MONTE CARLO SIMULATION OF INFINITE-DIMENSIONAL INTEGRALS

BY

BEN NIU

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Applied Mathematics
in the Graduate College of the
Illinois Institute of Technology

Approved _____
Advisor

Chicago, Illinois
May 2011

ACKNOWLEDGMENT

This dissertation could not have been written without Dr. Fred J. Hickernell who not only served as my supervisor but also encouraged and challenged me throughout my academic program. He and the other faculty members, Dr. Igor Cialenco, Dr. Greg Fasshauer, Dr. Dietmar Rempfer, guided me through the dissertation process, never accepting less than my best efforts. I appreciate the joint work with Dr. Müller-Gronbach and Dr. Klaus Ritter. I would also like to thank Mr. Samuel Eddins, my supervisor at IronBridge Capital Management, L.P., for his support. I thank them all.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

This thesis is motivated by pricing a path-dependent financial derivative, such as an Asian option, which requires the computation of the expectation of a payoff function, which depends on a Brownian motion. Employing a standard series expansion of the Brownian motion, the latter problem is equivalent to the computation of the expectation of a function of the corresponding i.i.d. sequence of random coefficients. This motivates the construction and the analysis of algorithms for numerical integration with respect to a product probability measure on the infinite-dimensional sequence. The class of integrands studied in this thesis resides in the unit ball in a reproducing kernel Hilbert space obtained by superposition of weighted tensor product spaces of functions of finitely many variables. Combining tractability results for high-dimensional integration with the multi-level technique we obtain new algorithms for infinite-dimensional integration. These deterministic multi-level algorithms use variable subspace sampling and they are superior to any deterministic algorithm based on fixed subspace sampling with respect to the respective worst case error. Numerical experiment results are presented at the end.

CHAPTER 1

INTRODUCTION

The original motivation of this thesis is to address the computational challenges arising from modern computational finance. Many typical problems in this area can be represented as problems of computing high-dimensional integration. In this thesis, we focus on the high-dimensional financial options pricing, especially options with path-dependent properties. High-dimensional integrals in hundreds or thousands of variables occur commonly in finance. In addition, it is common that such finance problems yield quite complicated integrands, which can not be evaluated analytically and precisely. A common approach is to use *numerical integration* for the approximation of integrals. These integrals have to be computed numerically to within a tolerance.

Numerical integration methods are generally described as combining evaluation of the integrand to get an approximation to the integral. Classical methods for multidimensional numerical integration are typically Cartesian products of one-dimensional rules such as the trapezoidal rule and Simpson's rule (see [DR84]). However, those frameworks only work well for dimensions up to 3 or 4. Another approach is to consider the multiple integrals as repeated one-dimensional integrals by appealing to Fubini's theorem. This approach requires an exponential increase in the function evaluations as the number of dimensions increase. This is the so-called *curse of dimensionality*, see [NW08, NW10, TWW88].

A solution to the high-dimensional integration problem is the Monte Carlo (MC) method, which is a class of computational algorithms relying on repeated random sampling. An enhanced approach is the quasi-Monte Carlo (QMC) method which is a deterministic version of a Monte Carlo method in the sense that low discrepancy sequences are used instead of random sampling. The low discrepancy sequences are well chosen deterministic points with good distributional properties. Please refer to Chapter 2 for an introduction to MC and QMC simulations and low discrepancy sequences.

This chapter is organized as follows. Section 1.1 provides the basic definitions, valuation methods, and applications of financial options. A literature review on the existing algorithms for evaluating high-dimensional integrals can be found in Section 1.2. Section 1.3 outlines the structure of the whole thesis.

1.1 Option Pricing

1.1.1 Definition. In finance, an option is a derivative financial instrument establishing a contract between two parties concerning the buying or selling of an asset at a reference price, which is called the strike price, during a specific time frame. During this time frame, the buyer of an option gains the right, but not the obligation, to engage in some specific transaction on the asset, which the seller incurs the obligation to fulfill if so requested by the buyer. For example, a call option on a specific stock conveys the right to the buyer to buy the stock at the pre-determined price in the future. Typically, if the price of the stock has surpassed the strike price, the buyer pays the strike price to actually purchase the stock and then sells the stock and pockets the profit. The buyer of the call option needs to pay a non-refundable premium for the legal right to exercise the call at the price. This premium is defined as the option price.

The mathematical theory of option pricing is both elegant and practical. The proper development of the theory and tools requires a book-length treatment; Therefore we highlight some classical principles of the theory, especially those highly related to the applicability of Monte Carlo to the calculation of prices.

- If a derivative security can be perfectly replicated through trading in other assets, then the price of the derivative security is the cost of the replicating trading strategy.
- Discounted asset prices are martingales under a probability measure associated with the choice of discount factor. Prices are expectation of the discounted payoffs under such a martingale measure.

- In a complete market, any payoff can be synthesized through a trading strategy, and the martingale measure associated with a discount factor is unique. In an incomplete market there are derivative securities that can not be perfectly hedged; the price of such a derivative is not completely determined by the prices of other assets.

1.1.2 Option pricing valuation. More specifically, the pricing of derivative securities can be summarized as follows. Consider a derivative security with a payoff at time T specified through a function g of the prices of the underlying assets, as in the case of a standard call or put option. To price the derivative, we model the dynamics of the underlying assets under the risk-neutral measure, ensuring that discounted asset prices are martingales. The price of the derivative is then given by $\mathbb{E}[\exp(-rT)g(S(\cdot))]$, where $g(\cdot)$ represents the payoff function and $S(\cdot)$ is the stock price dynamic.

The classical theory of option pricing is based on the pioneering work by Black and Scholes in 1973, [BS73]. To illustrate the model, consider the pricing of a call option on a stock. The underlying asset price is governed by a stochastic differential equation

$$\mathrm{d}S(t) = \mu S(t)\,\mathrm{d}t + \sigma S(t)\,\mathrm{d}B(t),$$

where $B(t)$ is a Brownian motion, the $\mathrm{d}B^0(t)$ term here stands in for any and all sources of uncertainty in the price history of the stock. The constant μ and σ stand for the expected return and volatility of the underlying stock. Under the risk-neutral measure, the stock price dynamics are given by

$$\mathrm{d}S(t) = rS(t)\,\mathrm{d}t + \sigma S(t)\,\mathrm{d}B(t), \tag{1.1}$$

where $B(t)$ is a standard Brownian motion under the risk-neutral measure, r is the risk-free rate. Solving this stochastic differential equation explicitly implies that

$$S(t) = S(0)\exp((r-\sigma^2/2)t + \sigma B(t)). \tag{1.2}$$

For a call option that has strike K, expiration T and a general payoff function g, its price at time 0 is given by

$$\text{option price} = \mathbb{E}[\text{discounted payoff}] = \mathbb{E}[\exp(-rT)g(S(\cdot)].$$

To evaluate this expectation, we simulate paths of the underlying assets over the time interval $[0,T]$, simulating according to the risk-neutral dynamics. On each path we calculate the discounted payoff $\exp(-rT)(S(\cdot))$; the average across paths is the estimate of the derivative price. For example, let us take a look at the arithmetic mean Asian call option. The payoff function depends on the underlying stochastic model for the asset price, $S(t)$, which is governed by a stochastic differential equation, for example, as in (1.1). Denote the option price as I, an arithmetic mean Asian call option is defined as:

$$\begin{aligned} \text{payoff}(B(\cdot)) &= e^{-rT}\max\left(\frac{1}{T}\int_0^T S(t)\,\mathrm{d}t - K, 0\right), \\ I &= \mathbb{E}[\text{payoff}(B(\cdot))], \quad \text{where } B(t) \text{ is a Brownian motion.} \end{aligned}$$

In order to approximate I, one may simulate n different Brownian motion paths, evaluate the payoff function, then take the sample average. The simulation of Brownian motion paths is discussed later in Chapter 2.

1.1.3 Styles of options. In finance, the vast majority of options are either European or American style options. These options are referred to as "vanilla options". The other styles of options are categorized as "exotic options". We focus on vanilla option and a special group of options from "exotic options", path-dependent options. Path-dependent options can pose challenging problems in valuation and hedging.

For vanilla options, the key difference between American and European options relates to when the options can be exercised. An European option may be exercised only at the expiration date of the option while the American option can be exercised at any time before the expiration date.

For exotic options, the option pricing formula is generally transferred to an infinite-dimensional integration problem, which is the main topic of this thesis. Different from vanilla options, exotic options' payoffs depend on the path of the underlying asset. More specifically, the payoff at maturity depends not just on the value of the underlying asset at maturity, but as its value at several times during the contract's life. For example, an Asian option depends on the sample average of the underlying asset over the life of the option, a lookback option depends on the maximum or minimum of the underlying asset over the life of the contract, a barrier option pays nothing if a certain level is not reached by the underlying. More details of those options are introduced in Chapter 6.

1.2 Existing Algorithms for Evaluating High Dimensional Integrals

Many papers over the past decade have considered the approximation of the integral of a function f over the $d-$dimensional unit cube,

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

For the classical problem with a large d, the integral is usually approximated by the algorithms

$$\hat{I}_n(f) = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i),$$

where the point set $\mathscr{P} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n\}$ are well chosen from the unit cube $[0,1]^d$.

If $\mathscr{P}$ is simple random sequence, it corresponds to the MC algorithm. By the law of large numbers, this method achieves a $1/\sqrt{n}$ convergence rate, which means the error of the estimator is independent of the dimension. If $\mathscr{P}$ is the low discrepancy sequence, the method is called the quasi-Monte Carlo method which is usually considered to be more accurate than Monte Carlo method, but for the MC method, the error measure is much easier. The error bound of QMC method is $\mathscr{O}(n^{-1+\varepsilon})$, which is almost twice the convergence order of the simple Monte Carlo method. See [Hic98, Nie92] for details of the error analysis. However, [CM96, CMO97, Mor98] remarked that the advantage of QMC method is

great only if the integrand is smooth and the number of dimensions of the integral is small. In [CM96], the valuation of a mortgage-backed security with dimensions as high as 360 was presented to address this issue. Note that conventionally QMC methods are totally deterministic and thus yield no statistical error estimation as in the MC methods. However, an extended technique, so called the randomized quasi-Monte Carlo which mixes the MC and QMC, extends the benefits of MC to QMC techniques. The randomized QMC is conducted by randomizing the deterministic integration points used in the QMC method. This allows the combination of the faster convergence rate of QMC and the statistical error estimation of MC. See [LL02] for a detailed discussion of the recent advances randomized QMC methods.

There are many results dealing with the error analysis of the high-dimensional integrations, where the number of dimensions is arbitrary but finite. See [HW00, SW98, SW01, Woź00, Wan03] and the books [NW08, NW10] for the tractability results for multivariate integration problems. There have also been investigations of the error estimation for the randomized QMC method. See [Owe94, Owe97b, Owe97a, Owe98, Yue99] for the study of variance. See [YH01, YH05, HY00, HH99] for the study of root mean square discrepancies of the scrambled nets and sequences. The error analysis of scrambled QMC quadrature rules in the worst-case, random-case and average-case settings can be found in [HHY01].

In recent years, integration problems with infinitely many variables become more and more popular. The general problem is to study numerical integration with respect to probability measures μ on infinite-dimensional spaces $\mathscr{X}$,

$$I(f) = \int_{\mathscr{X}} f(\mathbf{x})\,\mu(d\mathbf{x}), \qquad f \in F,$$

where $\mathscr{X} \subset D^{\mathbb{N}}$ to be defined in (3.5), and F is some prescribed class of functions. The error bounds and tractability of dimensionally unbounded integration problems within the framework of weighted reproducing kernel Hilbert spaces were studied in [HW02]. The

theory of reproducing kernels is used to derive a quadrature error bound. For the infinite-dimensional quadrature problems, the cost of function evaluation is assumed to be dimension dependent and both deterministic and randomized algorithms have been studied. See [CDMGR09] for integration on separable Banach spaces and [Gne10, HMGNR10, NHMGR10, NH10, KSWW10] for integration on the sequence space $\mathbb{R}^{\mathbb{N}}$. For the study of tractability we refer in particular to the monograph series [NW08, NW10].

Recently, multi-level algorithms have been employed for infinite-dimensional integration. In [CDMGR09, Gne10, HMGNR10, KSWW10, NHMGR10], the multi-level methodology, which was introduced by [Hei98] in the context of integral equations and [Gil, Gil08, GW09, Hei98] in the context of stochastic differential equations, plays a key role. In contrast to the common single-level approach, a multi-level algorithm evaluates f at points from a hierarchy of finite-dimensional subspaces. This type of sampling has turned out to be superior to fixed subspace sampling for a number integration problems. Here superiority refers to a comparison of specific algorithms based on numerical experiments or upper bounds for their error and cost, or a comparison based on the analysis of minimal errors, i.e., on the study of upper and lower bounds. See [Gne10] for detailed discussions on finite order weights, where improved lower bounds for the exponent of tractability for general finite-order weights and improved upper bounds for three newly defined classes of finite-order weights are presented. In [Gne10], Gnewuch also extends the worst case error analysis to the case of arbitrary weighs, which concludes a lower bound on the exponent of tractability for arbitrary weights and a constructive upper bound for product weights.

1.3 Outline of the Thesis

In Chapter 2, we propose several methodologies for the simulation of Brownian motions, which are essential in the option pricing application, including the *Time Discrete Method*, *Karhunen-Loève Expansion Method* and the *Brownian Bridge Method*. The mechanics of each method are discussed. Then, we talk about *Low Discrepancy Sampling*

which features the quasi-Monte Carlo simulation. The construction and property of several important low discrepancy sequences are presented and the general error bound for the low discrepancy sampling is also discussed. This chapter ends with introductions of the single-level algorithm and multi-level algorithm. A toy example approximated by the single-level algorithm is also shown.

Chapter 3 focuses on the classes of integrands studied. The basic idea is to consider the infinite-dimensional integration as the limiting case of the finite-dimensional integration. First, we introduce the reproducing kernel Hilbert space which serves as a fundamental tool to facilitate the error analysis later. This chapter then proposes the basic assumption on the univariate probability measure that defines μ above. The reproducing kernel K_1 for the univariate function and the decaying coordinate weights γ_j associated with the corresponding reproducing kernel Hilbert spaces are defined. The function space for functions of finitely many variables is constructed in the setting of reproducing kernel Hilbert spaces, followed by the function space for functions of countably infinite many variables. We then describe the integration within those function classes. This chapter ends with the application of the uniform distribution on different domains.

In Chapter 4 and Chapter 5, we provide the error bound either in the worst-case setting or the randomized setting. We first introduce the *fixed subspace* and *variable subspace* sampling cost models and minimal errors. For the worst-case setting, the error bound of the single-level algorithm is derived, followed by the upper error bound of the multi-level algorithms. The lower bounds in both cost models are also briefly discussed later in this chapter. An application of those results for the unform distribution on $[0,1]$ and $K_1 : (x,t) \mapsto \min(x,t)$ is shown. In the randomized setting, the lower and upper bounds for the single-level algorithm are presented. Finally, the upper bound for the multi-level algorithm is studied.

Chapter 6 talks about the implementation of the single-level and multi-level algo-

rithm. A dynamic algorithm is introduced to optimally search for the required amount of sample paths at each level and the levels for the multi-level algorithm. Root mean square errors are estimated to measure the approximation error. Two types of financial options, the geometric mean Asian option and Barrier option are priced based on the algorithm we introduce.

Chapter 7 is a summary of independent work which investigates the error estimator for quasi-Monte Carlo algorithms. We present some preliminary results here.

CHAPTER 2

ALGORITHMS AND ILLUSTRATIVE EXAMPLES

2.1 Simulating Random Variables and Stochastic Processes

This section discusses the mechanics of stochastic processes simulation. We focus on the Brownian motion since the most popular model used for the option pricing is governed by the Brownian motion, which is a fundamental tool for modeling random phenomenon and for stochastic calculus in general. Following [KS91, Shr04], we define

Definition 2.1 *A standard Brownian motion is a stochastic process* $B : [0, +\infty) \to \mathbb{R}$ *such that*

(1) $B(\cdot)$ *is a continuous function on* $[0, +\infty)$.

(2) $B(0) = 0$.

(3) For all $0 \leq \tau \leq t$, $B(t) - B(\tau)$ *is normally distributed with zero mean and variance* $(t - \tau)$. *In addition,* $B(t) - B(\tau)$ *is independent of* $B(t') - B(\tau')$ *given that* $0 \leq \tau' < t' < \tau < t$.

We admit that there could be other choices, e.g., the poisson process used to model the jump diffusion model. One of the drawbacks for Brownian motion is that the continuous property may miss the discontinuous behavior which is common in finance.

One important task is to simulate the Brownian motion path accurately and efficiently. For computational purposes it is useful to consider the discretized Brownian motion where $B(t)$ is specified at discrete t values. As follows, we introduce two methods approximating Brownian motion paths. One is the time-discrete method based on the the independent increment property of the Brownian motion. Another method is to expand the Brownian motion as an infinite random series, which fits the algorithm introduced later.

2.1.1 The time-discrete method. As in [Hig01], we set $\delta = T/m$ for some positive integer m over the time interval $[0,T]$, and $t_j = j\delta$ for $j = 1,2,\ldots$. It can be implied from Definition 2.1 that $B(0) = 0$ and

$$B(t_j) = B(t_{j-1}) + \sqrt{\delta}X_j, \quad j = 1,2,\ldots,m,$$

where $X_1,\ldots,X_m$ are independent standard normal random variables, i.e., $N(0,1)$. The recursive relationship enables us to approximate the Brownian path at each specified time point $t_j, j = 1,2,\ldots,m$ as

$$B(t_k;X_1,X_2,\ldots) = \sqrt{\delta}(X_1 + X_2 + \cdots + X_k), \tag{2.1}$$

where $X_1, X_2, \ldots$ are i.i.d. $N(0,1)$.

