

use the same number of samples for computing each image. We will show that ABMC achieves a smaller root mean square error (RMSE) than the non-adaptive version for the same number of samples. Furthermore, we also show that ABMC produces images which are visually more pleasant than those produced with NABMC.

The second goal of this section is to compare ABMC with adaptive Quasi-Monte Carlo (AQMC). Such a comparison will be made by comparing the RMSE of images produced using ABMC and AQMC where the sample sets are exactly the same for both methods. Finally, we evaluate the performance of the adaptive approach for different choices of samples distributions.

6.1 Experimental set up

Equipment The results were generated with the Mitsuba raytracer [Jak10] at a resolution of 1024×1024 pixels, using a 64 bit machine equipped with a 2GHz Intel Core i7 processor and 8GB of RAM.

A simple case test For the moment, the values of the samples are computed from an environment map. The extension to global illumination will be addressed in a future work. The reference images are computed using importance sampling.

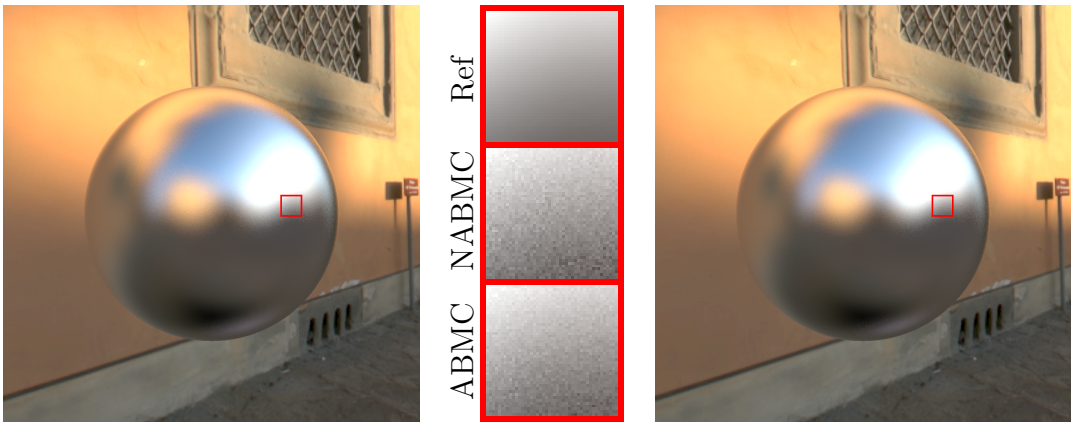
The oversampling criterion The oversampling criterion used is the samples color entropy [RFS03]. Note that this is only a partial implementation of the method described by [RFS03], since the authors also use the samples geometric entropy based on the distance traversed by the ray used to collect the sample value. This geometric entropy is not possible to compute in the context of environment maps, but it will be used in a future work when adaptive BMC will be extended to global illumination. We can thus expect some improvements w.r.t. the oversampling criterion used in this section.

6.2 The effectiveness of BMC adaptive sampling

In this chapter we describe a solution for synthesizing an image using adaptive BMC. Such an approach is motivated by better distributing the samples across the scene, depending on the local characteristics of the incident radiance function. With this approach, we expect to produce better images than with a non-adaptive BMC approach for the same number of samples, i.e., with a lower RMSE and a lower visible noise.

Fig. 4.2 shows a comparison of adaptive and non-adaptive BMC. The number of samples used for NABMC is equal to 60 samples per integral evaluation. As regards ABMC, 40 samples per integral were used for the first sampling level and 40 for the second, making a total of 80 samples for the latter. Since nearly half of the image has been oversampled, ABMC uses in average 60 samples per integral, and consequently a total number of samples roughly equal to that of NABMC.

The close-up views show that ABMC achieves lower visual noise when compared to NABMC. Furthermore, the RMSE of the image produced using NABMC is 12.1% larger



Non-adaptive BMC (NABMC)

Adaptive BMC (ABMC)

Figure 4.2 – Adaptive BMC vs non-adaptive BMC. Each of the methods uses approximately the same number of samples to compute the image (60 samples per integral evaluation). the NABMC uses 60 samples for all integrations while ABMC uses 40 and 80 for not oversampled and oversampled zones respectively. The RMSE of NABMC is 12.1% larger than that of ABMC.

than that of the images synthesized with ABMC. We have also obtained significant quality improvement using different number of samples. As an example, when using NABMC with 30 samples per integral, the RMSE is 8.2% larger than that of ABMC with 20 samples for the first level and 40 for the second (making an average of 30 samples per integral).