2.1.2 The expansion method. In the theory of stochastic processes, a stochastic process can be represented as an infinite linear combination of orthogonal functions. In this section, we introduce an alternative approach to simulate the Brownian path with the expansion methodology. Please refer to [Mck69, Chapter. 1] for more detailed analysis.

Recall that the covariance matrix of the random vector $\mathbf{X} = \big(B(t_1),\cdots,B(t_m)\big)$ is,

$$Q = \begin{pmatrix} t_1 & t_1 & \cdots & t_1 \\ t_1 & t_2 & \cdots & t_2 \\ \vdots & \vdots & \vdots & \vdots \\ t_1 & t_2 & \cdots & t_n \end{pmatrix}.$$

Thus the characteristic function of Q is

$$\psi_{\mathbf{X}}(u) = \mathbb{E}[e^{iu^T\mathbf{X}}] = e^{-\frac{1}{2}u^TQu}, \tag{2.2}$$

with respect to the normal distribution with mean zero and covariance matrix Q. It can be easily checked that (2.2) satisfies the Kolmogorov consistency conditions, so that the process with finite dimensional distributions given by (2.2) exists on some probability space.

However, we want to construct the process in such a way that the sample functions are continuous, which does not follow from the consistency theorem.

We know that any function $f \in L_2[0,1]$ can be expanded as a Fourier series

$$f(t) = a_0 + \sum_{i=1}^{\infty} a_i \sin 2\pi it + \sum_{i=1}^{\infty} b_i \cos 2\pi it.$$

In fact, the trigonometric functions $1, \sqrt{2}\sin 2\pi it, \sqrt{2}\cos 2\pi it, i = 1,2,\cdots$ provide an orthonormal basis of $\mathscr{H} = L_2[0,1]$. For $f, g \in \mathscr{H}$, the norm and inner product are defined by

$$\|f\| = \sqrt{\int_0^1 f^2(t)\,\mathrm{d}t}, \quad \langle f, g\rangle = \int_0^1 f(t)g(t)\,\mathrm{d}t.$$

Suppose $\Phi = \{\phi_1, \phi_2, \cdots\}$ is a countable set of orthonormal functions. Let $\mathscr{H}_n = \{\sum_{i=1}^n \alpha_i\phi_i : \alpha \in \mathscr{R}^n\}$ be the linear subspace spanned by $\{\phi_1, \cdots \phi_n\}$. We define the projection onto $\mathscr{H}_n$ of an arbitrary $f \in \mathscr{H}$ as

$$\hat{f}_n = \sum_{i=1}^{n} \langle f, \phi_i\rangle \phi_i,$$

and Φ is complete if, for any $f \in \mathscr{H}$, $\hat{f}_n \to f$ as $n \to \infty$. In this case we say Φ is a complete orthonormal basis (CONB). For a CONB, the Parseval equality states that

$$\|f\|^2 = \sum_{i=1}^{\infty} \langle f, \phi_i^2\rangle.$$

Let $\{\phi_i\}$ be an arbitrary CONB of $\mathscr{H}$ and $X_1, X_2, \cdots$ be a sequence of i.i.d. random variables defined on a probability space $(\Omega, \mathscr{F}, P)$, with $X_i \sim N(0,1)$. For $n = 1,2,\cdots$, define

$$B^n(t) = \sum_{i=1}^{n} X_i \int_0^t \phi_i(u)\,\mathrm{d}u. \tag{2.3}$$

Theorem 2.1 *For each t, $B^n(t)$ is a Cauchy sequence in $L_2(\Omega, \mathscr{F}, P)$ whose limit $B(t)$ is a normal random variable with mean zero and variance t. For any two time t, τ, $\mathbb{E}[B(t)B(\tau)] = t \wedge \tau$, where $t \wedge \tau = \min(t, \tau)$.*

Proof 2.1 *Define the Heaviside function*

$$I_t(u) = \begin{cases} 1, & u < t \\ 0, & u \geq t \end{cases}$$

Then, $\int_0^t \phi_i(u)\,\mathrm{d}u = \langle I_t, \phi_i \rangle$. *Since* $\{\phi_i\}$ *is a CONB,*

$$I_t = \sum_{i=1}^{\infty} \langle I_t, \phi_i \rangle \phi_i, \quad \textit{and } \|I_t\|^2 = \sum_{i=1}^{\infty} \langle I_t, \phi_i \rangle^2.$$

Thus for $n > m$

$$\begin{aligned} \mathbb{E}\left((B^n(t) - B^m(t)\right)^2 &= \mathbb{E}\left(\sum_{i=m+1}^{n} X_i \int_0^t \phi_i(u)\,\mathrm{d}u\right)^2 \\ &= \sum_{i=m+1}^{n} \langle I_t, \phi_i \rangle^2 \to 0, \quad as \quad m,n \to \infty. \end{aligned}$$

Thus $B^n(t)$ *is a Cauchy sequence in* $L_2(\Omega, \mathscr{F}, P)$. *Denoting* $B(t)$ *as the limit of* $B^n(t)$, *then* $\mathrm{var}(B(t)) = \lim_{n\to\infty} \mathrm{var}(B^n(t)) = t$. *It follows that*

$$\mathbb{E}[B(t)B(\tau)] = \sum_{i=1}^{\infty} \langle I_t, \phi_i \rangle \cdot \langle I_\tau, \phi_i \rangle = \langle I_t, I_\tau \rangle = t \wedge \tau.$$

It remains to show that $B(t)$ *is normal. Note that* $B^n(t)$ *is a finite sum of normal random variables and is therefore normal, with variance* $\sigma_n^2 = \sum_{i=1}^n \langle I_t, \phi_i \rangle^2$. *Hence the characteristic function of* $B^n(t)$ *is* $\chi_n(u) = \mathbb{E}[e^{iuB^n(t)}] = e^{-\sigma_n^2 u^2/2}$, *which converges to* $\chi(u) = e^{-tu^2/2}$ *as* $n \to \infty$. *Now* $B^n(t) \to B(t)$ *in* L_2 *implies that there is a sub-sequence* $B^{n_k}(t)$ *such that* $B^{n_k}(t) \to B(t)$ *a.s. as* $k \to \infty$. *It follows from the bounded convergence theorem that* $\mathbb{E}[e^{iuB^{n_k}(t)}] \to \mathbb{E}[e^{iuB(t)}]$ *and hence* $\mathbb{E}[e^{iuB(t)}] = \chi(u)$. *Therefore* $B(t) \sim N(0,t)$.

Theorem 2.1 demonstrates that the Brownian motion 'exists' in the sense of the appropriate covariance function, but the continuous property of the Brownian motion has not yet be proved. The following shows that $B(t) = \lim_{n\to\infty} B^n(t)$ is a continuous stochastic process. The Haar functions $f_0, f_{i,j}, i, j = 1, 2, \ldots$ are introduced to facilitate the proof.

They are defined by $f_0(t) = 1$, with $k = 2j-1$,

$$f_{i,j} = \begin{cases} 2^{(i-1)/2}, & \frac{k-1}{2^i} \le t \le \frac{k}{2^i}, \\ -2^{(i-1)/2}, & \frac{k}{2^i} \le t \le \frac{k+1}{2^i}, \\ 0, & \text{elsewhere.} \end{cases}$$

Theorem 2.2 *The Haar functions are a complete orthonormal basis in* $L_2[0,1]$.

Lemma 2.1 *Suppose that, for* $n = 1,2,\cdots, f_n : [0,1] \to \mathscr{R}$ *is a continuous function, and that* f_n *converges uniformly to a function* f*, i.e., given* $\varepsilon > 0$ *there is a number* N *such that* $n \ge N$ *implies* $|f_n(t) - f(t)| < \varepsilon$ *for any* $t \in [0,1]$*. Then* f *is a continuous function.*

The idea now is to show that $B^n(t) \to B(t)$ uniformly almost surely when we take the CONB $\{\phi_i\}$ to be the Haar functions. Then, consider the indefinite integrals of the Haar function, called *Schauder functions* $F_0, F_{i,j}$, i.e. $F_0(t) = t$, and

$$F_{i,j}(t) = \begin{cases} 2^{(i-1)/2}(t - (k-1)/2^i), & \frac{k-1}{2^i} \le t \le \frac{k}{2^i}, \\ 2^{(i-1)/2}((k+1)/2^i - t), & \frac{k}{2^i} \le t \le \frac{k+1}{2^i}, \\ 0, & \text{elsewhere.} \end{cases}$$

Imagine Schauder functions as 'little tents' of height $2^{-(i+1)/2}$. Suppose $X_0, X_{i,j}, i, j = 1,2,\ldots$ are independent $N(0,1)$ random variables in the probability space $(\Omega, \mathscr{F}, P)$, and for $t \in [0,1], i = 1,2,\ldots$, define

$$B^n(t) = X_0 F_0(t) + \sum_{i=1}^{n} Y_i(t), \tag{2.4}$$

where

$$Y_i(t) = \sum_{j=1}^{2^{i-1}} X_{i,j}(\omega) F_{i,j}(t).$$

For each n, the sample function $t \mapsto B^n(t)(t)$ is a continuous function. It remains to show that $B^n(t) \to B(t)$ as $n \to \infty$.

Theorem 2.3 *The sequence $B^n(t)$ defined by* (2.4) *converges uniformly in t. Thus the process $B(t) = \lim_{n\to\infty} B^n(t)$ is a stochastic process with continuous sample paths.*

Proof 2.2 *The proof is an application of the Borel-Cantelli Lemma. Define*

$$H_i = \max_{t\in[0,1]} |Y_i(t)|.$$

Since for fixed i, the Schauder functions $F_{i,1}, F_{i,2}, \ldots$ are non-zero on disjoint intervals, then

$$H_i = 2^{-(i+1)/2} \max_{1\le j\le 2^{i-1}} |X_{i,j}|.$$

Thus for any constant c_i

$$\begin{aligned} P\left[H_i > 2^{-(i+1)/2} c_i\right] &= P\left[\max_j |X_{i,j} > c_i|\right] \\ &= P\bigcup_j \left[|X_{i,j}| > c_i\right] \le \sum_j P\left[|X_{i,j}| > c_i\right] \\ &\le 2^{i-1} \frac{2}{c_i\sqrt{2\pi}} e^{-\frac{1}{2}c_i^2}. \end{aligned} \tag{2.5}$$

Then, choose $c_i = \theta\sqrt{2i\log(2)}$ for some $\theta > 1$. Then the right hand side of (2.5) *is const $\times\, 2^{(1-\theta^2)i}\frac{1}{\sqrt{i}}$, which is convergent. In addition, $b_i := 2^{-(i+1)/2} c_i = \theta\sqrt{i2^{-i}\log 2}$ is also a convergent sequence. From* (2.5) *and the Borel-Cantelli Lemma,*

$$P[H_i > b_i \quad \textit{infinitely often}] = 0,$$

i.e., there exists N such that

$$H_i \le b_i, \quad \textit{for } i \ge N.$$

This shows that H_i is convergent and completes the proof.

According to Theorem 2.3, the Brownian motion expanded as an infinite random series in form of (2.3) satisfies all the properties in Definition 2.1. For a Brownian motion defined on $[0,T]$, we represent $B(t)$ as

$$B(t) = \sum_{j=1}^{\infty} X_j \cdot e_j(t), \qquad X_1, X_2, \ldots \text{ i.i.d. } N(0,1). \tag{2.6}$$

Here $e_j(\cdot), j = 1,2,\ldots$ are orthonormal basis of $\mathscr{H} = L_2(0,T)$. Let us consider the covariance kernel of $B(t)$. Given any $t, \tau \in [0,T]$,

$$\text{cov}(B(t),B(\tau)) = \mathbb{E}[B(t)B(\tau)] = \sum_{j=1}^{\infty} e_j(t)e_j(\tau) = \min(t,\tau).$$

This is due to the orthonormal property of the function $e_j(t)$. There might be various choices for $e_j(t)$ because any orthonormal basis in $L_2[0,T]$ guarantee the existence of the expansion (2.6). The following provides two examples of the expansion.

2.1.2.1 The Karhunen-Loève expansion. The Karhunen-Loève expansion is simply a special case of the expansion (2.6). As in [Adl90], solving the eigenvalue problem of the covariance function of the Brownian motion $B(t)$ yields a particular class of basis. Those basis are exclusive for the standard Brownion motion because the integral equation defining the eigenvalue problem is easily solved.

The Karhunen-Loève expansion for $B(t)$ defined on $[0,1]$ is

$$B(t) = \sum_{j=0}^{\infty} X_j \frac{2}{(2j+1)\pi} \sin \frac{t}{\sqrt{\lambda_j}}.$$

More generally, the Brownian motion $B(t)$ defined on $[0,T]$ can be expanded as

$$B(t,\mathbf{X}) = \sum_{j=1}^{\infty} X_j \frac{\sqrt{2T}}{\pi(j-1/2)} \sin\left(\pi\left(j-\frac{1}{2}\right)\frac{t}{T}\right), \tag{2.7}$$

where $X_1, X_2, \ldots$ are i.i.d. $N(0,1)$. In practical computation, the Karhunen-Loève expansion is truncated at a finite dimension, d. Given a vector $\mathbf{x} = (x_1, x_2, \ldots, x_d)$, a realization of $B(t)$ is

$$B(t) \approx \hat{B}_d(t;x_1,\ldots,x_d) = \sqrt{2T} \sum_{j=1}^{d} x_j \frac{\sin\left(\left(j-\frac{1}{2}\right)\pi t/T\right)}{\left(j-\frac{1}{2}\right)\pi}.$$

Note that the truncated Karhunen-Loève expansion corresponds formally setting, $X_{d+1} = X_{d+2} = \cdots = 0$. Below we want to construct a sequence of approximate realizations of the Brownian motion. This can be done as $\hat{B}_d(t;\mathbf{x}_i), i = 1,2,\ldots$, where $\mathbf{x}_i$ might come from

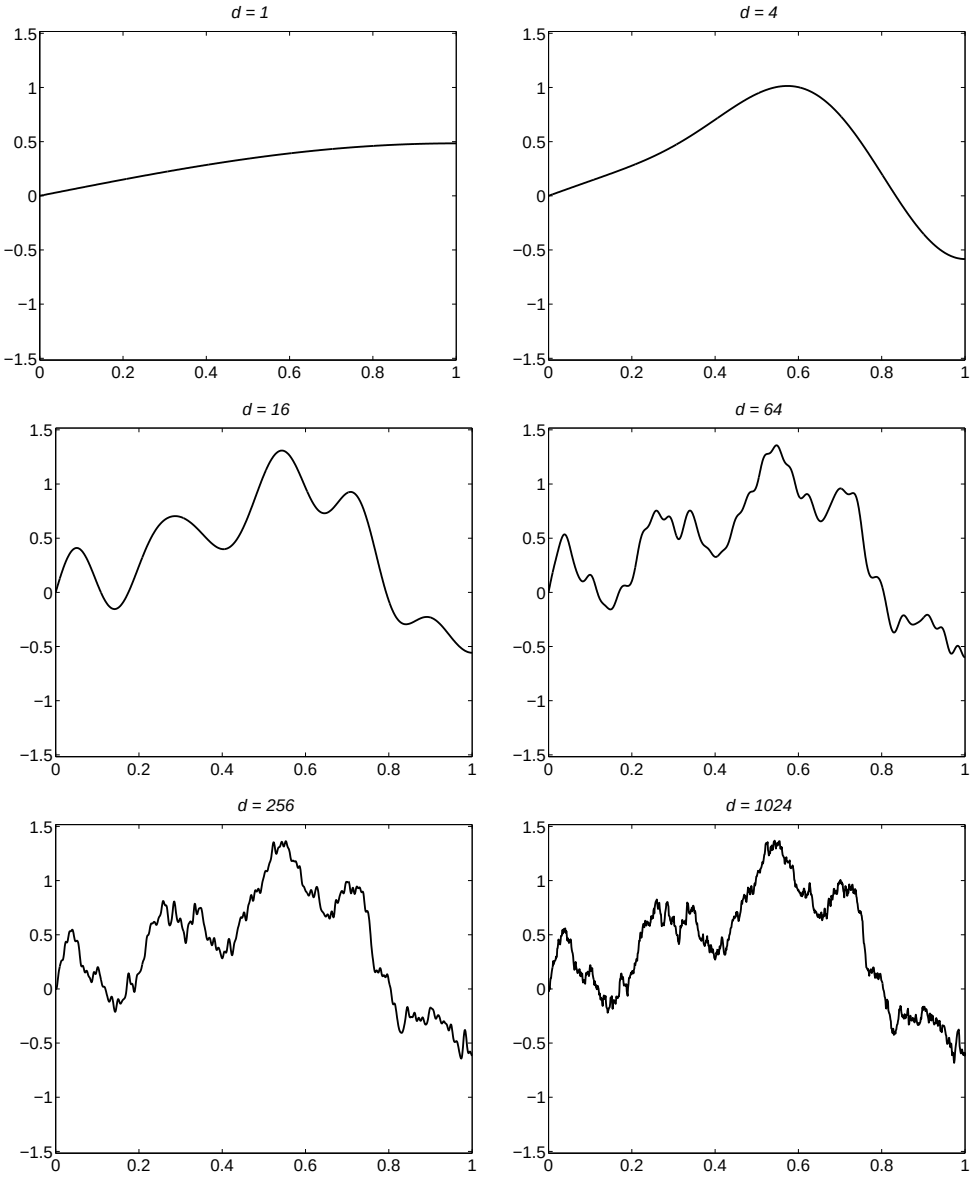


Figure 2.1. Simulation of Brownian motion $B(t)$ using Karhunen-Loève expansion by taking $d = 1, 4, 16, 64, 256, 1024$.

the simple random sequence, grid, centroidal Voronoi tessellation, Latin hypercube, or low discrepancy sequences to be discussed later.

Figure 2.1 demonstrates the simulation of a Brownian motion $B(t)$ over the interval $[0,1]$ taking various truncated dimension $d = 1,4,16,64,256,1024$. As seen from (2.7), as j increases, the magnitude of the expansion series diminishes while the frequency increases because of the sine function. Those plots show that the most variance of $B(t)$ is captured by the first few dimensions, i.e., small values of d yield a good approximation of $B(t)$. As d increases, the basic shape of $B(t)$ is maintained while more and more details are added. This implies that we may not need to select a very large d to truncate the Karhunen-Loève expansion series, only a reasonably small d yields accurate results.

2.1.2.2 The Brownian bridge. A Brownian bridge process $B(t)$ defined on $[0,T]$ is a continuous stochastic process with the following properties:

(i) $\mathbb{E}[B(t)] = 0, \quad 0 \le t \le T.$

(ii) $\mathbb{E}[B(t)B(\tau)] = \tau \wedge t - \tau t, \quad 0 \le \tau, \quad t \le T.$

In practice, the Brownian bridge is constructed by extending the time-discrete approach with continued refinements of the time mesh. For example, suppose a Brownian motion defined on the time interval $[0,1]$. We first sample $B(1)$, and $B(0) = 0$ by the definition, the Brownian bridge is constructed by filling in additional points in $[0,1]$, that is to interpolate between the already generated points $B(0)$ and $B(1)$, $B(1/2)$ is generated. We then continue to generate $B(1/4)$ and $B(3/4)$ by interpolation given the already generated points $B(0), B(1/2), B(1)$. This process continues by filling in the midpoint of each subintervals.

More generally, for any $k \in \mathbb{N}_0$, let $t_{k+1} = \big(1 - \psi_2(k)\big)\, T$, where ψ_2 is the radical-inverse function in base 2 defined in (2.12), which maps each k to a point in $[0,1)$ by reflecting digits of the binary expansion for k about the decimal points. The Brownian

bridge is mathematically defined as

$$\begin{aligned} B(t_0) &= B(0) = 0, \\ B(t_1;X_1) &= B(T) = \sqrt{T}X_1, \\ B(t_{k+1};X_1,\ldots,X_{k+1}) &= \frac{1}{2}\left[B\left(t_{k+1}-\frac{T}{2^{m_k+1}}\right)+B\left(t_{k+1}+\frac{T}{2^{m_k+1}}\right)\right] \\ &\quad +\sqrt{\frac{T}{2^{m_k+2}}}X_{k+1}, \quad k=1,2,\ldots,d, \end{aligned}$$

where $X_1, X_2, \ldots$ are $N(0,1)$. The key point here is to expand the Brownian motion $B(t)$ by (2.6) using the above sampling regime. Taking Schauder functions (with a more convenient index set) as the basis, we have $e_{1,0}(t) = t$ and

$$e_{k,m}(t) = \int_0^t 2^{(m-1)/2} \cdot \left(1_{[(k-1)/2^m,k/2^m]} - 1_{[k/2^m,(k+1)/2^m]}\right)(u)\,\mathrm{d}u \tag{2.8}$$

for $k \in \{2j-1 : j = 1,\ldots,2^{m-1}\}$ and $m \in \mathbb{N}$. Truncating the Brownian bridge at d steps corresponds to setting the variables $X_{d+1}, X_{d+2}, \ldots$ to the nominal value, 0, and then $B(t)$ becomes simply the linear interpolant between the two neighboring time mesh points t_{k_1} and t_{k_2}, where $0 \le k_1, k_2 \le d$. The cost of a d-step Brownian bridge is $\mathscr{O}(d)$ operations. Thus, the Brownian bridge generation of the Brownian motion fits the problem formulation previously.

Figure 2.2 shows the same phenomena as in Figure 2.1, the majority various parts of the Brownian motion $B(t)$ are approximated by taking a small d, and taking a large d means adding more fine details to the approximation.

Remark 2.1 *We introduced three different ways to simulate the Brownian motion at a specific time node t. Among those methods, the discrete time method is the easiest to implement, whereas the drawbacks are substantial. At specified time points,* (2.1) *yields a accurate approximation of the Brownian motion, while the discretization error is non-negligible. For example, in order to simulate the Brownian motion at time t, we need to interpolate*

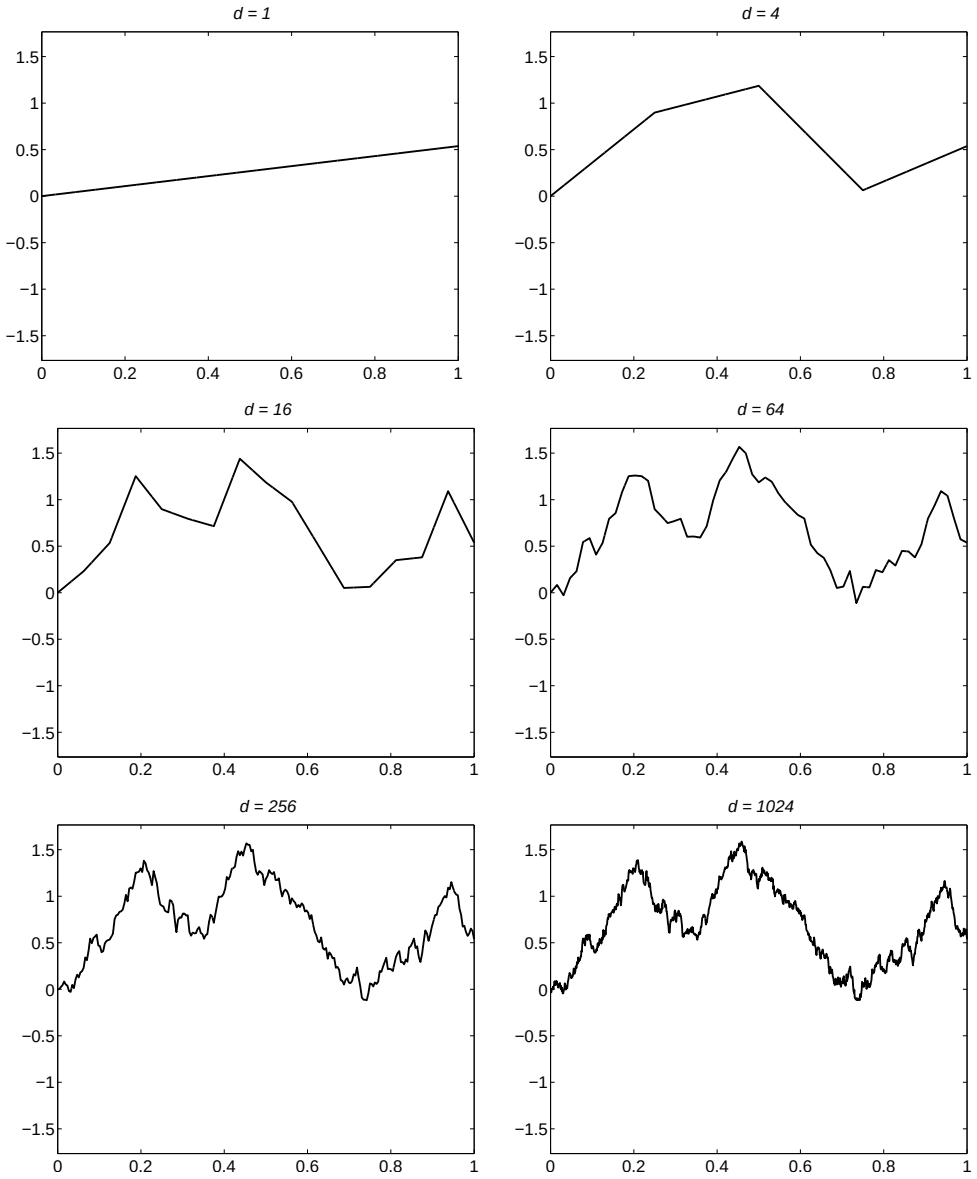


Figure 2.2. Simulation of Brownian motion $B(t)$ using Brownian Bridge method by taking $d = 1, 4, 16, 64, 256, 1024$.

based on two adjacent time points, i.e., $B(t) \approx \frac{B(t_k)(t_{k+1}-t_k)}{t_{k+1}-t_k} + \frac{B(t_{k+1}(t-t_k)}{t_{k+1}-t_k}$ *for* $t_k \le t \le t_{k+1}$. *In addition, the computational cost is also a burden if the discretization dimension m has to be chosen large to meet the accuracy tolerance. While for the Karhunen-Loève expansion and Brownian bridge method, only a few terms are needed to approximate the Brownian motion at a specific time t by truncating the infinite random series.*

Let us explain how the simulated Brownian motion fits into the framework of option pricing. The Brownian motion $B(\cdot)$ drives a SDE that determines the asset price $S(\cdot) = \Gamma(B(\cdot))$ for some function Γ. An example is introduced in (1.2) where

$$\Gamma(u)(t) = S(0)\exp((r-\sigma^2/2)t + \sigma u(t)).$$

The payoff function of a path-dependent option is given by $\varphi(S(\cdot))$. In the case of an arithmetic mean Asian call option,

$$\varphi(v) = e^{-rT}\max\left(T^{-1}\int_0^T v(t)\,\mathrm{d}t - K, 0\right).$$

Then the fair price of the option is $\mathbb{E}(\varphi(\Gamma(B(\cdot))))$.

The Brownian motion is approximated by taking the series expansion (2.6) with an i.i.d. standard normal random variables $\{X_j\}$ and a sequence of functions $e_j(t) \in C([0,T])$. Possible choices of basis functions e_j are provided by either the Karhunen-Loève expansion or the Brownian bridge construction method.

We further denote the fair price of the option as

$$\mathbb{E}(\varphi(S)) = \mathbb{E}(f)$$

with $f : \mathbb{R}^{\mathbb{N}} \to \mathbb{R}$ given by

$$f(\mathbf{X}) = \varphi \circ \Gamma\left(\sum_{j=1}^{\infty} X_j \cdot e_j\right).$$

Thus, the option pricing problem is represented as an infinite-dimensional integral, i.e.,

$$\mathbb{E}(f(\mathbf{X})) = \int_{\mathscr{X}} f(\mathbf{x})\,\mu(\mathrm{d}\mathbf{x}),$$

where we define $\mathscr{X}$ and μ later in Chapter 3. To approximate that integral, we illustrate a finite sample average,

$$\mathbb{E}(f(\mathbf{x})) \approx \frac{1}{n}\sum_{i=1}^{n} f(x_{i,1},\ldots,x_{i,d},0,\ldots)$$

for some well-chosen $\mathbf{x}_i \in \mathscr{R}^d$. This corresponds to the Karhunen-Loève expansion and Brownian bridge expansion truncated at d terms.

2.2 Low Discrepancy Sampling

The low discrepancy sequence is closely related to the implementation of QMC methods, which perform the following approximation scheme

$$\int_{[0,1]^d} f(\mathbf{x})\,\mathrm{d}\,\mathbf{x} \approx \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i), \tag{2.9}$$

where $\mathbf{x}_1,\ldots,\mathbf{x}_n$ are evenly distributed points in $[0,1]^d$. Note that the dependence of QMC methods on the problem dimension is one of the features that most distinguishes them from Monte Carlo. In MC simulation, taking an scalar i.i.d. sequence of uniform distribution $U_1,U_2,\ldots$ and forming vector $(U_1,\ldots,U_d),(U_{d+1},\ldots,U_{2d}),\ldots$ returning an i.i.d random sequence from the d-dimensional hypercube. In QMC, the construction of the points depends explicitly on the dimension of the problem, which means one can not construct the vector by taking sets of d consecutive elements from a scalar sequence.

The goal of the construction of low discrepancy sequences is to find points $\mathbf{x}_i$ such that the error in the approximation (2.9) is small for a broad class of integrands f. This is equivalent to choosing the points $\mathbf{x}_i$ which fill in the hypercube uniformly. To facilitate the description, we cite the definition for the discrepancy, which is a precise measure of the uniformity. From [Nie92, DP10]

Definition 2.2 *Let $\mathscr{P} = \{\mathbf{x}_i\}_n^{i=1}$ be a finite point set in $[0,1]^d$. For an arbitrary subset J of $[0,1]^d$, define*

$$A(J;\mathscr{P}) := \sum_{i=1}^{n} \psi_J(\mathbf{x}_i),$$

where ψ_J is the characteristic function of J. Thus $A(J;\mathscr{P})$ is the counting function that indicates the number of i with $1 \le i \le n$ for which $\mathbf{x}_i \in J$. Let $\mathscr{J}$ be a nonempty family of Lebesgue-measurable subset of $[0,1]^d$, then a general notion of the discrepancy of the point set $\mathscr{P}$ is defined by

$$D_n(\mathscr{J};\mathscr{P}) = \sup_{J\in\mathscr{J}} \left| \frac{A(J;\mathscr{P})}{n} - \lambda_d(J) \right|,$$

where $\lambda_d(J)$ is the $d-$dimensional Lebesgue measure of J.

Note that if the points of $\mathscr{P}$ have a very uniform distribution over $[0,1]^d$, then the discrepancy is close to 0 for some well-chosen collection of Borel set J, e.g., all subintervals of $[0,1]^d$. By suitable specializations of the family $\mathscr{J}$, one can obtain one of the most important concepts: the *star discrepancy*.

Definition 2.3 *The star discrepancy $D_n^* = D_n^*(\mathbf{x}_1,\ldots,\mathbf{x}_n)$ of the point set $\mathscr{P}$ is defined by*

$$D_n^*(\mathscr{P}) = D_n(\mathscr{J};\mathscr{P}),$$

where $\mathscr{J}$ is the family of all sub-cubes of $[0,1]^d$ of the form $\prod_{i=1}^d [0,u_i)$ with $0 \le u_i \le 1$ for $1 \le i \le n$.

In addition to the appealing property as a measure of uniformity, it plays a more important role in building the error bound for the approximation (2.9). This is known as the *Koksma-Hlawka inequality*. [Hla61].

Theorem 2.4 *If f has bounded variation $V(f)$ on $[0,1]^d$ in the sense of Hardy and Krause, then, for any points $\mathbf{x}_1,\ldots,\mathbf{x}_n \in [0,1]^d$, we have*

$$\left| \int_{[0,1]^d} f(\mathbf{x})\,\mathrm{d}\,\mathbf{x} - \frac{1}{n}\sum_{i=1}^n f(\mathbf{x}_i) \right| \le V(f) D_n^*(\mathbf{x}_1,\ldots,\mathbf{x}_n).$$

Note that $V(f)$ is a measure for the variation of f, the precise definition can be found in [Nie92]. Koksma-Hlawka inequality indicates that the point sets with small star discrepancy guarantee small approximation errors in QMC method. From [Nie92], for any $n \geq 2, d \geq 1$, the least order of magnitude that can be achieved is

$$D_n^*(\mathscr{P}) = \mathscr{O}(n^{-1}\log(n)^{d-1}), \tag{2.10}$$

with the leading constant independent of n. Any point set $\mathscr{P}$ satisfying (2.10) is called a *low discrepancy set*. The most useful concept is the *low discrepancy sequence*:

Definition 2.4 *Given an infinite sequence $\mathscr{S}$ of points in $[0,1]^d$ such that for all $n \geq 2$, the star discrepancy $D_n^*(\mathscr{S})$ of the first n terms of $\mathscr{S}$ satisfies*

$$D_n^* = \mathscr{O}(n^{-1}\log(n)^d),$$

where the leading constant is independent of n. Then, this sequence is called the low discrepancy sequence.

Next, various constructions of low discrepancy sequences are reviewed and several types of low discrepancy sequences are studied.

2.2.1 Nets and sequences. Before proceeding with the construction of low discrepancy sequences, we introduce a specific class of one-dimensional low discrepancy sequences called *Van der Corput sequences*, which serves as the key element of many multidimensional constructions.

As in [Nie92], for any integer $b > 2$, any integer n can be represented as the following unique digit expansion

$$n = \sum_{j=0}^{\infty} a_j(n) b^j \tag{2.11}$$

in base b, with all but finitely many of the coefficient $a_j(n)$ equal to zero.

Definition 2.5 *For any integer $b \geq 2$, the radical-inverse function ψ_b in base b is defined by*

$$\psi_b(n) = \sum_{j=0}^{\infty} \frac{a_j(n)}{b^{j+1}}, \tag{2.12}$$

where n is given by its digit expansion in (2.11)*. Then the Van der Corput sequence in base b is the sequence $x_0, x_1, \ldots$ with $x_n = \psi_b(n)$ for all $n \geq 0$.*

All Van der Corput sequences are low discrepancy sequences. They can be used as a building block to construct higher dimension low discrepancy sequences. A valuable tool to construct low discrepancy sequences is (t,m,s)-net and a (t,s)-sequence introduced by [Nie87]. The parameter s denotes the dimension of the sequence, t is the quantitative measure of the uniformity. A (t,m,s)-net is a finite set of points in $[0,1]^s$ with uniformity t; a (t,s)-sequence is a sequence of points, where certain segments of it form (t,m,s)-nets. According to [Nie10],

Definition 2.6 *Let $s \geq 1, b \geq 2$, and $0 \leq t \leq m$ be integers and $\mathscr{M}_{b,m,t}^{(s)}$ be the collection of all sub-cubes J of $[0,1]^s$ of the form*

$$J = \prod_{i=1}^{s} \left[a_i b^{-d_i}, (a_i + 1) b^{-d_i} \right]$$

with integers $d_i \geq 0$ and $0 \leq a_i \leq b^{d_i}$ for $1 \leq i \leq s$ and $\lambda_s(J) = b^{t-m}$. Then an $(\mathscr{M}_{b,m,t}^{(s)}; \lambda_s)$-uniform point set consisting of b^m points in $[0,1]^s$ is called as (t,m,s)-net in base b. Here, $(\mathscr{M}_{b,m,t}^{(s)}; \lambda_s)$-uniformness means $\frac{A(J;p)}{n} = \lambda_s(J)$ as defined in Definition 2.2.