6.3 Comparing adaptive BMC with adaptive QMC

Fig. 4.3 shows a test case used to assess the quality of the images generated using adaptive BMC and adaptive QMC. The images have been generated using 40 samples for the first sampling level and 40 for the second level, making a total of 80 samples for the oversampled zones. The samples position is the same for both methods and is generated using a Sobol (0,2)-sequence. The image on the right in Fig. 4.3, shows a sampling map. The zones selected for oversampling are shown in red, while the zones where only 40 samples are taken are shown in green. The close-up views on the center of Fig. 4.3 show a comparison between adaptive BMC and adaptive QMC for a zone of the image which has been oversampled. It can be seen that the BMC-based image is closer to the reference than the QMC-based image. The RMSE of the oversampled zone (in red) for QMC is 6.83% higher than that of BMC for the same zone.

Fig. 4.4 shows different RMSE curves calculated by only taking into account the oversampled zone of the image. The scene used to produce these curves is the sphere scene from Figs. 4.2 and 4.3. The results were generated using 10 samples for the first level of sampling and $n_2 \in \{10, 30, 70\}$ samples for the second level. The lines in the plot correspond to the following methods:

- adaptive QMC using a Sobol sequence;

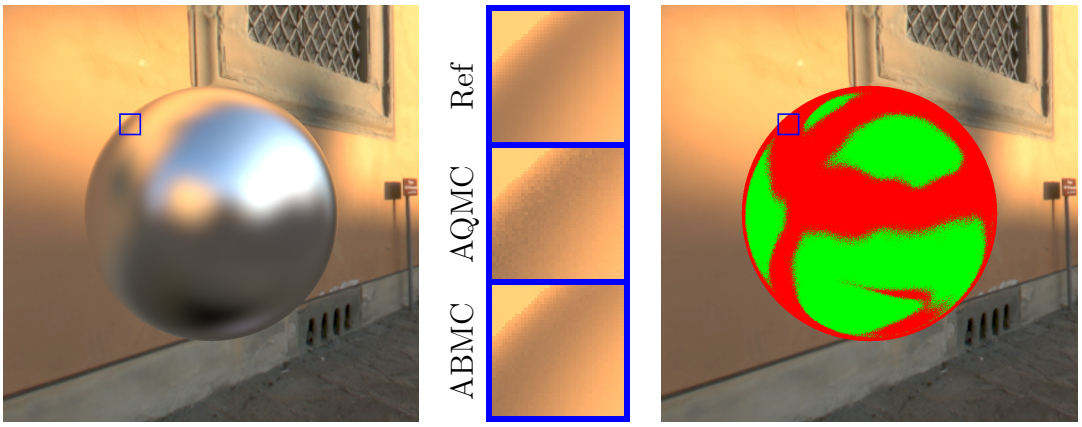


Figure 4.3 – Adaptive BMC vs adaptive QMC. Left: reference image. Right: sampling map. The zones selected for oversampling are colored in red, while the zones not oversampled are colored in green. Center: close-up views showing a comparison between adaptive BMC and adaptive QMC for a zone of the image which has been oversampled. For the oversampled zone (in red at the right), the RMSE of adaptive QMC is 6.83% higher than that of adaptive BMC.

- adaptive BMC using a Sobol sequence;
- adaptive QMC using spherical Fibonacci (SF) point sets;
- adaptive BMC using SF point sets;
- adaptive BMC using the same optimized sample sets as in the previous chapter;

Note that the sample sets used in adaptive BMC with optimal sample sets are independently optimized for each sampling level. This means that, as explained in Section 5.1, the samples positions are not optimal as a whole since the second level samples set is produced without taking into account the samples set from the first level. To alleviate the effect of superposing samples of different levels, the samples set of level two is rotated by an angle of π w.r.t. the first level samples set. The same approach is also taken when using the spherical Fibonacci point sets.