Definition 2.7 *A sequence of points $x_1, x_2, \ldots$ in $[0,1)^d$ is a $(t,s)-$sequence in base b if for all $m > t$ each segment $\{x_i : jb^m < i \leq (j+1)b^m\}, j = 0, 1 \ldots,$ is a $(t,m,s)-$ net in base b.*

Remark 2.2 *It should be noted from the definitions that the smaller values of t indicate the greater uniformity. In addition, all other things be equal, a smaller base value b is preferred because the uniformity of $(t,m,s)-$nets and $(t,s)-$sequences are exhibited in set of b^m*

points; the larger the value b, the more points are required. Please refer to [Nie92, DP10] for an extensive analysis of discrepancy bounds for $(t,m,s)-$*nets and* $(t,s)-$*sequences.*

There are a few types of specific nets and sequences with important applications and the ways to generate those sequences are different. Those low discrepancy sequences are in arbitrary dimensions, examples include the Halton sequence, the Faure sequence and the Sobol' sequence. In the following, the construction and properties of the Sobol' sequence are discussed because of the important application in financial calculations. Please refer to [Nie92] for the original discussion of the Halton sequence and the Faure sequence.

2.2.2 Sobol' sequences. We now describe how to construct an $d-$dimensional Sobol' sequence. Sobol' [Sob67] gave the first construction of Sobol' sequence using primitive polynomials. Simply speaking, the construction method of Sobol' sequences starts from the Van der Corput sequence, the various coordinates of an $d-$dimensional Sobol' sequence are from permutations of segments of the Van der Corput sequence by taking multiplying (binary) expansions of consecutive integers by a set of generator matrices, one for each dimension. Note that each coordinate of a Sobol' sequence has its own generator matrix $V_l, l = 1,2,\ldots$, where the elements of V_l are either 0 or 1, the columns of V_l are the binary expansions of a set of direction numbers $v_{1,l},\ldots,v_{r,l}$, r is the number of terms in the binary expansion of k_l as follows

$$k_l = a_{0,l}(k) + 2a_{1,l}(k) + \cdots + 2^{r-1}a_{r-1,l}(k).$$

Consider

$$\begin{pmatrix} y_{1,l}(k) \\ y_{2,l}(k) \\ \vdots \\ y_{r,l}(k) \end{pmatrix} = V_l \begin{pmatrix} a_{0,l}(k) \\ a_{1,l}(k) \\ \vdots \\ a_{r-1,l}(k) \end{pmatrix} \mod 2; \tag{2.13}$$

here $y_{1,l}(k),\ldots,y_{r,l}(k)$ are coefficients of the binary expansion of the kth point in the sequence, which means

$$x_{k_l} = \frac{y_{1,l}(k_l)}{2} + \frac{y_{2,l}(k_l)}{4} + \cdots + \frac{y_{r,l}(k_l)}{2^r}.$$

We can also represent the formulation in (2.13) as

$$a_{0,l}(k_l)v_{1,l} \oplus a_{1,l}(k_l)v_{2,l} \oplus \cdots \oplus a_{r-1,l}(k_l)v_{r,l},$$

with $v_{j,l}$ as the jth column of the generator matrix V_l, which is called the direction number, and $\oplus$ as the binary addition.

The generator matrix V_l is specified by choosing a primitive polynomial over binary arithmetic in the form of

$$x^q + c_1 x^{q-1} + \cdots + c_{q-1}x + 1,$$

where c_i are either 0 or 1. The primitive polynomial's coefficients must be chosen such that it is irreducible and the smallest power p for which the polynomial divides $x^p + 1$ is $2^q - 1$. This polynomial therefore defines a recurrence relation

$$m_{j,l} = 2c_{1,l}m_{j-1,l} \oplus 2^2 c_{2,l}m_{j-2,l} \oplus \cdots \oplus 2^{q-1}c_{q-1,l}m_{j-q+1,l} \oplus 2^q m_{j-q,l} \oplus m_{j-q,l},$$

and the direction numbers are defined by

$$v_{j,l} = \frac{m_j}{2^j}. \tag{2.14}$$

Note that $m_{j,l}$ must be specified initially to fully define the direction numbers, and it is general to choose an odd integer less than 2^j for the initial value of $m_{j,l}$.

As an example, suppose that $d = 3$ and choose the primitive polynomial as $x^3 + x + 1$, so that the recurrence relation for $m_{j,l}$ becomes

$$m_{j,l} = 4m_{j-2,l} \oplus 8m_{j-3,l} \oplus m_{j-3,l}$$

with the initial values $m_{1,l} = 1, m_{2,l} = 3, m_{3,l} = 7$. We can then compute $m_{4,l}, m_{5,l}, \ldots$ using the above recursive equation,

$$m_{4,l} = 12 \oplus 8 \oplus 1 = 5, \qquad m_{5,l} = 28 \oplus 24 \oplus 3 = 7,$$

and so on. By using (2.14), the first five direction numbers are

$$v_{1,l} = 0.1, v_{2,l} = 0.11, v_{3,l} = 0.111, v_{4,l} = 0.0101, v_{5,l} = 0.00111,$$

and the corresponding generator matrix is

$$V_l = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

Then, the calculation of the sequence $x_1, x_2, \ldots$ is performed by (2.13). The following shows how to create the first three coordinates,

$$V_l \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, V_l \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, V_l \begin{pmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

and the 3-dimensional point is $(\frac{1}{2}, \frac{3}{4}, \frac{1}{4})$.

2.2.3 Lattice rules. The construction methods discussed in Section 2.2.1 and Section 2.2.2 are exclusively based on extending the Van der Corput sequence to higher dimensions. The lattice rules framework is a different mechanism for generating low discrepancy

sequences which are suitable for multivariate integrations. We briefly talk about the basic concepts regarding generating the lattice methods. See Chapter 5 of [Nie92], [HHLL00] and [LL00] for a deeper understanding of the lattice rule.

The most popular lattice rules is the rank-1 lattice rule for n points in dimension d which is constructed as

$$\left\{ \frac{k}{n}\mathbf{v} \mod 1, \quad k = 0,1,\ldots,n-1 \right\},$$

where $\mathbf{v}$ is an $d-$dimensional vector of integers. The modular operation is applied separately to each dimension of the sequence. It is required that n and the coordinates of $\mathbf{v}$ have 1 as their greatest common divisor such that the point set defined above has n distinct points. More generally, there exists an $n-$point lattice rule for rank r,

$$\left\{ \sum_{i=1}^{r} \frac{k_i}{n_i} v_i \mod 1, \quad k_i = 0,1,\ldots,n_i-1, \quad i = 1,\ldots,r \right\},$$

where $v_1,\ldots,v_r$ are linearly independent integer vectors and $n_1,\ldots,n_r$ are integers greater or equal to 2 with each n_i dividing n_{i+1} and $n_1 \cdots n_r = n$. Note that v_i and n_i must have 1 as their greatest common divisor to guarantee the distinctness of n points.

Different from the nets and sequences framework, the lattice method is only applicable to fixed-size point sets rather than the infinite-dimensional sequence. Please refer to [HHLL00] for extending fixed-size lattice rules to infinite-dimensional sequences.

2.3 Computing Expectations by Sample Averages

As we mentioned in previous sections, Monte Carlo algorithms are implemented in this thesis as a computational tool for approximating infinite-dimensional integrations. The following provides a brief introduction to Monte Carlo simulation algorithms. Then, we introduce two important algorithms: the single-level algorithm and the multi-level algorithm designed for infinite-dimensional integration problems.

2.3.1 Monte Carlo simulations. Monte Carlo methods are a class of computational algo-

rithms that rely on repeated sampling to compute their results. In the scenario of this thesis, Monte Carlo type algorithms are especially applied to high dimensional integration problems. For examples, for the computational finance examples mentioned in Section 1.1.2, the dimension of the integration could go as high as hundreds or thousands, even infinity. We need to compute an expected value of a function $f(\cdot)$ given a specified distribution $\rho(\cdot)$ over $\mathbb{R}^d$:

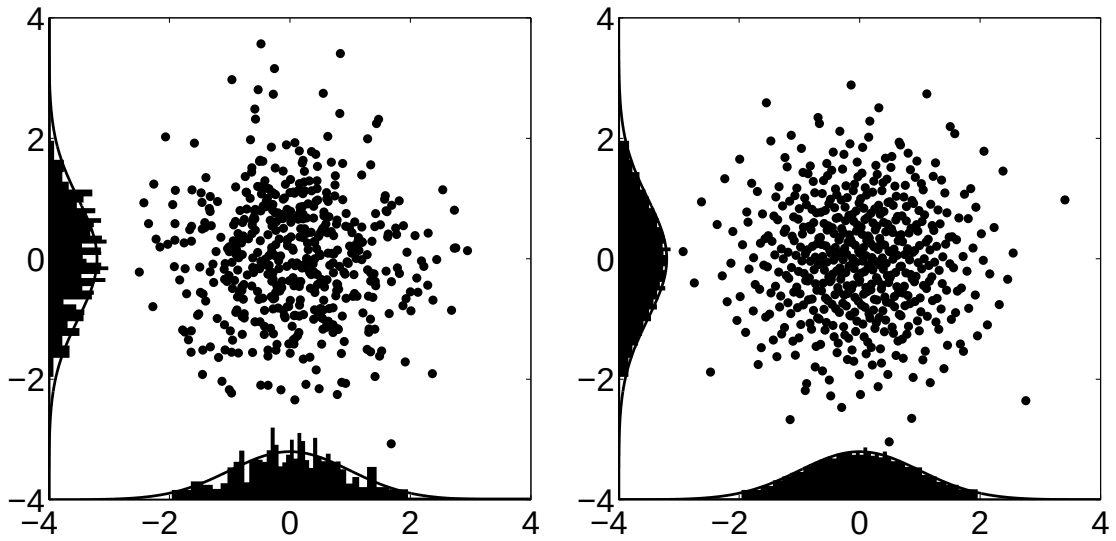
$$I(f) = \mathbb{E}[f(\mathbf{X})] = \int_{\mathbb{R}^d} f(\mathbf{x})\rho(\mathbf{x})\,\mathrm{d}\mathbf{x}. \tag{2.15}$$

The general form of the simple Monte Carlo simulation of the integral (2.15) can be summarized as follows:

- Run the random number generator to draw n i.i.d. random vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n$ from the target distribution ρ.
- For each drawn random vector $\mathbf{X}_i$, evaluate the function f.
- Compute the sample average to obtain the Monte Carlo estimator:

$$\hat{I}(f) := \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{X}_i). \tag{2.16}$$

In terms of the quasi-Monte Carlo methods, the only difference to the Monte Carlo method is that it is a purely deterministic method while the Monte Carlo method is randomized. In Monte Carlo methods, pseudo-random sequences are used to approximate the integration. Figure 2.3 demonstrates the inverse normal transformation of 2-dimensional pseudo random sequences and Sobol' sequences. We can see that pseudo random sequences are randomly spread out, there are regions where there are no points and other regions where there are clusters of points. In addition, the marginal distributions deviate from the normal distribution. Sobol' sequences are evenly dispersed, which yields the uniformity, and the marginal distributions are close to the normal distribution. Low discrepancy sequences are discussed in details in Section 2.2 of Chapter 2.



(a) Pseudo Random Sequence (b) Sobol' Sequence

Figure 2.3. The plot shows the inverse normal transformation of 2-dimensional pseudo random sequences and Sobol' sequences. The bell shape curves are marginal distribution

2.3.1.1 The single level algorithm. In this thesis we study the integration on the sequence space $\mathbb{R}^{\mathbb{N}}$, as we wish to compute the expectation

$$I(f) = \mathbb{E}(f(X_1, X_2, \ldots)) \tag{2.17}$$

for functions $f : \mathbb{R}^{\mathbb{N}} \to \mathbb{R}$, where $(X_1, X_2, \ldots)$ is an i.i.d. sequence of random variables with a common univariate distribution function ρ_1 on a Borel subset $D \subseteq \mathbb{R}$. Important examples include the uniform distribution ρ_1 on $[0,1]$ and the standard normal distribution on the real line. In a reasonable approach to compute the integral $I(f)$ all but finitely many random variables X_j are replaced by some nominal value c of the distribution ρ_1, and the aim is to construct deterministic quadrature formulas based on a finite number of function values $f(x_{i,1}, \ldots, x_{i,d_i}, c, c, \ldots)$.

Remark 2.3 *In the rest of thesis, $X_i, i = 1,2,\ldots$ represent the standard normal random variables while $x_i, i = 1,2,\ldots$ are sampled random sequence approximating the distribution of X_i.*

Define $\Psi_{1:d}$ as a projection operator which projects functions in $\mathbb{R}^{\mathbb{N}}$ to $\mathbb{R}^d$. Let $\Psi_{1:d}f$ denote the corresponding function of the first d variables, i.e.,

$$(\Psi_{1:d}f)(\mathbf{x}) = f(x_1,\dots,x_d,c,c\dots) \tag{2.18}$$

for $d \in \mathbb{N}$ and $\mathbf{x} \in \mathbb{R}^{\mathbb{N}}$. For example, an equal weight quadrature formula that uses n function values in a fixed dimension d takes the form

$$\hat{I}(f) = \frac{1}{n}\sum_{i=1}^{n}(\Psi_{1:d}f)(\mathbf{x}_i) = \frac{1}{n}\sum_{i=1}^{n} f\left(x_{i,1},\dots,x_{i,d},c,c,\dots\right) \tag{2.19}$$

for some design of points $(x_{i,1},\dots,x_{i,d}) \in \mathbb{R}^d$. This is the so-called single level algorithm. The cost of a single function evaluation is assumed to be given by d^s for some $s \geq 0$, so that the cost of the quadrature formula $\hat{I}$ is given by

$$N = n \cdot d^s,$$

which corresponds to the fixed subspace sampling model of [CDMGR09]. Clearly, $s = 0$ is unrealistic in practical applications, while $s = 1$ is a reasonable choice in many situations.

Let us return to the option pricing example to demonstrate the single-level algorithm. The aim is to evaluate $I(g) = \mathbb{E}[g(B(\cdot;X_1,X_2,\dots))] = \mathbb{E}[f(X_1,X_2,\dots)]$, where g is the payoff and I is the option price. The approximation of I takes the form of an equally weighted sample average:

$$\hat{I} = \frac{1}{n}\sum_{i=1}^{n} g\left(\hat{B}_d(\cdot;x_{i,1},\dots,x_{i,d})\right). \tag{2.20}$$

Before developing a general theory for this problem, a simple example is given to illustrate how the error of $\hat{I}$ depends both on n and d and how the error analysis is facilitated by splitting the error into two parts. Evaluate $\mathbb{E}\left[\int_0^1 B^2(t)\,\mathrm{d}t\right]$ using (2.20). In this case

$$\begin{aligned} g\left(B(\cdot;X_1,X_2,\dots)\right) &= \int_0^1 [B(\cdot;X_1,X_2,\dots)]^2\,\mathrm{d}t = \sum_{j=1}^{\infty}\frac{X_j^2}{(j-\frac{1}{2})^2\pi^2}, \\ I = \mathbb{E}[g(B(\cdot;X_1,X_2,\dots))] &= \sum_{j=1}^{\infty}\frac{\mathbb{E}[X_j^2]}{(j-\frac{1}{2})^2\pi^2} = \sum_{j=1}^{\infty}\frac{1}{(j-\frac{1}{2})^2\pi^2} = \frac{1}{2}. \end{aligned}$$

The first d terms in the series for I can be identified as

$$\begin{aligned} I_d &= \mathbb{E}[g(B(\cdot;X_1,X_2,\ldots))|X_{d+1}=X_{d+2}=\cdots=0] \\ &= \sum_{j=1}^{d}\frac{\mathbb{E}[X_j^2]}{(j-\frac{1}{2})^2\pi^2}=\sum_{j=1}^{d}\frac{1}{(j-\frac{1}{2})^2\pi^2}. \end{aligned}$$

The expression I_d is introduced here to facilitate splitting the error. From (2.20), the estimator for I is

$$\hat{I}=\frac{1}{n}\sum_{i=1}^{n}g(\hat{B}_d(\cdot;x_{i,1},\ldots,x_{i,d})) \quad = \quad \sum_{j=1}^{d}\frac{\frac{1}{n}\sum_{i=1}^{n}x_{i,j}^2}{(j-\frac{1}{2})^2\pi^2}.$$

Then, the error of approximating I can be written as a sum of two parts:

$$\begin{aligned} \overbrace{I-\hat{I}}^{\text{error}} &= \overbrace{I-I_d}^{\text{truncated expansion error}} + \overbrace{I_d-\hat{I}}^{\text{finite sample error}} \\ &= \underbrace{\sum_{j=d+1}^{\infty}\frac{1}{\left(j-\frac{1}{2}\right)^2\pi^2}}_{\text{independent of design}} + \underbrace{\sum_{j=1}^{d}\frac{\mathbb{E}[X_j^2]-\frac{1}{n}\sum_{i=1}^{n}x_{i,j}^2}{\left(j-\frac{1}{2}\right)^2\pi^2}}_{\text{independent of } X_{d+1},\ldots}. \end{aligned}$$

The first term is independent of the design $\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$, whereas the second term is independent of the coordinates after dimension d.

Simple Monte Carlo sampling If $\{\mathbf{x}_1,\ldots,\mathbf{x}_n\}$ is a set of d-dimensional i.i.d. standard normal random variables, the explicit mean square error (MSE) can be written as a sum of the squared bias and the variance:

$$\begin{aligned} \mathrm{MSE}(\hat{I}) &= \mathbb{E}\left(I-\hat{I}\right)^2=(I-I_d)^2+\mathbb{E}(I_d-\hat{I})^2 \\ &= \underbrace{\left(\sum_{j=d+1}^{\infty}\frac{1}{\left(j-\frac{1}{2}\right)^2\pi^2}\right)^2}_{\text{bias}^2} + \underbrace{\frac{1}{n}\sum_{j=1}^{d}\frac{2}{\left(j-\frac{1}{2}\right)^4\pi^4}}_{\text{variance}}. \end{aligned}$$

The bias depends primarily on the truncated dimension d, while the variance depends primarily on the sample size n. They can be approximated by

$$\text{bias} = \sum_{j=d+1}^{\infty} \frac{1}{\pi^2(j-1/2)^2} \sim \int_d^{\infty} \frac{1}{\pi^2 x^2}\,\mathrm{d}x = \frac{1}{\pi^2 d}, \quad \text{as } d \to \infty.$$

$$\begin{aligned}\text{variance} &= \frac{2}{n}\sum_{j=1}^{d} \frac{1}{\pi^4(j-1/2)^4} = \frac{2}{n}\frac{1}{\pi^4}\left[\sum_{j=1}^{\infty} \frac{1}{(j-1/2)^4} - \sum_{j=d+1}^{\infty} \frac{1}{(j-1/2)^4}\right] \\ &\sim \frac{2}{n}\frac{1}{\pi^4}\left(\frac{\pi^4}{6} - \int_d^{\infty} \frac{1}{x^4}\,\mathrm{d}x\right) = \frac{1}{3n}\left(1 - \frac{2}{\pi^4 d^3}\right), \quad \text{as } d \to \infty.\end{aligned}$$

Hence, the root mean square error (RMSE) is:

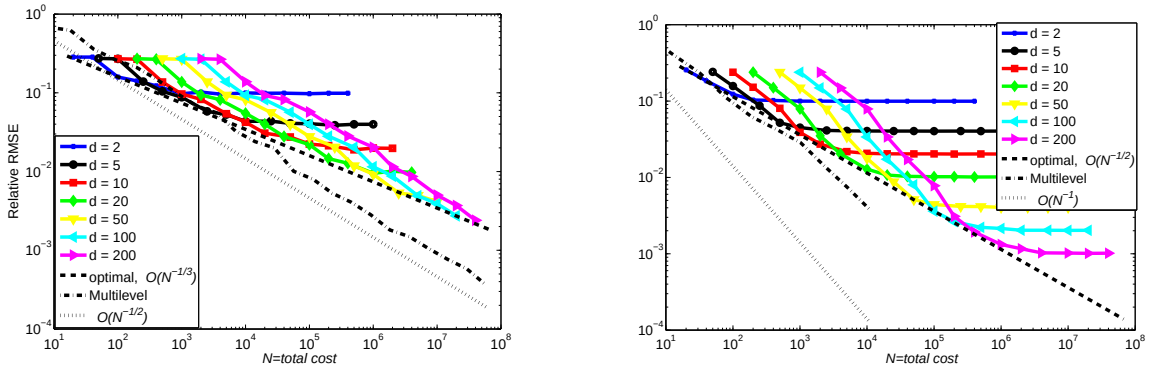
$$\text{RMSE}(\hat{I}) \sim \sqrt{\frac{1}{\pi^4 d^2} + \frac{1}{3n}\left(1 - \frac{2}{\pi^4 d^3}\right)} \sim \sqrt{\frac{1}{\pi^4 d^2} + \frac{1}{3n}}, \quad \text{as } d \to \infty.$$

Quasi-Monte Carlo sampling If $\{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ is a set of d-dimensional vectors, whose components are the inverse normal transformation of the low discrepancy sequence, e.g., a scrambled Sobol' sequence, it is technically more difficult to get an explicit expression for the variance, however, the bias is the same. It can be observed from the result of numerical experiments that

$$\text{RMSE}(\hat{I}) \sim \sqrt{\frac{1}{\pi^4 d^2} + \frac{C}{n^2}}, \quad \text{as } d \to \infty.$$

Figure 2.4 shows that for small, fixed values of d, the relative RMSE converges very quickly to the limiting value, i.e., the bias. For large values of d, the relative RMSE is dominated by the sampling error for moderate values of n. The errors obtained using the optimal n and d for a given $N = nd$ are obtained empirically. The optimal convergence rate in the simple random sequence case is $\mathcal{O}(N^{-1/3})$, whereas the scrambled Sobol' sequence can achieve a superior $\mathcal{O}(N^{-1/2})$ convergence rate. These empirical orders of convergence, $\beta = 1/3$ and $1/2$ are determined by the formula

$$\beta = \text{argmin}_b C(b), \quad \text{where } C(b) = \max\{C : C\left(N/N_0\right)^{-b} \leq \text{RMSE}(N)\ \forall N\}$$



(a) Simple Random Sequences

(b) Scrambled Sobol' Sequences

Figure 2.4. The root mean square error for the single-level algorithm.

The value of N_0 corresponds to the midpoint of the values of N considered in the numerical experiments.

In the derivation above, the approximation error of I is conveniently split into two parts, i.e., the truncated expansion error, and the finite sample error. This split is exploited in the later parts of this thesis where worst-case error bounds for a general function is derived.

2.3.1.2 The multi-level algorithm. Instead of a single truncating dimension d used in the single-level algorithm, consider a sequence of increasing dimensions $0 = d_0 < d_1 < \cdots < d_{L+1} = \infty$. We adopt the truncating projection operator introduce in (2.18). A multi-level algorithm is based on the increasing sequence of dimensions, and functions f of infinitely many variables are decomposed as

$$f = \Psi_{1:d_1} f + \sum_{\ell=2}^{L} (\Psi_{1:d_\ell} f - \Psi_{1:d_{\ell-1}} f) + f - \Psi_{1:d_L} f. \tag{2.21}$$

By convention, define $\Psi_0 f = 0$ and $\Psi_\infty f = f$. Roughly speaking, $(\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})f$ yields the part of f that depends on the first d_ℓ variables, but not only on the first $d_{\ell-1}$ variables. For example, let us consider the function $f(\mathbf{x}) = x_1(x_2 + x_3)$ as well as $c = 0$. Then $(\Psi_{1:4} - \Psi_{1:2})f(\mathbf{x}) = x_1 x_3$ and $\Psi_{1:2} f(\mathbf{x}) = x_1 x_2$. Using this notation, the telescopic sum for the

infinite-dimensional expectation in (2.17) can be rewritten as

$$I(f) = \sum_{l=1}^{L+1} I_l \left(\Psi_{1:d_\ell} f - \Psi_{1:d_{\ell-1}} f \right), \tag{2.22}$$

where $I_1, \ldots, I_L$ are the suitable quadrature formulas in dimension $d_1, \ldots, d_L$.

The multi-level algorithm for approximating $I(f)$ approximates each term in (2.22), except for the last one, by a Monte Carlo or quasi-Monte Carlo algorithm of the form of (2.16). The last term in (2.22) is approximated by 0 since it is assumed to be small. Specifically, the multi-level algorithm takes the form

$$\begin{aligned} \hat{I}(f) &= \sum_{\ell=1}^{L+1} \hat{I}_\ell(\Psi_{1:d_\ell} f - \Psi_{1:d_{\ell-1}} f), \\ \hat{I}_\ell(\Psi_{1:d_\ell} f - \Psi_{1:d_{\ell-1}} f) &= \frac{1}{n_l} \sum_{i=1}^{n_l} \left[f(\mathbf{x}^{(l)}_{i,1:d_\ell}, \mathbf{c}) - f(\mathbf{x}^{(l)}_{i,1:d_{\ell-1}}, \mathbf{c}) \right], \\ & \qquad \ell = 1, \ldots, L, \\ \hat{I}_{L+1}(\Psi_{1:d_{L+1}} f - \Psi_{1:d_L} f) &= 0. \end{aligned}$$

A single n_ℓ-point design of dimension d_ℓ is used to compute the level ℓ approximation, $\hat{I}_\ell$. All dimensions are used to evaluate $f(\mathbf{x}^{(\ell)}_{i,1:d_\ell}, \mathbf{c})$ and the first $d_{\ell-1}$ dimensions are used to evaluate $f(\mathbf{x}^{(l)}_{i,1:d_{\ell-1}}, \mathbf{c})$.

As in [CDMGR09], the cost of the multi-level algorithm $\hat{I}$ corresponds to variable subspace sampling, i.e., the cost of evaluating $f(\mathbf{x}^{(\ell)}_{i,1:d_\ell}, \mathbf{c}) - f(\mathbf{x}^{(\ell)}_{i,1:d_{\ell-1}}, \mathbf{c})$ for a single i is proportional to d_ℓ. So the cost of $\hat{I}_\ell$ is proportional to $n_\ell d_\ell$, and the total computational cost of $\hat{I}(f)$ is of $\mathscr{O}(N)$, where

$$N = \sum_{\ell=1}^{L} n_l d_\ell \tag{2.24}$$

Refer to [KSWW10], there is a more general definition of the cost function than (2.24). For each f and a subset $u \in \mathbb{U} \in \mathbb{R}^{\mathbb{N}}$, the cost of computing $f(\mathbf{x}_u; \mathbf{c})$ is equal to $\$\left(|u|\right)$ for a given cost function $\$: \mathbb{N} \to [1, \infty)$. One example of the cost function is

$$\$(d) = \left[\max(1, d)\right]^s \quad \text{for } s \geq 0. \tag{2.25}$$

The cost function definition in (2.24) is corresponding to choose $s = 1$ in (2.25). However, in practice, s could be different from 1 because of various types of function f. For example, pricing the Asian call option in Section 1.1.2, yields the cost function $N = \sum_{\ell=1}^{L} n_\ell d_\ell^{d_\ell m}$, where m is the time discretization dimension when computing the mean value of the stock price. In the rest of the thesis, we focus on the more general definition of the computational cost.

As shall be seen, the advantage of the multi-level algorithm is that one can choose n_ℓ to decrease as d_ℓ increases to obtain greater efficiency than the single-level algorithm.

CHAPTER 3

SPACES OF FUNCTIONS

3.1 Reproducing Kernel Hilbert Spaces

In this section, we begin with the introductory discussion of the reproducing kernel Hilbert spaces. The integrand f, which depends on a countably infinite number of variables, is constructed as the countable sum of functions of a finite number of variables. The Hilbert space containing f is likewise constructed as the tensor product space of a countable number of reproducing kernel Hilbert spaces. See similar work in [KSWW10, Owe98].

Reproducing kernel Hilbert spaces arise in a number of areas, including approximation theory, statistics, machine learning theory, group representation theory and various areas of complex analysis. In this thesis, we consider Hilbert spaces over the field of real numbers, $\mathbb{R}$. The reproducing kernel Hilbert space is a Hilbert space of functions in which pointwise evaluation is a continuous linear functional. Equivalently, they are spaces that can be defined by reproducing kernels.

Definition 3.1 *Given a set X, let $\mathscr{F}(X,\mathbb{R})$ denote the set of all functions from X to $\mathbb{R}$. This is a vector space over $\mathbb{R}$ with the usual operations of addition, $(f+g)(x) = f(x)+g(x)$, and scalar multiplication$(\lambda \cdot f)(x) = \lambda \cdot (f(x))$. We say that $\mathscr{H}$ is a* ***reproducing kernel Hilbert space (RKHS)*** *on X over $\mathbb{R}$, provided that:*

- *$\mathscr{H}$ is a vector subspace of $\mathscr{F}(X,\mathbb{R})$,*
- *$\mathscr{H}$ is equipped with an inner product, $\langle , \rangle$,*
- *$\mathscr{H}$ is complete under the norm induced by the inner product, making it into a Hilbert space,*
- *for every $y \in X$, the linear evaluation functional $E_y : \mathscr{H} \to \mathbb{R}$, defined by $E_y(f) = f(y)$, is bounded.*

Note that if $\mathscr{H}$ is a RKHS on X, then since every bounded linear functional is given by the inner product with a unique vector in $\mathscr{H}$, there exists a unique vector $K_x \in \mathscr{H}$ for every $x \in X$ such that for every $f \in \mathscr{H}, f(x) = \langle f, K_x \rangle$. The Riesz representation theorem implies the following definition of the reproducing kernel.

Definition 3.2 *A function*

$$K : X \times X \to \mathbb{R}$$

is a reproducing kernel of the Hilbert space $\mathscr{H}$ if and only if

A1. $\forall x \in X, K(\cdot, x) \in \mathscr{H}$

A2. $\forall x \in X, \forall f \in \mathscr{H}, \langle f, K(\cdot, x) \rangle = f(x)$.

The last condition is called the reproducing property, which means the value of a function f at any point x is reproduced by the inner product of f with the kernel K. It is also clear that

$$\forall (x, y) \in X \times X \qquad K(x, y) = \langle K(\cdot, x), K(\cdot, y) \rangle.$$

Remark 3.1 *The reproducing kernel $K(\cdot, \cdot)$ is entirely determined by the RKHS $\mathscr{H}$ because the Riesz representation theorem guarantees, for every x in X, that the element K_x satisfying $f \in \mathscr{H}, f(x) = \langle f, K_x \rangle$ is unique.*

Example 3.1 *Let us examine a specific example for RKHS. Given $(e_1, e_2, \ldots, e_n)$ as an orthonormal basis in $\mathscr{H}$ and define the kernel function*

$$K(x, y) = \sum_{i=1}^{n} e_i(x) e_i(y).$$

Then for any $y \in X$,

$$K(\cdot, y) = \sum_{i=1}^{n} e_i(y) e_i(\cdot)$$

belongs to $\mathscr{H}$ and for any function

$$\phi(\cdot) = \sum_{i=1}^{n} \lambda_i e_i(\cdot)$$

in $\mathscr{H}$, we have

$$\begin{aligned} \langle \phi, K(\cdot, y) \rangle_{\mathscr{H}} &= \left\langle \sum_{i=1}^{n} \lambda_i e_i(\cdot), \sum_{i=1}^{n} e_i(y) e_i(\cdot) \right\rangle_{\mathscr{H}} = \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i e_j(y) \langle e_i, e_j \rangle_{\mathscr{H}} \\ &= \sum_{i=1}^{n} \lambda_i e_i(y) \\ &= \phi(y). \end{aligned}$$

Thus, K is the reproducing kernel for $\mathscr{H}$. In addition to the reproducing property described in Definition 3.2, for the point evaluation, there is also a reproducing property for the bounded, linear functional of functions within the RKHS. This nice property is used extensively later for quantifying the error in the Monte Carlo algorithm. Any bounded, linear functional L on $\mathscr{H}(K)$ may be represented as

$$L(g) = \langle g_L, g \rangle_{\mathscr{H}(K)}, \qquad \forall g \in \mathscr{H}(K),$$

for some particular $g_L \in \mathscr{H}(K)$, and

$$g_L(x) = \langle K(\cdot, x), g_L \rangle_{\mathscr{H}(K)} = L(K(\cdot, x)).$$

3.2 Construction of Space of Integrands

The basic problem is to measure how well $I(f) = \mathbb{E}[f(X_1, X_2, \ldots)]$ is approximated by $\hat{I}(f)$, where $(X_1, X_2, \ldots)$ is an i.i.d. random sequence with common probability density function $\rho_1(\cdot)$. The support of ρ_1 is assumed to be D, where D is some open, half-open, or closed interval, which may be finite, semi-infinite or infinite. For example, ρ_1 might be the uniform density on $D = [0,1]$ or the standard normal density, which is defined on $D = \mathbb{R}$. The domain of f can be considered to be $D^{\mathbb{N}}$, where $\mathbb{N}$ is the set of natural numbers. Here $D^{\mathbb{N}}$ is the set of infinite sequences whose elements lie in D. The function f, which depends

on a countably infinite number of variables, is constructed as the countable sum of functions of a finite number of variables. The Hilbert space containing f is likewise constructed as the tensor product space of a countable number of reproducing kernel Hilbert spaces. See similar work in [KSWW10, Owe98], where the function with infinite-dimensional variables is decomposed a sequence of finite dimensional functions. In this section, we begin with assumptions and basic definitions of the construction of the function spaces, then the functions spaces for finite dimensional integrations and infinite-dimensional integrations.

3.2.1 Assumptions. We follow the approach from [HW02, KSWW10], and μ is the product measure on the space $D^{\mathbb{N}}$ corresponding to the probability measure ρ_1 on $D \subseteq \mathbb{R}$. The construction of spaces of functions with an infinite number of variables $x_1, x_2, \ldots \in D$ is based on a reproducing kernel K_1 for functions of a single variable $x \in D$ and on a family of weights γ_u, which indicate the importance of the variables x_j with $j \in u$ for finite sets $u \subset \mathbb{N}$.

For $\mathbf{x} = (x_j)_{j\in\mathbb{N}} \in D^{\mathbb{N}}$ and $\emptyset \neq u \subset \mathbb{N}$ we put $\mathbf{x}_u = (x_j)_{j\in u} \in D^u$. Unless stated otherwise we use u, v, and w to denote finite subsets of $\mathbb{N}$ in the sequel. For any two positive real numbers y and z, we write $y \preceq z$ if $y \leq cz$ holds with a constant $c > 0$. Furthermore, $y \asymp z$ means $y \preceq z$ and $z \preceq y$.

Let $K_\emptyset = 1$, and so the reproducing kernel Hilbert space $\mathscr{H}(K_\emptyset)$, with the reproducing kernel $K_\emptyset$ is the Hilbert space of constant functions, i.e., $\langle f_\emptyset, g_\emptyset \rangle_{\mathscr{H}(K_\emptyset)} = f_\emptyset g_\emptyset$ for all $f_\emptyset, g_\emptyset \in \mathscr{H}(K_\emptyset)$.