The results show that adaptive BMC outperforms adaptive QMC when using the same sample sets. For example, the RMSE of adaptive BMC is consistently smaller than that of QMC when both methods use a Sobol $(0, 2)$-sequence (roughly 5% lower). The results also show that Fibonacci point sets and optimized sample sets seem to have an advantage over a Sobol sequence. Nevertheless, the use of such an approach should be progressively more and more inefficient as the number of sampling levels increases. This is because the samples from high sampling levels would be more likely to overlap with samples from previous levels. In fact this situation is already present in Fig. 4.4 even if only two sampling levels have been used. When using 40 samples (10 for the first level and 30 for the second) the methods which rely on SF and optimized sample sets are suddenly less efficient compared to the methods using a Sobol sequence.

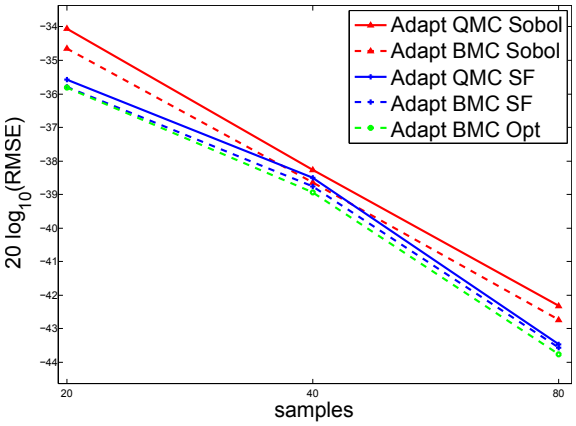


Figure 4.4 – RMSE for adaptive BMC and adaptive QMC using different sample sets.

6.4 A closer look at adaptive BMC

In this section we will address two questions which can arise when analyzing the results obtained using adaptive BMC. The first one is whether or not there is a loss of performance of adaptive BMC w.r.t. non-adaptive BMC when using exactly the same sampling positions (and thus the same information). The second is whether or not a different distribution of the samples to be taken between level one and level two would allow for better performances of BMC than those reported in Fig. 4.4.

The answers to these questions can be found in Fig. 4.5. The RMSE curve named 'Adapt BMC Sobol' is the same as in Fig. 4.4 and was taken as a reference. Recall that this curve was generated using always 10 samples for the first sampling level. The continuous red curve corresponds to the RMSE of NABMC using a Sobol sequence. The continuous blue curve gives the RMSE of ABMC using a Sobol sequence, where half of the samples are used for the first level and the other half for the second.

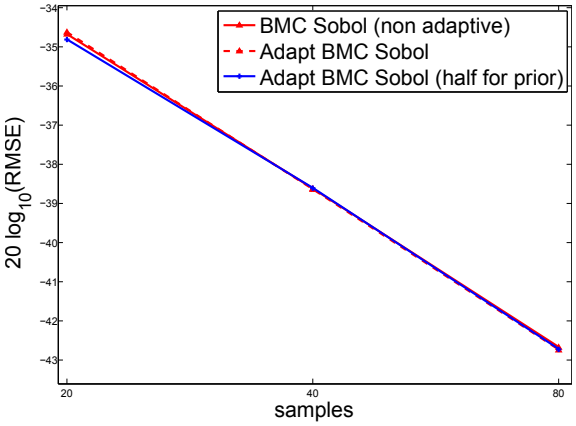


Figure 4.5 – RMSE for different settings of BMC.

The results show that there is no significant difference, in terms of RMSE, for each of the methods. Indeed, this result is expected since all three methods use exactly the same information. Nevertheless, there might be some room to improve the

RMSE curves of the adaptive sampling methods if the information produced by the first sampling level were used to determine the sampling positions of the second level. This would make the adaptive methods use more information than the NABMC and potentially lead to better results.

7 Conclusion

The application of BMC using an iterative approach, which allows refining previous estimates, is of particular interest since it exploits the prior knowledge from previous iterations. The main challenge of this approach is how to maintain the Bayesian quadrature computation tractable. More precisely, the calculation of the $\mathbf{z}_2$ vector of integrals becomes even more complex than in the single estimate case. In this chapter we have presented a solution for computing $\mathbf{z}_2$ which resorts to the same spherical Gaussian framework from the previous chapter to perform a fast quadrature computation. Furthermore, our solutions allow reusing a part of the computations of the first level estimate. We have also shown that the BMC adaptive approach can outperform the QMC adaptive approach in terms of RMSE.