Next, consider a measurable, symmetric, positive semi-definite kernel function

$$K_1 : D \times D \to \mathbb{R}$$

as the building block used to construct the Hilbert space $\mathscr{H}(K)$. By the definition, K_1

satisfies

$$K_1(x,y) = K_1(y,x), \quad \forall x,y \in D,$$
$$\sum_{i,j=1}^{n} b_i K_1(x_i,x_j) b_j \geq 0, \quad \forall \mathbf{b} \in \mathbb{R}^n,\ \forall x_1,\ldots,x_n \in D,\ \forall n \in \mathbb{N}.$$

In addition, it is assumed that the *nominal value* $c \in D$ satisfies

$$K_1(x,c) = 0, \quad \forall x \in D. \tag{3.1a}$$

This means that the K_1 is the reproducing kernel for a Hilbert space, $\mathscr{H}(K_1)$, of functions on D that vanish at c, which implies that $f(c) = 0$, the only constant function in $\mathscr{H}(K_1)$ is the zero function. An example is $D = \mathbb{R}$, $c = 0$ and $K_1(x,y) = \frac{1}{2}|x| + \frac{1}{2}|y| - \frac{1}{2}|x-y|$. Further assumptions are made on the finiteness of K_1 and its integrability with respect to the probability density ρ_1, namely,

$$h_1(x) := \int_{\overline{D}} K_1(x,y)\,\rho_1(y)\,\mathrm{d}y \quad \forall x \in D, \tag{3.1b}$$
$$h_1 \in \mathscr{H}(K_1), \tag{3.1c}$$
$$m := \int_{\overline{D}^2} K_1(x,y)\,\rho_1(x)\rho_1(y)\,\mathrm{d}x\,\mathrm{d}y < \infty, \tag{3.1d}$$
$$M := \int_{\overline{D}} K_1(x,x)\rho_1(x)\,\mathrm{d}x < \infty. \tag{3.1e}$$

For the example of K_1 mentioned above and ρ_1, the standard normal density function, these conditions are satisfied. The distinction between D and $\overline{D}$ above means that the evaluation of $f \in \mathscr{H}(K_1)$ at any point $x \in D$ is a bounded functional, whereas the evaluation of f at $x \in \overline{D} \setminus D$ may be an unbounded functional.

The kernel K_1 is used as the building block here to construct the reproducing kernel for Hilbert spaces of functions of several variables. Let $\mathbf{c} = (c,c,\ldots)$. In the previous examples, $\mathbf{c}$ was chosen to be $\mathbf{0}$. Also, let $1:d$ denote the set $\{1,2,\ldots,d\}$. For any $u \subset \mathbb{N}$, let $|u|$ denote its cardinality. Define $\mathbb{U}$ as the set of subsets of $\mathbb{N}$ with finite cardinality, i.e.,

$$\mathbb{U} = \left\{u \subset \mathbb{N} : |u| < \infty\right\}.$$

Let $\gamma = (\gamma_1, \gamma_2, \ldots)$ be a sequence of non-increasing non-negative weights that satisfy the following summability condition:

$$\sum_{j=d}^{\infty} \gamma_j < \alpha d^{-2q}, \quad d \in \mathbb{N}, \qquad \gamma_1 \geq \gamma_2 \geq \cdots \geq 0, \tag{3.2}$$

for some positive constants α and q, which also implies that

$$\sum_{j=1}^{\infty} \gamma_j < \infty. \tag{3.3}$$

We define

$$\gamma_u = \prod_{j \in u} \gamma_j, \tag{3.4}$$

for every $u \subset \mathbb{U}$.

Under these assumptions the appropriate choice of a domain of functions of infinitely many variables is given by

$$\mathscr{X} = \left\{ \mathbf{x} \in D^{\mathbb{N}} : \sum_{j=1}^{\infty} \gamma_j K_1(x_j, x_j) < \infty \right\}. \tag{3.5}$$

Note that $\mathscr{X} = D^{\mathbb{N}}$ follows from (3.2), if K_1 is a bounded kernel on $D \times D$. In general the complement $D^{\mathbb{N}} \setminus \mathscr{X}$ is a μ-null set, as shown in the following lemma.

Lemma 3.1 *The set $\mathscr{X}$ satisfies $\mu(\mathscr{X}) = 1$.*

Proof 3.1 *By $Y_j(\mathbf{x}) = \gamma_j K_1(x_j, x_j)$ we get a sequence of non-negative random variables on $D^{\mathbb{N}}$. Clearly,*

$$\mathbb{E}\left(\sum_{j=1}^{\infty} Y_j \right) = \sum_{j=1}^{\infty} \mathbb{E}(Y_j) = \sum_{j=1}^{\infty} \gamma_j \int_D K_1(x,x)\, \rho_1(x)\, \mathrm{d}x < \infty.$$

Hence $\sum_{j=1}^{\infty} Y_j < \infty$ μ-almost surely.

We add that without condition (3.1e) we always have $\mu(\mathscr{X}) \in \{0, 1\}$, which follows from Kolmogorov's Zero-One Law. We stress that $\mathscr{X}$ contains every $\mathbf{x} \in D^{\mathbb{N}}$ that is constant outside of some finite subset of $\mathbb{N}$.

3.2.2 Functions of finitely many variables. We intend to define the space for functions with infinitely many variables. First, we construct spaces of functions $f : \mathscr{X} \to \mathbb{R}$ that only depend on a finite number of variables. For $u \neq \emptyset$ we consider the reproducing kernel

$$k_u(\mathbf{x}, \mathbf{y}) = \prod_{j \in u} K_1(x_j, y_j), \qquad \mathbf{x}, \mathbf{y} \in \mathscr{X},$$

as well as the associated Hilbert space

$$\mathscr{H}_u = \mathscr{H}(k_u).$$

Furthermore, we put $k_\emptyset = 1$ and

$$\mathscr{H}_\emptyset = \mathscr{H}(1).$$

The following lemma appears in [HW02, Sec. 2] for the case of a bounded kernel K_1 and $D = [0,1]$. The same argument can be easily extended to the more general case considered here.

Lemma 3.2 *For $\mathbf{x}, \mathbf{y} \in \mathscr{X}$ and $f \in \mathscr{H}_u$ we have*

$$\mathbf{x}_u = \mathbf{y}_u \quad \Rightarrow \quad f(\mathbf{x}) = f(\mathbf{y}).$$

Lemma 3.3 *Suppose that $f_i \in \mathscr{H}_{u_i}$ with pairwise different finite sets $u_1, \ldots, u_n \subset \mathbb{N}$. Then $\sum_{i=1}^n f_i = 0$ implies $f_i = 0$ for every i.*

Proof 3.2 *Inductively we conclude as follows. Let $n \geq 2$, put $u = \bigcup_{i=1}^n u_i$, and fix $\ell \in u \setminus \bigcap_{i=1}^n u_i$. Choose $a_j \in D$ for $j \in u \setminus \{\ell\}$ and $c \in D$, and consider the functions $g_i : D \to \mathbb{R}$ that are given by*

$$g_i(y) = f_i(\mathbf{x})$$

with $\mathbf{x} \in \mathscr{X}$ defined by

$$x_j = \begin{cases} y, & \text{if } j = \ell, \\ c_j, & \text{if } j \in u \setminus \{\ell\}, \\ c, & \text{otherwise.} \end{cases}$$

Put $I_1 = \{i : \ell \in u_i\}$ and $I_2 = \{i : \ell \notin u_i\}$, and note that $I_1 \neq \emptyset$ and $I_2 \neq \emptyset$. Suppose that $i \in I_1$. We apply Lemma A.1 and Lemma A.2 from the Appendix with $E = \mathcal{X}$, $E_1 = D^{\{\ell\}}$, $E_2 = \{\mathbf{x} \in D^{\mathbb{N}\setminus\{\ell\}} : \sum_{j\neq\ell} \gamma_j K_1(x_j, x_j) < \infty\}$, and

$$J(\mathbf{x}, \mathbf{y}) = L(x_\ell, y_\ell) = \alpha_1 K_1(x_\ell, y_\ell),$$

where

$$\alpha_1 = \prod_{j \in u \setminus \{\ell\}} K_1(c_j, c_j),$$

to conclude that $g_i \in \mathcal{H}(K_1)$. Otherwise, i.e., if $i \in I_2$, then $f_i \in \mathcal{H}_{u_i}$ together with Lemma 3.2 implies that g_i is constant. By assumption $\sum_{i\in I_1} g_i = -\sum_{i\in I_2} g_i$, so that both sums vanish according to (3.1a)*. Since the values of x_j with $j \in u \setminus \{\ell\}$ have been chosen arbitrarily, we obtain $\sum_{i\in I_1} f_i = 0$ and $\sum_{i\in I_2} f_i = 0$ from Lemma 3.2.*

To introduce the kernel associated with the function of finitely many variables, we use the weight defined in (3.4) to consider the weighted sum

$$K_v(\mathbf{x}, \mathbf{y}) = \sum_{u \subseteq v} \gamma_u k_u(\mathbf{x}, \mathbf{y}), \qquad \mathbf{x}, \mathbf{y} \in \mathcal{X},$$

of reproducing kernels k_u. Clearly K_v is a reproducing kernel, too, and due to Lemma 3.3 the corresponding Hilbert space satisfies

$$\mathcal{H}(K_v) = \bigoplus_{u \subseteq v} \mathcal{H}(\gamma_u k_u)$$

with pairwise orthogonal spaces $\mathcal{H}(\gamma_u k_u)$. See [HW02, Lemma 3] for this fact and also for the following conclusion in the case of a bounded kernel K_1 and $D = [0, 1]$.

Lemma 3.4 *The space $\mathcal{H}(K_v)$ consists of all functions*

$$f = \sum_{u \subseteq v} f_u, \qquad f_u \in \mathcal{H}_u.$$

Furthermore,

$$\|f\|_{K_v}^2 = \sum_{u \subseteq v} \gamma_u^{-1} \|f_u\|_{\mathcal{H}(k_u)}^2.$$

Remark 3.2 *Due to Lemma 3.2 and Lemma 3.4, every function $f \in \mathscr{H}(K_v)$ may be identified with a function on $D^{|v|}$, and K_v may be identified with a kernel on $D^{|v|} \times D^{|v|}$ as well. For consistency we prefer to work with the domain $\mathscr{X}$ throughout this thesis.*

Remark 3.3 *By definition* (3.2) *of the weights γ_u we have*

$$K_v(\mathbf{x},\mathbf{y}) = \prod_{j\in v} \big(1+\gamma_j K_1(x_j,x_j)\big)\,.$$

Hence K_v is of tensor product form, and $\mathscr{H}(K_v)$ is the tensor product space

$$\mathscr{H}(K_v) = \bigotimes_{j\in v} \mathscr{H}(1+\gamma_j K_1),$$

considered as a space of functions on $D^{|v|}$.

3.2.3 Function of infinitely many variables. For $d \in \mathbb{N}$, let $1:d$ denote the set $\{1,\dots,d\}$. We consider the limit of the sequence of kernels $K_{1:d}$.

Lemma 3.5 *For $\mathbf{x},\mathbf{y} \in \mathscr{X}$ we have*

$$\sum_u \gamma_u\, |k_u(\mathbf{x},\mathbf{y})| < \infty.$$

Proof 3.3 *Note that*

$$\begin{aligned}\sum_u \gamma_u\, |k_u(\mathbf{x},\mathbf{y})| &= \sum_u \prod_{j\in u} \gamma_j\, |K_1(x_j,y_j)| \\ &\le \left(\sum_u \prod_{j\in u} \gamma_j K_1(x_j,x_j)\right)^{1/2} \left(\sum_u \prod_{j\in u} \gamma_j K_1(y_j,y_j)\right)^{1/2}\end{aligned}$$

due to (3.2)*. Furthermore,*

$$\sum_u \prod_{j\in u} \gamma_j K_1(x_j,x_j) = \prod_{j=1}^{\infty} (1+\gamma_j K_1(x_j,x_j)) \le \exp\left(\sum_{j=1}^{\infty} \gamma_j K_1(x_j,x_j)\right) < \infty$$

by definition of $\mathscr{X}$.

Due to Lemma 3.5 the limit

$$K(\mathbf{x},\mathbf{y}) = \sum_u \gamma_u k_u(\mathbf{x},\mathbf{y}) = \sum_u \gamma_u \prod_{j\in u} K_1(x_j,y_j) = \prod_{j=1}^{\infty}[1+\gamma_j K_1(x_j,y_j)], \qquad \mathbf{x},\mathbf{y}\in\mathscr{X}, \tag{3.6}$$

of the sequence of kernels $K_{1:d}$ defines a measurable kernel K on $\mathscr{X}\times\mathscr{X}$.

If $d<d'$ then $\mathscr{H}(K_{1:d})\subseteq\mathscr{H}(K_{1:d'})\subseteq\mathscr{H}(K)$, and $\bigcup_{d=1}^{\infty}\mathscr{H}(K_{1:d})$ is a dense linear subspace of $\mathscr{H}(K)$. More precisely, the following holds true by the argument in [HW02, Cor. 5] for the case of a bounded kernel K_1 and $D=[0,1]$.

Lemma 3.6 *The Hilbert space $\mathscr{H}(K)$ consists of all functions*

$$f = \sum_{u\in\mathbb{U}} f_u \tag{3.7}$$

with $f_u\in\mathscr{H}(k_u)$ such that

$$\sum_{u\in\mathbb{U}} \gamma_u^{-1}\|f_u\|^2_{\mathscr{H}(k_u)} < \infty.$$

Moreover, $\langle f,g\rangle_{\mathscr{H}(K)} = \sum_{u\in\mathbb{U}}\gamma_u^{-1}\langle f_u,g_u\rangle_{\mathscr{H}(k_u)}$ for $f,g\in\mathscr{H}(K)$.

Remark 3.4 *According to Lemma 3.6, $(\mathscr{H}(k_u))_{u\in\mathbb{U}}$ is a family of closed and pairwise orthogonal subspaces of $\mathscr{H}(K)$ and f_u is the orthogonal projection of f onto $\mathscr{H}(k_u)$. Roughly speaking, f_u represents the joint effect of the variables x_j with $j\in u$ on the function f.*

Following the argument in [HW02], it can be shown that $\mathscr{H}(K_u)\bigcap\mathscr{H}(K_v)=\{0\}$ for any $u\neq v$ because $\mathscr{H}(1)\bigcap\mathscr{H}(K_1)=\{0\}$. An outline of this argument is as follows. Let $f\in\mathscr{H}(K_u)\bigcap\mathscr{H}(K_v)$, and without loss of generality, assume that there exists a $j\in u\setminus v$. Since $f\in\mathscr{H}(K_u)$, it follows that $f(\mathbf{x})=0$ for all $\mathbf{x}\in D^{|u\cup v|}$ with $x_j=c$. On the other hand, since $f\in\mathscr{H}(K_v)$, it follows that f does not depend on the value of x_j, so $f(\mathbf{x})=0$ for all $\mathbf{x}\in D^{|u\cup v|}$. Because $\mathscr{H}(K_u)$ and $\mathscr{H}(K_v)$ have only the zero function in common for $u\neq v$, they are orthogonal subspaces of $\mathscr{H}(K)$, i.e.,

$$\mathscr{H}(K) = \bigoplus_{u\in\mathbb{U}}\mathscr{H}(K_u), \qquad \text{where} \quad \mathscr{H}(K_u)\perp\mathscr{H}(K_v), \quad \forall u\neq v.$$

To compute $I(f) = \mathbb{E}(f(X_1, X_2, \ldots))$ all but finitely many random variables X_j are replaced by the nominal value c. Adopting the truncation operator defined in (2.18), we denote

$$(\Psi_v f)(\mathbf{x}) = f(\mathbf{x}_v, \mathbf{c})$$

for $\mathbf{x} \in \mathscr{X}$ and $v \in \mathbb{U}$, where $(\mathbf{x}_v, \mathbf{c})$ is used to denote the sequence $\mathbf{y} \in \mathscr{X}$ with $y_j = x_j$ for $j \in v$ and $y_j = c$ otherwise. For $f = \sum_{u \in \mathbb{U}} f_u$ according to Lemma 3.6 we get $f_u(\mathbf{x}_v, \mathbf{c}) = 0$ if $u \not\subseteq v$ because of (3.1a), and therefore

$$\Psi_v f = \sum_{u \subseteq v} f_u.$$

We conclude that $\Psi_v f$ is the orthogonal projection of $f \in \mathscr{H}(K)$ into the Hilbert subspace $\mathscr{H}(K_v)$ with reproducing kernel

$$K_v(\mathbf{x}, \mathbf{y}) = \sum_{u \subseteq v} \gamma_u k_u(\mathbf{x}, \mathbf{y}) = \prod_{j \in v} \left[1 + \gamma_j K_1(x_j, y_j)\right].$$

We add that $f \in \mathscr{H}(K_v)$ may be identified with a function $f : D^{|v|} \to \mathbb{R}$ of the variables x_j with $j \in v$ only.

3.2.4 Integration with respect to the product measure μ. The infinite-dimensional kernel function K defined in (3.6) is not necessarily finite for all $\mathbf{x}, \mathbf{y} \in D^{\mathbb{N}}$, particularly for unbounded K_1, and for that reason, the kernel K is not necessarily a reproducing kernel. Likewise, $\mathscr{H}(K)$ is not necessarily a reproducing kernel Hilbert space, since function evaluation may not be a bounded functional at every point in $D^{\mathbb{N}}$. However, this is true for any function f with the domain $\mathscr{X}$.

Beside those points in $\mathscr{X}$, the function evaluation could be bounded for certain points $\mathbf{y}$. Specifically, consider points of the form $\mathbf{y} = (\mathbf{y}_u, \mathbf{c}) \in D^{\mathbb{N}}$ where $u \in \mathbb{U}$, i.e., $y_j = c$ for $j \notin u$. By the definition of the Hilbert spaces $\mathscr{H}(K_v)$ it follows that $f_v(\mathbf{y}_u, \mathbf{c})$ vanishes

for $v \not\subseteq u$, and so $f(\mathbf{y}_u, \mathbf{c}) = \sum_{v \subseteq u} f_v(\mathbf{y}_v)$. Since this sum is finite, one may write

$$\begin{aligned} f(\mathbf{y}_u, \mathbf{c}) &= \sum_{v \subseteq u} f_v(\mathbf{y}_v) = \sum_{v \subseteq u} \langle K_v(\cdot, \mathbf{y}_v), f_v \rangle_{\mathscr{H}(K_v)} \\ &= \sum_{v \subseteq u} \langle K_v(\cdot, \mathbf{y}_v), f_v \rangle_{\mathscr{H}(K_v)} + \sum_{\substack{v \in \mathbb{U} \\ v \not\subseteq u}} \langle 0, f_v \rangle_{\mathscr{H}(K_v)} \\ &= \left\langle \sum_{v \subseteq u} K_v(\cdot, \mathbf{y}_v), \sum_{v \in \mathbb{U}} f_v \right\rangle_{\mathscr{H}(K)} = \left\langle \sum_{v \subseteq u} \prod_{j \in v} \gamma_j K_1(\cdot, y_j), f \right\rangle_{\mathscr{H}(K)} \\ &= \left\langle \prod_{j \in u} [1 + \gamma_j K_1(\cdot, y_j)], f \right\rangle_{\mathscr{H}(K)} = \left\langle K(\cdot, (\mathbf{y}_u, \mathbf{c})), f \right\rangle_{\mathscr{H}(K)}. \end{aligned} \tag{3.8}$$

One can claim that $|f(\mathbf{y}_u, \mathbf{c})| < \infty$ since

$$|f(\mathbf{y}_u, \mathbf{c})| = \left| \left\langle K(\cdot, (\mathbf{y}_u, \mathbf{c})), f \right\rangle_{\mathscr{H}(K)} \right| \leq \|K(\cdot, (\mathbf{y}_u, \mathbf{c}))\|_{\mathscr{H}(K)} \|f\|_{\mathscr{H}(K)}.$$

In addition, (3.2) implies that

$$\begin{aligned} \|K(\cdot, (\mathbf{y}_u, \mathbf{c}))\|^2_{\mathscr{H}(K)} &= K\big((\mathbf{y}_u, \mathbf{c}), (\mathbf{y}_u, \mathbf{c})\big) = \prod_{j \in u} \big[1 + \gamma_j K_1(y_j, y_j)\big] \\ &\leq \prod_{j \in u} \big[1 + \gamma_j L\big] = \prod_{j \in u} \exp\big(\log(1 + \gamma_j L)\big) \leq \exp\Big(L \sum_{j \in u} \gamma_j\Big) < \infty, \end{aligned} \tag{3.9}$$

where $L = \max_{j \in u} K_1(y_j, y_j)$. In fact, even if for points of the from $\mathbf{y} = (\mathbf{y}_u, \mathbf{a})$, where $\mathbf{a}$ may not be equal to the nominal value vector $\mathbf{c}$, the above argument still holds. This is because $\|K(\cdot, (\mathbf{y}_u, \mathbf{c}))\|^2_{\mathscr{H}(K)} = \prod_{j \in u} \big[1 + \gamma_j L\big] + \prod_{j \notin u} [1 + \gamma_j K_1(a, a)]$, the first term in this equation is shown to be finite by (3.9), and the second term's finiteness is due to the decaying property of the weights γ_j and the finiteness of $K_1(a, a)$ for $a \in D$.

Thus, $K(\cdot, (\mathbf{y}_u, \mathbf{c}))$ is the representer for function evaluation at $(\mathbf{y}_u, \mathbf{c})$. From the formula for $f(\mathbf{y}_u, \mathbf{c})$ in (3.8), one can recursively write the effects or pieces of f in terms of the anchor $\mathbf{c}$ as in [HW02]:

$$f_\emptyset = f(\mathbf{c}), \quad f_u(\mathbf{y}_u) = f(\mathbf{y}_u, \mathbf{c}) - \sum_{v \subset u} f_v(\mathbf{y}_v), \quad u \in \mathbb{U}.$$

From the argument above, the representer for function evaluation at $(\mathbf{y}_{1:d}, \mathbf{c}_{d+1:\infty})$

is $K^{(d)}(\cdot, \mathbf{y}_{1:d})$, where $K^{(d)}$ is defined by

$$\begin{aligned} K^{(d)}(\mathbf{x}_{1:d}, \mathbf{y}_{1:d}) &= \sum_{u \subseteq 1:d} K_u(\mathbf{x}_u, \mathbf{y}_u) = \prod_{j=1}^{d} [1 + \gamma_j K_1(x_j, y_j)] \\ &= K\left((\mathbf{x}_{1:d}, \mathbf{c}), (\mathbf{y}_{1:d}, \mathbf{c})\right). \end{aligned}$$

The orthogonality of the subspaces $\mathscr{H}(K_u)$ and $\mathscr{H}(K_v)$ for $u \neq v$ implies that

$$f_1(\cdot, \mathbf{c}_{d+1:\infty}) \perp f_2 - f_2(\cdot, \mathbf{c}_{d+1:\infty}), \quad \forall f_1, f_2 \in \mathscr{H}(K), \tag{3.10}$$

a fact used later to split the error of approximating the integral into two parts. Note that the Monte Carlo type estimator for the expectation, $\hat{I}$, based on $\{\mathbf{x}_{i,1:d}\}_{i=1}^{n} \subset I^d$ can be represented as an inner product in $\mathscr{H}(K)$:

$$\begin{aligned} \hat{I} = \frac{1}{n} \sum_{i=1}^{n} f(\mathbf{x}_{i,1:d}, \mathbf{c}) &= \left\langle \frac{1}{n} \sum_{i=1}^{n} K^{(d)}(\cdot, \mathbf{x}_{i,1:d}), f \right\rangle_{\mathscr{H}(K)} \\ &= \left\langle \frac{1}{n} \sum_{i=1}^{n} K\left(\cdot, (\mathbf{x}_{i,1:d}, \mathbf{c})\right), f \right\rangle_{\mathscr{H}(K)}. \end{aligned} \tag{3.11}$$

Having defined the Hilbert space of functions defined in $D^{\mathbb{N}}$, it is now possible to define the expectation of such functions, f. This expectation is constructed in terms of the expectations of pieces of f. For any $u \in \mathbb{U}$, $\mathbb{E}[f_u(\mathbf{X}_u)]$ is a $|u|$-dimensional integral, which may represented as

$$\begin{aligned} \mathbb{E}[f_u(\mathbf{X}_u)] &= \int_{\overline{D}^u} f_u(\mathbf{x}_u) \prod_{j \in u} \rho_1(x_j) \, \mathrm{d}\mathbf{x}_u = \langle h_u, f_u \rangle_{\mathscr{H}(K_u)} \\ &= \langle h_u, f \rangle_{\mathscr{H}(K)}, \quad \forall f_u \in \mathscr{H}(K_u),\ f \in \mathscr{H}(K), \\ h_u(\mathbf{x}_u) &= \int_{\overline{D}^{|u|}} K_u(\mathbf{x}_u, \mathbf{y}_u) \left(\prod_{j \in u} \rho_1(y_j) \right) \mathrm{d}\mathbf{y}_u \\ &= \int_{\overline{D}^{|u|}} \left(\prod_{j \in u} \gamma_j K_1(x_j, y_j) \rho_1(y_j) \right) \mathrm{d}\mathbf{y}_u = \prod_{j \in u} \gamma_j h_1(x_j) \in \mathscr{H}(K_u). \end{aligned}$$

The formula for h_u follows from the reproducing property of the kernel K_u, and the fact that $h_u \in \mathscr{H}(K_u)$ follows from assumption (3.1c) on h_1. The d-dimensional approximate

expectation I_d can be represented as

$$\begin{aligned} I_d &= \mathbb{E}[f(X_1,\ldots,X_d,\mathbf{c})] = \mathbb{E}[f(\mathbf{X})|X_{d+1:\infty}=\mathbf{c}] \\ &= \sum_{u\subseteq 1:d} \mathbb{E}[f_u(\mathbf{X}_u)] = \sum_{u\subseteq 1:d} I_u = \sum_{u\subseteq 1:d} \langle h_u, f\rangle_{\mathscr{H}(K)} \\ &= \left\langle \sum_{u\subseteq 1:d} h_u, f \right\rangle_{\mathscr{H}(K)} = \left\langle h^{(d)}, f \right\rangle_{\mathscr{H}(K)}, \\ h^{(d)}(\mathbf{x}_{1:d}) &= \sum_{u\subseteq 1:d} h_u(\mathbf{x}_u) = \prod_{j=1}^{d} \left[1+\gamma_j h_1(x_j)\right], \\ \|h^{(d)}\|^2_{\mathscr{H}(K)} &= \sum_{u\subseteq 1:d} \|h_u\|^2_{\mathscr{H}(K)} = \prod_{j=1}^{d}[1+\gamma_j m] < \infty \end{aligned}$$

by (3.2). The finiteness of $\|h^{(d)}\|^2_{\mathscr{H}(K)}$ is justified by the same argument as in (3.9).

The expectation of the whole function is defined as the countable sum of the expectations of the parts, namely,

$$I = \mathbb{E}[f(X_1,X_2,\ldots)] := \sum_{u\in\mathbb{U}} \langle h_u, f\rangle_{\mathscr{H}(K)} = \langle h, f\rangle_{\mathscr{H}(K)}, \tag{3.12}$$

$$\begin{aligned} h(\mathbf{x}) &:= \sum_{u\in\mathbb{U}} h_u(\mathbf{x}_u) = \sum_{u\in\mathbb{U}} \prod_{j\in u} \gamma_j h_1(x_j) = \prod_{j=1}^{\infty} \left[1+\gamma_j h_1(x_j)\right], \\ \|h\|^2_{\mathscr{H}(K)} &= \sum_{u\in\mathbb{U}} \|h_u\|^2_{\mathscr{H}(K)} = \prod_{j=1}^{\infty}[1+\gamma_j m] \le e^{\alpha m} < \infty, \end{aligned} \tag{3.13}$$

where α is defined in (3.2). Another way to represent h is

$$h(\mathbf{x}) = \langle h, K(\cdot,\mathbf{x})\rangle_{\mathscr{H}(K)} = \int_{\mathscr{X}} K(\mathbf{x},\mathbf{y})\,\mu(\mathrm{d}\mathbf{y}), \qquad \mathbf{x}\in\mathscr{X}. \tag{3.14}$$

By definition and the summability condition on γ in (3.2), it follows that $h\in\mathscr{H}(K)$, and so it represents a bounded linear functional on $\mathscr{H}(K)$, namely, the expectation. Note that $h^{(d)}(\mathbf{x}_{1:d}) = h(\mathbf{x}_{1:d},\mathbf{c}_{d+1:\infty})$, so it follows that

$$I_d = \left\langle h^{(d)}, f \right\rangle_{\mathscr{H}(K)} = \left\langle h(\cdot,\mathbf{c}_{d+1:\infty}), f \right\rangle_{\mathscr{H}(K)}. \tag{3.15}$$

It can be shown by Theorem 4.1 in Section 4.2 that $\lim_{d\to\infty} I_d = I$, since the worst-case bias in (4.5) vanishes as $d\to\infty$.

Note that $\hat{I}, I$ and I_d defined in (3.11), (3.12) and (3.15) together all depend on f. In the discussion that follows, this f dependence is sometimes written explicitly.

3.3 Examples

We provide two examples with ρ_1 being the uniform distribution on $D = [0,1]$ and $\mathscr{X} = D^{\mathbb{N}}$ in both cases. Let $W_2^1([0,1])$ consist of all absolutely continuous functions $f : [0,1] \to \mathbb{R}$ with square-integrable derivatives, and let the norm on $W_2^1([0,1])$ be given by

$$\|f\|^2 = \left(\int_0^1 f(y)\,\mathrm{d}y\right)^2 + \gamma^{-1}\int_0^1 (f')^2(y)\,\mathrm{d}y$$

for some $\gamma > 0$. Then we have

$$W_2^1([0,1]) = \mathscr{H}(1+\gamma K_1),$$

where

$$K_1(x,y) = 1/3 + (x^2+y^2)/2 - \max(x,y), \qquad x,y \in [0,1]. \quad (3.16)$$

The covariance kernel K_1 clearly satisfies the basic properties of the one-dimensional kernel function, and (3.1a) holds, too, since

$$\mathscr{H}(K) = \left\{ f \in W_2^1([0,1]) : \int_0^1 f(y)\,\mathrm{d}y = 0 \right\}.$$

For $u \neq \emptyset$ the space $\mathscr{H}_u$ consists of all continuous functions f such that $f(\mathbf{x})$ depends only on $\mathbf{x}_u$, $f^{(u)} \in L_2([0,1]^u)$ for the weak derivative $f^{(u)} = \frac{\partial^{|u|}}{\partial \mathbf{x}_u} f$, and $\int_0^1 f(\mathbf{y})\,\mathrm{d}y_j = 0$ for every $j \in u$. Furthermore,

$$\|f\|_{K_u}^2 = \int_{[0,1]^u} \left(f^{(u)}(\mathbf{y})\right)^2 \mathrm{d}\mathbf{y}. \quad (3.17)$$

It follows that $\mathscr{H}(K_{1:d}) \subseteq G_d$ for $d \in \mathbb{N}$, where G_d denotes the class of continuous functions f such that $f(\mathbf{x})$ depends only on $\mathbf{x}_{1:d}$ and f has square-integrable weak derivatives $f^{(u)}$ for every $u \subseteq 1:d$.

Let I_v denote integration with respect to the variables y_j with $j \in v$, i.e., $I_v f : D^{\mathbb{N}\setminus v} \to \mathbb{R}$ for an integrable function $f : D^{\mathbb{N}} \to \mathbb{R}$. Suppose that $f = \sum_{u\subseteq 1:d} f_u \in \mathscr{H}(K_{1:d})$ according to Lemma 3.4. Since

$$I_{1:d\setminus v}(f) = \sum_{u\subseteq v} I_{1:d\setminus v}(f_u) = \sum_{u\subseteq v} f_u,$$

we can recursively determine the components f_u of f. In fact,

$$f_\emptyset = I_{1:d}(f) \tag{3.18}$$

and, for $v \neq \emptyset$,

$$f_v = I_{1:d\setminus v}(f) - \sum_{u\subsetneq v} f_u. \tag{3.19}$$

Conversely, suppose that $f \in G_d$, and define f_v for $v \subseteq 1:d$ by means of this recursion. We get $f_v \in \mathscr{H}_v$ with

$$f_v^{(v)} = \Big(I_{1:d\setminus v}(f)\Big)^{(v)} = I_{1:d\setminus v}(f^{(v)}).$$

We conclude that $\mathscr{H}(K_{1:d}) = G_d$ is a weighted Sobolev-Hilbert space with the norm given by

$$\|f\|_{\mathscr{H}(K_{1:d})} = \sum_{u\subseteq 1:d} \gamma_u^{-1} \int_{[0,1]^u} \left(\int_{[0,1]^{1:d\setminus u}} f^{(u)}(\mathbf{x})\, \mathrm{d}\mathbf{x}_{1:d\setminus u} \right)^2 \mathrm{d}\mathbf{x}_u.$$

See [YH05, Sec. 3]. Observe that $\sum_{u\subseteq 1:d} f_u$ is the ANOVA decomposition of $f \in \mathscr{H}(K_{1:d})$, so that $\mathscr{H}(K_{1:d})$ is defined by imposing a smoothness assumption on the ANOVA terms f_u, namely existence and square integrability of the weak derivatives $f_u^{(u)}$. Moreover, $\|f\|^2_{\mathscr{H}(K_{1:d})}$ is a weighted average of the squared L_2-norms of these weak derivatives. See [NW08, Sec. 5.3.1].

Note that the recursion (3.18) and (3.19) is valid, too, for $f \in \mathscr{H}(K)$ if $1:d$ is replaced by $\mathbb{N}$. Moreover, it extends to the case of any kernel K_1 with the property (3.1a), if we replace integration with respect to a single variable by the functional $f \mapsto \langle f, 1\rangle_{\mathscr{H}(1+K_1)}$, which is then applied to all variables y_j with $j \in \mathbb{N}\setminus v$.

As a second example consider the covariance kernel

$$K_1(x,y) = \min(x,y), \qquad x,y \in [0,1], \tag{3.20}$$

of a Brownian motion, which can be treated analogously to the kernel given by (3.16), if integration of a function $f : [0,1] \to \mathbb{R}$ is replaced by evaluation of f at the point zero. In particular, K_1 satisfies (3.1a), and for $u \neq \emptyset$ the corresponding space $\mathscr{H}_u$ consists of all continuous functions $f : D^{\mathbb{N}} \to \mathbb{R}$ such that $f(\mathbf{x})$ depends only on $\mathbf{x}_u$, $f^{(u)} \in L_2([0,1]^u)$, and $f(\mathbf{x}) = 0$ if $\mathbf{x}_j = 0$ for some $j \in u$. Moreover, $\|f\|^2_{\mathscr{H}(K_u)}$ is given by (3.17).