Conclusion and Future Work 5

Contents

The spherical sampling of the incident radiance function entails a high computational cost. Therefore the illumination integral must be evaluated using a limited set of samples. In this thesis we have presented three different contributions which allow better estimates for the same number of samples. These contributions improve the state of the art solutions in computer graphics by taking into account information which had so far been underexploited (or even neglected) by the previous approaches. In the following, we summarize our three contributions and suggest ideas for future work.

1 Spherical Fibonacci point sets for illumination integrals

We have presented an algorithm for efficient generation of high quality spherical Quasi-Monte Carlo (QMC) sequences for approximating illumination integrals. This algorithm is simpler to implement than the other tested QMC spherical sample sets and yields higher quality results, allowing to save a significant amount of sample rays for the same image quality. Furthermore, it only requires a single sequence to synthesize an image, which makes is particularly well-suited to GPU implementations.

The main reason for the improvement brought by spherical Fibonacci point sets is that they better suit the spherical geometry. The other methods, in contrast, by focusing on the unit square distribution, neglect the fact that the discrepancy properties of the samples set on the unit square are unpaired by the spherical projection. The presented method, however, does not allow for incrementally adding new samples so as to implement an adaptive sampling strategy. Future work could thus tackle this problem.

2 A spherical Gaussian framework for Bayesian Monte Carlo rendering

This contribution represents a major step toward reaching a general Bayesian Monte Carlo (BMC) approach for solving global illumination problems. Until now, the application of BMC for rendering purposes had been limited to the diffuse BRDF case. We have proposed a general Bayesian Monte Carlo solution that allows dealing with non-diffuse BRDFs thanks to a spherical Gaussian-based framework. We have also proposed a fast hyperparameters determination method which avoids learning the hyperparameters for each BRDF.

The results confirm that Bayesian Monte Carlo outperforms state-of-the-art Quasi-Monte Carlo methods. In particular we achieve quality improvements over state of the art spherical QMC integration presented in this thesis (spherical Fibonacci) at comparable computational cost. Future work could aim at extending the use of BMC to other problems of larger dimensionality, e.g., measured BRDFs, multiple reflections, depth blur, multi-view rendering, among others.

3 Adaptive sampling with Bayesian Monte Carlo

Our third contribution consists in applying BMC with an adaptive approach for evaluating the illumination integral. The idea is to compute a first BMC estimate and, if the quality criterion is not met, directly inject the result as prior knowledge on a new estimate. The new estimate refines the previous estimate using a new set of samples, and the process is repeated until a satisfying result is achieved.

We have presented a solution which leverages the posterior Gaussian Process model of the integrand produced by the previous iteration. This solution benefits from the flexibility of the Bayesian approaches for introducing prior knowledge. Furthermore, we have shown that adaptive BMC can resort to the same spherical Gaussian framework presented in chapter 3. This is an important feature of our approach since it allows an efficient implementation of the Bayesian quadrature computation. Future work could be focused on adapting the sampling rate on a subregion of the domain of integration for which the incident radiance is sharper. Such an approach would be much simpler with BMC than with a traditional Monte Carlo technique, since BMC does not require the samples to be drawn according to a PDF.

Bibliography

[ABD12] C. Aistleitner, J.S. Brauchart, and J. Dick. Point sets on the sphere $\mathbb{S}^2$ with small spherical cap discrepancy. *Discrete & Computational Geometry*, pages 1–35, 2012. 6, 15, 20

[BBL+09] Jonathan Brouillat, Christian Bouville, Brad Loos, Charles Hansen, and Kadi Bouatouch. A Bayesian Monte Carlo approach to global illumination. *Computer Graphics Forum*, 28(8):2315–2329, 2009. 3, 30, 31, 32, 38, 39, 43

[BD11] J. S. Brauchart and J. Dick. Quasi-Monte Carlo rules for numerical integration over the unit sphere $\mathbb{S}^2$. *ArXiv e-prints*, January 2011. http://arxiv.org/abs/1101.5450. 6, 7, 8, 22

[Bec84] J. Beck. Sums of distances between points on a sphere — An application of the theory of irregularities of distribution to discrete geometry. *Mathematika*, 31(1):33–41, June 1984. 9

[Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. 37

[BSSW12] J. S. Brauchart, E. B. Saff, I. H. Sloan, and R. S. Womersley. QMC designs: optimal order Quasi Monte Carlo Integration schemes on the sphere. *ArXiv e-prints*, August 2012. http://arxiv.org/abs/1208.3267. 8

[Cae81] Terry Caelli. *Visual Perception, Theory and Practice*. Franklin Book Co (June 1981), 1981. 62

[CPC84] Robert L. Cook, Thomas Porter, and Loren Carpenter. Distributed ray tracing. In *Proceedings of the 11th annual conference on Computer graphics and interactive techniques*, SIGGRAPH '84, pages 137–145, New York, NY, USA, 1984. ACM. 2

[CYC+12] Zhonggui Chen, Zhan Yuan, Yi-King Choi, Ligang Liu, and Wenping Wang. Variational blue noise sampling. *IEEE Trans. Vis. Comput. Graph.*, 18(10):1784–1796, 2012. 10, 15

[dGBOD12] F. de Goes, K. Breeden, V. Ostromoukhov, and M. Desbrun. Blue noise through optimal transport. *ACM Trans. Graph. (SIGGRAPH Asia)*, 31, 2012. 5, 9, 10, 15, 19

[DHM+01] Philip Dutré, Paul Heckbert, Vincent Ma, Fabio Pellacini, Robert Porschka, Mahesh Ramasubramanian, Cyril Soler, and Greg Ward. Global illumination compendium. www.cs.kuleuven.ac.be/∼phil/GI/TotalCompendium.pdf, 2001. 12

[EPM+11] Mohamed S. Ebeida, Anjul Patney, Scott A. Mitchell, Andrew Davidson, Patrick M. Knupp, and John D. Owens. Efficient maximal Poisson-disk sampling. *ACM Transactions on Graphics*, 30(4):49, 2011. 10, 15

[Fat11] Raanan Fattal. Blue-noise point sampling using kernel density model. *ACM SIGGRAPH 2011 papers*, 28(3):48:1–10, 2011. 10, 15

[Gei07] Andreas Geiger. Gaussian process for machine learning, 2007. http://www.rainsoft.de/projects/gausspro.html. 33

[GHB+05] K.M. Gorski, Eric Hivon, A.J. Banday, B.D. Wandelt, F.K. Hansen, M. Reinecke, and M. Bartelmann. HEALPix - A framework for high resolution discretization, and fast analysis of data distributed on the sphere. *Astrophys. J.*, 622:759–771, 2005. ix, 12, 13

[GHSK06] Leonhard Grünschloß, Johannes Hanika, Ronnie Schwede, and Alexander Keller. (t, m, s)-nets and maximized minimum distance. In *Monte Carlo and Quasi-Monte Carlo Methods, Proceedings of the Seventh International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, 2006. 3

[GK08] Leonhard Grünschloß and Alexander Keller. (t, m, s)-nets and maximized minimum distance, part ii. In *Monte Carlo and Quasi-Monte Carlo Methods, Proceedings of the Eighth International Conference on Monte Carlo and Quasi-Monte Carlo Methods in Scientific Computing*, 2008. 11

[GKP94] Ronald L. Graham, Donald E. Knuth, and Oren Patashnik. *Concrete Mathematics: A Foundation for Computer Science*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2nd edition, 1994. 15

[Gon10] Álvaro González. Measurement of areas on a sphere using Fibonacci and Latitude–Longitude lattices. *Mathematical Geosciences*, 42:49–64, 2010. 15, 16

[HD12] Ferenc Huszár and David Duvenaud. Optimally-weighted herding is Bayesian quadrature. In *Uncertainty in Artificial Intelligence*, 2012. 66

[HJW+08] Toshiya Hachisuka, Wojciech Jarosz, Richard Peter Weistroffer, Kevin Dale, Greg Humphreys, Matthias Zwicker, and Henrik Wann Jensen. Multidimensional adaptive sampling and reconstruction for ray tracing. *ACM Trans. Graph.*, 27(3):33:1–33:10, August 2008. 62

[HN04] J H Hannay and J F Nye. Fibonacci numerical integration on a sphere. *Journal of Physics A: Mathematical and General*, 37(48):11591, 2004. 5, 9, 15, 16

[Jak10] Wenzel Jakob. Mitsuba renderer, 2010. http://www.mitsuba-renderer.org. 20, 50, 68

[Jen96] Henrik Wann Jensen. Global illumination using photon maps. In *Proceedings of the eurographics workshop on Rendering techniques '96*, pages 21–30, London, UK, 1996. Springer-Verlag. 2

[Kaj86] James T. Kajiya. The rendering equation. *SIGGRAPH Comput. Graph.*, 20(4):143–150, August 1986. 1, 2

[KC08] Jaroslav Křivánek and Mark Colbert. Real-time shading with filtered importance sampling. *Computer Graphics Forum*, 27(4):1147–1154, 2008. 31, 34

[Kel12] Alexander Keller. Quasi-Monte Carlo image synthesis in a nutshell, 2012. https://sites.google.com/site/qmcrendering/. 2, 6

[Ken13] Andrew Kensler. Correlated multi-jittered sampling. Technical report, Pixar, 2013. 9

[KGPB05] Jaroslav Křivánek, Pascal Gautron, Sumanta Pattanaik, and Kadi Bouatouch. Radiance caching for efficient global illumination computation. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):550–561, sept.-oct. 2005. 2

[KK02] Thomas Kollig and Alexander Keller. Efficient multidimensional sampling. *Comput. Graph. Forum*, 21(3):557–563, 2002. 3, 14, 19, 20, 21

[KN06] L. Kuipers and H. Niederreiter. *Uniform Distribution of Sequences*. Dover Books on Mathematics. Dover Publications, 2006. 8

[KPR12] Alexander Keller, Simon Premoze, and Matthias Raab. Advanced (quasi) Monte Carlo methods for image synthesis. In *ACM SIGGRAPH 2012 Courses*, SIGGRAPH '12, pages 21:1–46, New York, NY, USA, 2012. ACM. 14

[LD08] Ares Lagae and Philip Dutré. A comparison of methods for generating Poisson disk distributions. *Computer Graphics Forum*, 27(1):114–129, March 2008. 10, 14

[L'E09] Pierre L'Ecuyer. Quasi-monte carlo methods with applications in finance. *Finance and Stochastics*, 13(3):307–349, 2009. 7

[LP01] G. Larcher and F. Pillichshammer. Walsh series analysis of the L2-discrepancy of symmetrisized point sets. *Monatshefte für Mathematik*, 132(1):1–18, 2001. 5, 9, 10, 11, 14, 15, 20

[LPS86] A. Lubotzky, R. Phillips, and P. Sarnak. Hecke operators and distributing points on the sphere I. *Communications on Pure and Applied Mathematics*, 39(S1):S149–S186, 1986. 15

[LRU85] Mark E. Lee, Richard A. Redner, and Samuel P. Uselton. Statistically optimized sampling for distributed ray tracing. *SIGGRAPH Comput. Graph.*, 19(3):61–68, July 1985. 4, 62

[LW93] Eric P. Lafortune and Yves Willems. Bi-directional path tracing. In *Compugraphics '93*, pages 145–153, December 1993. 2

[LW95] Eric Lafortune and Yves Willems. A 5D tree to reduce the variance of Monte Carlo ray tracing. In *Rendering Techniques '95 (Proceedings of the Sixth Eurographics Workshop on Rendering)*, pages 11–20. Springer-Verlag, 1995. 53

[MFS04] Àlex Méndez-Feliu and Mateu Sbert. Comparing hemisphere sampling techniques for obscurance computation. In *3IA 04: Proceedings of the International Conference on Computer Graphics and Artificial Intelligence*, 2004. 6

[Mit87] Don P. Mitchell. Generating antialiased images at low sampling densities. *SIGGRAPH Comput. Graph.*, 21(4):65–72, August 1987. 4, 62

[Nie88] Harald Niederreiter. Low-discrepancy and low-dispersion sequences. *Journal of Number Theory*, 30(1):51 – 70, 1988. 7

[Nie92] Harald Niederreiter. *Random number generation and quasi-Monte Carlo methods*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992. 10, 11

[Nye03] J.F. Nye. A simple method of spherical near-field scanning to measure the far fields of antennas or passive scatterers. *Antennas and Propagation, IEEE Transactions on*, 51(8):2091 – 2098, aug. 2003. 15

[O'H91] A. O'Hagan. Bayes-Hermite quadrature. *J. Statist. Plann. Inference*, 29(3):245–260, 1991. 30, 31, 32

[Ost07] Victor Ostromoukhov. Sampling with polyominoes. *Proceedings of ACM SIGGRAPH 2007, ACM Transactions on Graphics*, 26(3), July 2007. 6

[PH10] Matt Pharr and Greg Humphreys. *Physically Based Rendering, Second Edition: From Theory To Implementation*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2nd edition, 2010. 7

[PHR06] T. Pfingsten, D. J. L. Herrmann, and C. E. Rasmussen. Model-based design analysis and yield optimization. *Semiconductor Manufacturing, IEEE Trans. on*, 19(4):475–486, 2006. 31

[Pur87] Werner Purgathofer. A statistical method for adaptive stochastic sampling. *Computers & Graphics*, 11(2):157 – 162, 1987. 62

[RFS02] Jaume Rigau, Miquel Feixas, and Mateu Sbert. New contrast measures for pixel supersampling. In *Proceedings of CGI 02*, pages 439–451. Springer-Verlag London Limited, 2002. 62

[RFS03] Jaume Rigau, Miquel Feixas, and Mateu Sbert. Entropy-based adaptive sampling. In *Graphics Interface*, pages 149–158, 2003. 62, 68

[RG02] Carl Edward Rasmussen and Zoubin Ghahramani. Bayesian Monte Carlo. In *Neural Information Processing Systems*, pages 489–496. MIT Press, 2002. 3, 31, 32

[RW06] C. E. Rasmussen and C. K. I. Williams. *Gaussian Process for Machine Learning*. MIT Press, 2006. 33, 34, 35, 42

[SC97] Peter Shirley and Kenneth Chiu. A low distortion map between disk and square. *J. Graph. Tools*, 2(3):45–52, December 1997. ix, 12, 13, 19, 20, 21

[SEB08] Peter Shirley, Dave Edwards, and Solomon Boulos. Monte Carlo and quasi-Monte Carlo methods for computer graphics. In Alexander Keller, Stefan Heinrich, and Harald Niederreiter, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 167–177. Springer Berlin Heidelberg, 2008. 6

[Sha48] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27, 1948. 62

[SK97] E. Saff and A. Kuijlaars. Distributing many points on a sphere. *The Mathematical Intelligencer*, 19:5–11, 1997. 9, 16, 43, 46

[Sob67] I. M. Sobol. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Math. and Math. Phys.*, 7:86–112, 1967. 5, 9, 11, 14, 15, 49

[SP06] Richard Swinbank and James Purser. Fibonacci grids: A novel approach to global modelling. *Quarterly Journal of the Royal Meteorological Society*, 132(619):1769–1793, 2006. 5, 15, 16

[Sve94] D. I. Svergun. Solution scattering from biopolymers: advanced contrast-variation data analysis. *Acta Crystallographica Section A*, 50(3):391–402, May 1994. 15

[Vea98] Eric Veach. *Robust Monte Carlo Methods for Light Transport Simulation*. PhD thesis, Stanford, CA, USA, 1998. AAI9837162. 2

[VG94] Eric Veach and Leonidas Guibas. Bidirectional estimators for light transport. In *Proceedings of Eurographics Rendering Workshop*, 1994. 2

[Vog79] Helmut Vogel. A better way to construct the sunflower head. *Mathematical Biosciences*, 44(3–4):179–189, 1979. 16

[WRC88] Gregory J. Ward, Francis M. Rubinstein, and Robert D. Clear. A ray tracing solution for diffuse interreflection. *SIGGRAPH Comput. Graph.*, 22(4):85–92, June 1988. 2

[WRG+09] Jiaping Wang, Peiran Ren, Minmin Gong, John Snyder, and Baining Guo. All-frequency rendering of dynamic, spatially-varying reflectance. *ACM Trans. Graph.*, 28:133:1–133:10, December 2009. 40

[XBZ+05] Qing Xu, Shiqiang Bao, Rui Zhang, Ruijuan Hu, and Mateu Sbert. Adaptive sampling for Monte Carlo global illumination using Tsallis entropy. In *Computational Intelligence and Security, International Conference, Proceedings, Part II*, volume 3802 of *Lecture Notes in Computer Science*, pages 989–994. Springer, 2005. 63

[XFSS07] Qing Xu, Miquel Feixas, Mateu Sbert, and Jizhou Sun. A new adaptive sampling technique for Monte Carlo global illumination. In *CAD/Graphics*, pages 191–196, 2007. 4, 63

[XSFS10] Qing Xu, M. Sbert, M. Feixas, and R. Scopigno. A new refinement criterion for adaptive sampling in path tracing. In *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, pages 1556 –1561, 07/2010 2010. 4, 63