To illustrate the role of the weights in case of (3.16) as well as in the case of (3.20) we consider a sequence of real numbers $(\eta_j)_{j\in\mathbb{N}}$ such that $\sum_{j=1}^{\infty} |\eta_j| < \infty$, and we define

$$f(\mathbf{x}) = \sum_{j=1}^{\infty} \eta_j x_j^2, \qquad \mathbf{x} \in D^{\mathbb{N}}.$$

Then

$$f = f_\emptyset + \sum_{j=1}^{\infty} f_{\{j\}} = \sum_{j=1}^{\infty} g_{\{j\}},$$

with $f_\emptyset = 1/3 \sum_{j=1}^{\infty} \eta_j$ and $f_{\{j\}}(\mathbf{x}) = \eta_j (x_j^2 - 1/3)$ as well as $g_{\{j\}}(\mathbf{x}) = \eta_j x_j^2$. In the case of the kernel given by (3.16) we have $f_{\{j\}} \in \mathscr{H}_{\{j\}}$ and

$$\|f_\emptyset\|_{K_\emptyset} = 1/3 \left| \sum_{j=k}^{\infty} \eta_j \right|, \qquad \|f_{\{j\}}\|^2_{K_{\{j\}}} = 4/3\, \eta_j^2.$$

If K is given by (3.20) then $g_{\{j\}} \in \mathscr{H}_{\{j\}}$ and

$$\|g_{\{j\}}\|^2_{\mathscr{H}(k_{\{j\}})} = 4/3\, \eta_j^2.$$

Thus $f \in \mathscr{H}(K)$ iff

$$\sum_{j=1}^{\infty} \frac{\eta_j^2}{\gamma_j} < \infty$$

in both cases. For instance, if $\gamma_j \asymp j^{-(1+\delta)}$ with any $\delta > 0$ then it suffices to have $\eta_j \asymp j^{-\alpha}$ with $\alpha > 1 + \delta/2$.

CHAPTER 4

RESULTS IN THE WORST-CASE SETTING

In this chapter, we present two cost models, the *fixed subspace sampling* corresponding to the single-level algorithm and the *variable subspace sampling* associated with the multi-level algorithm, both of which have been introduced in [CDMGR09]. Based upon those models, we define minimal errors in the worst-case setting. Our goal is to study the asymptotic behavior of the minimal error. This chapter begins with a brief description of the cost models and errors in the worst-case setting. Then the error analysis for the infinite-dimensional quadrature problems is presented in Theorem 4.1 and Theorem 4.4. We also compare our work with [KSWW10], whose results are stated in Theorem 4.6. This chapter ends with an application of the theoretical result.

4.1 Cost Models and Minimal Errors

4.1.1 The cost models. Throughout this thesis we assume that algorithms for approximation of $I(f)$ have access to the function f via an oracle (subroutine) that provides values $f(\mathbf{x})$ for points $\mathbf{x} \in \mathbb{R}^{\mathbb{N}}$ or a subset thereof. For convenience we define $f(\mathbf{x}) = 0$ for $\mathbf{x} \in \mathbb{R}^{\mathbb{N}} \setminus \mathscr{X}$, so that the integrands f are defined on the whole space $\mathbb{R}^{\mathbb{N}}$.

We study two cost models for the infinite-dimensional quadrature problem. Let $1:d = \{1,\dots,d\}$ and

$$\mathscr{X}_{1:d} = \{\mathbf{x} \in D^{\mathbb{N}} : x_{d+1} = x_{d+2} = \cdots = c\}.$$

This means that $\mathscr{X}_{1:d}$ is the set of all points whose coordinates assumed to be the nominal value with dimensional d_{l+1} and larger. Clearly $\mathscr{X}_{1:d} \subseteq \mathscr{X}$, and $\mathscr{X}_{1:d}$ may be considered as an d-dimensional affine subspace of $\mathbb{R}^{\mathbb{N}}$. As the basic assumption in both models, for every dimension d an oracle is available that provides values of f at any knot $\mathbf{x} \in \mathscr{X}_{1:d}$, and the cost of a single function evaluation by means of this oracle is given by d^s with some fixed parameter $s > 0$. In the fixed subspace model every quadrature formula $\hat{I}$ uses

a single oracle for some dimension d to provide all function values $f(\mathbf{x}_i)$. Accordingly, the cost $C_s^{\mathrm{fix}}(\hat{I})$ of $\hat{I}$ is defined by

$$C_s^{\mathrm{fix}}(\hat{I}) = n \cdot \left(\min\{d \in \mathbb{N} : \{\mathbf{x}_1, \ldots, \mathbf{x}_n\} \subseteq \mathscr{X}_{1:d}\}\right)^s$$

if $\mathbf{x}_1, \ldots, \mathbf{x}_n$ are the pairwise different knots of $\hat{I}$. In the variable subspace model a quadrature formula may use oracles for different dimensions. This leads to the definition of the cost

$$C_s^{\mathrm{var}}(\hat{I}) = \sum_{i=1}^{n} \left(\min\{d \in \mathbb{N} : \mathbf{x}_i \in \mathscr{X}_{1:d}\}\right)^s$$

of $\hat{I}$ in the variable subspace sampling model. Similar cost models in the randomized setting are introduced in Section 5.1.

4.1.2 Defining the error. Given the assumptions on the function space $\mathscr{H}(K)$ in Chapter 3, $f \mapsto I(f)$ defines a bounded linear functional on $\mathscr{H}(K)$, this functional corresponds to the expectation integral, see [HMGNR10, Sec. 2.5] and [NH10] . We approximate $I(f)$ by means of quadrature formulas

$$\hat{I}(f) = \sum_{i=1}^{n} \frac{1}{n} f(\mathbf{x}_i) \tag{4.1}$$

with knots $\mathbf{x}_i \in \mathscr{X}$, whose components coincide with the nominal value c for all but finitely many coordinates. The worst-case error of $\hat{I}$ on function classes F, is defined as

$$e_{\mathrm{wor}}(\hat{I}, F) = \sup_{f \in F} |I(f) - \hat{I}(f)|,$$

where we are primarily interested in the case of the unit ball in $\mathscr{H}(K)$:

$$F = B(K) = \{f \in \mathscr{H}(K) : \|f\|_{\mathscr{H}(K)} \leq 1\}.$$

For a cost budget $N \in \mathbb{N}$ and $\dagger \in \{\mathrm{fix}, \mathrm{var}\}$ the minimal errors in the corresponding cost models are defined by the infinitum over all algorithms with cost no greater than N:

$$e_s^{\dagger}(N, F) = \inf\{e_{\mathrm{wor}}(\hat{I}, F) : C_s^{\dagger}(\hat{I}) \leq N\}.$$

Clearly $C_s^{\mathrm{var}}(\hat{I}) \leq C_s^{\mathrm{fix}}(\hat{I})$, and therefore

$$e_s^{\mathrm{var}}(N,F) \leq e_s^{\mathrm{fix}}(N,F).$$

The asymptotic behavior of the minimal errors $e_s^{\mathrm{fix}}(N,F)$ and $e^{\mathrm{var}}(N,F)$ are studied and compared to results from [KSWW10, Gne10]. Furthermore, we study the construction of quadrature formulas with cost bounded by N and error close to the corresponding minimal error. In order to simplify the presentation we introduce the exponents

$$\lambda_s^{\dagger}(F) = \sup\{\chi > 0 : \sup_{N\in\mathbb{N}} e_s^{\dagger}(N,F)\cdot N^{\chi} < \infty\} \tag{4.2}$$

for $\dagger \in \{\mathrm{fix}, \mathrm{var}\}$. Note that both $e_s^{\dagger}(N,F)$ and $C_s^{\dagger}(\hat{I})$ correspond to the worst-case setting, we drop the notation *wor* for simplicity.

4.2 The Error Analysis of the Single-Level Algorithm

Continuing from the discussion in Section 3.2.4, where we represent the infinite-dimensional integration I, the $d-$dimensional integration I_d and the estimator $\hat{I}$ with inner products in the reproducing kernel Hilbert space $\mathscr{H}(K)$, we are able to conclude the following theorem [NH10]:

Theorem 4.1 *Suppose that K_1 is a symmetric, real-valued, positive semi-definite kernel function defined on D^2 that satisfies the assumptions* (3.1) *and* (3.2)*. Here D may be a finite semi-infinite, or infinite interval. Consider a Hilbert space $\mathscr{H}(K)$ of functions $f : D^{\mathbb{N}} \to \mathbb{R}$, which is defined above in terms of K_1 and the weights γ_j satisfying assumptions* (3.2)*. Then the worst-case error for approximating the expectation of these functions by the single-level algorithm $\hat{I}$ in* (2.19) *is*

$$\begin{aligned} e_{wor}(\hat{I},F) &= \sup_{f\in F} |I(f) - \hat{I}(f)| \\ &= \sqrt{\textit{worst-bias}^2(d;K) + \mathscr{D}^2\left(\{\mathbf{x}_{i,1:d}\}_{i=1}^n; K^{(d)}\right)}, \end{aligned}$$

where

$$\textit{worst-bias}^2(d;K) = \prod_{j=1}^{d}[1+\gamma_j m]\left[\prod_{j=d+1}^{\infty}[1+\gamma_j m]-1\right],$$

$$\mathscr{D}^2(\{\mathbf{x}_{i,1:d}\}_{i=1}^n;K^{(d)}) = \prod_{j=1}^{d}\left[1+\gamma_j m\right] - \frac{2}{n}\sum_{i=1}^{n}\prod_{j=1}^{d}[1+\gamma_j h_1(x_{i,j})] + \frac{1}{n^2}\sum_{i,k=1}^{n}\prod_{j=1}^{d}[1+\gamma_j K_1(x_{i,j},x_{k,j})],$$

where $h_1(\cdot)$ and m are defined in (3.1b) *and* (3.1d) *respectively.*

Proof 4.1 *The error $I-\hat{I}$ can be written explicitly as a sum of two inner products, using the expressions derived in* (3.11)*,* (3.12)*, and* (3.15)*:*

$$\overbrace{I-\hat{I}}^{\textit{error}} = \overbrace{I-I_d}^{\textit{truncated expansion error}} + \overbrace{I_d-\hat{I}}^{\textit{finite sample error}} \tag{4.3}$$

$$= \left\langle h-h(\cdot,\mathbf{c}_{d+1:\infty}),f\right\rangle_{\mathscr{H}(K)} + \left\langle h(\cdot,\mathbf{c}_{d+1:\infty}) - \frac{1}{n}\sum_{i=1}^{n}K(\cdot,(\mathbf{x}_{i,1:d},\mathbf{c})),f\right\rangle_{\mathscr{H}(K)}.$$

The two functions on the left sides of the inner products in (4.3) *are orthogonal by* (3.10)*. This orthogonality, along with the Pythagorean theorem allows one to compute a tight error bound as follows:*

$$\sup_{||f||_{\mathscr{H}(K)}\leq 1}|I(f)-\hat{I}(f)|^2 = \left\|h-h(\cdot,\mathbf{c}_{d+1:\infty})\right\|^2_{\mathscr{H}(K)} + \left\|h(\cdot,\mathbf{c}_{d+1:\infty}) - \frac{1}{n}\sum_{i=1}^{n}K(\cdot,(\mathbf{x}_{i,1:d},\mathbf{c}))\right\|^2_{\mathscr{H}(K)}. \tag{4.4}$$

The first of these terms is identified as the squared bias, because it does not vanish even as the sample size tends to infinity:

$$\begin{aligned}\textit{worst-bias}^2(d;K) :&= \left\|h-h(\cdot,\mathbf{c}_{d+1:\infty})\right\|^2_{\mathscr{H}(K)} = \|h\|^2_{\mathscr{H}(K)} - \left\|h(\cdot,\mathbf{c}_{d+1:\infty})\right\|^2_{\mathscr{H}(K)}\\ &= \prod_{j=1}^{d}[1+\gamma_j m]\left[\prod_{j=d+1}^{\infty}[1+\gamma_j m]-1\right].\end{aligned} \tag{4.5}$$

The second term in (4.4)*, which is the worst-case error for approximating a d-dimensional integral by a sample average of function values, is the squared discrepancy, which has arisen in many similar error analysis, e.g., [Hic98].*

How fast d goes to ∞ affects how fast the bias vanishes. From the summability condition on the weights γ_j in (3.2) it follows that

$$\prod_{j=d+1}^{\infty}(1+\gamma_j m) \;=\; \exp\left(\sum_{j=d+1}^{\infty}\log(1+\gamma_j m)\right) \le \exp\left(\sum_{j=d+1}^{\infty}\gamma_j m\right) \le e^{\alpha m(d+1)^{-2q}}.$$

From the above and (3.13), the squared worst-case bias satisfies

$$\text{worst-bias}^2(d;K) \le e^{\alpha m}\left(e^{\alpha m(d+1)^{-2q}}-1\right) \le \frac{C_1^2}{d^{2q}}, \tag{4.6}$$

where C_1 is some constant and q is the decay order defined in (3.2).

Next, we investigate the mean square discrepancies for the simple i.i.d. random sequence in the randomized worst-case setting and the discrepancy for the low discrepancy sequence in the worst-case setting.

Simple Monte Carlo Sampling: For simple Monte Carlo sampling, the mean square discrepancy is given by [Hic98],

$$\mathbb{E}\left[\mathscr{D}^2\left(\{\mathbf{x}_{i,1:d}\}_{i=1}^n; K^{(d)}\right)\right] = \frac{1}{n}\left\{\prod_{j=1}^{d}[1+\gamma_j M]-\prod_{j=1}^{d}[1+\gamma_j m]\right\} \le \frac{C_2^2}{n},$$

where $C_2^2 = \prod_{j=1}^{\infty}[1+\gamma_j M]-\prod_{j=1}^{\infty}[1+\gamma_j m]$. Note that m and M are defined in (3.1d) and (3.1e). Although the convergence rate is relatively slow, it is dimension independent. Combining this with (4.6), the randomized squared worst-case error satisfies

$$\mathbb{E}\left[\sup_{\|f\|_{\mathscr{H}(K)}\le 1}\left|\hat{I}(f)-I(f)\right|\right]^2 \le \frac{C_1^2}{d^{2q}}+\frac{C_2^2}{n}.$$

Quasi-Monte Carlo Sampling: Results on strong tractability, e.g., [Woź00], show that it is possible to obtain

$$\mathscr{D}\left(\{\mathbf{x}_{i,1:d}\}_{i=1}^{n}; K^{(d)}\right) \leq \frac{C_2}{n^p},$$

for good designs $\{\mathbf{x}_{i,1:d}\}$ with $\gamma_j \to \infty$ fast enough. Here C_2 is a constant independent of d and n but dependent on γ. This implies that the squared worst-case error for the low discrepancy sequence is

$$\sup_{\|f\|_{\mathscr{H}(K)} \leq 1} \left|\hat{I}(f) - I(f)\right|^2 \leq \frac{C_1^2}{d^{2q}} + \frac{C_2^2}{n^{2p}}. \tag{4.7}$$

The value of p is determined by the specific sequence used. Table 4.1 gives some values for p depending on q for rank-1 lattices and the Niederreiter (t,s) sequence [Nie92, Sec. 4.5], where q is the decay order of the weights defined in (3.2). The simple random i.i.d sequence is a specific choice with $p = 1/2$.

Table 4.1. The choice of p and q based on different designs

Design	p
Simple random sequence	1/2
Rank-1 lattices [HSW04b]	$\min\left(1, q+\frac{1}{2}\right) - \varepsilon$
Niederreiter (t,s)-sequence $(q > 1/2)$ [YH06]	$\min\left(1, \frac{q}{2}+\frac{1}{4}\right) - \varepsilon$

The explicit upper error bounds for the simple random sequence and the low discrepancy sequence lead to the minimization of the worst-case error, with which an optimal relationship between the sample size, n, and the truncated dimension, d, can be derived. The cost of the approximation algorithm is $N = nd$, as in the fixed subspace sampling cost model from [CDMGR09]. Note that we assume $s = 1$ in this section.

By minimizing the upper bound in (4.7) with respect to d and n, given a budget of

$N = nd$, it is found that d should be chosen as

$$\begin{aligned} d &= \left[\sqrt{\frac{q}{p}}\frac{C_1}{C_2}n^p\right]^{\frac{1}{q}} = \mathscr{O}\left(n^{\frac{p}{q}}\right), \\ e_1^{\text{fix}}(N,F) &\leq \left[\left(\frac{q}{p+q}\right)^{-q/2} C_1^p C_2^r N^{-pq}\right]^{\frac{1}{p+q}} \\ &= \mathscr{O}\left(N^{\frac{-pq}{p+q}}\right). \end{aligned}$$

The above equation describes how to choose d as a function of p, q. The value of p is commonly determined by the smoothness of the function and the quality of the designs as shown in Table 1. The choice of q depends on how fast the weight γ_j decays to 0. Its choice for practical applications is still an open question. Note that for the illustrated example in Section 2.3.1.1 , $q = 1$ and $p = 1/2$ (simple Monte Carlo) or 1 (quasi-Monte Carlo). This yields values of $\beta = pq/(p+q)$ corresponding to $1/3$ for simple Monte Carlo sampling and $1/2$ for quasi-Monte Carlo sampling, as observed empirically in Figure 2.4.

4.3 The Error Analysis of the Multi-Level Algorithm

Let us recall from Section 2.3.1.2 for the construction of the multi-level algorithm. For $f \in \mathscr{H}(K)$ we employ the orthogonal decomposition

$$f = \Psi_{1:d_1} f + \sum_{\ell=2}^{L} (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}}) f + f - \Psi_{1:d_L} f. \tag{4.8}$$

For integration of these parts we choose quadrature formulas $\hat{I}_1, \ldots, \hat{I}_L$, and we apply the so-called multi-level algorithm

$$\hat{I} = \hat{I}_1 \circ \Psi_{1:d_1} + \sum_{\ell=2}^{L} \hat{I}_\ell \circ (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}}) \tag{4.9}$$

for integration of $\Psi_{1:d_L} f$.

Suppose that

$$\hat{I}_\ell(f) = \sum_{i=1}^{n_\ell} a_i^{(\ell)} f(\mathbf{x}_i^{(\ell)})$$

with coefficients $a_i^{(\ell)} \in \mathbb{R}$ and knots $\mathbf{x}_i^{(\ell)} \in \mathscr{X}$. Then the quadrature formulas $\hat{I}_\ell \circ \Psi_{1:d_\ell}$ and $\hat{I}_\ell \circ \Psi_{1:d_{\ell-1}}$ for $\ell > 1$ use the coefficients $a_i^{(\ell)} \in \mathbb{R}$ together with the knots

$$((\mathbf{x}_i^{(\ell)})_{1:d_\ell}, \mathbf{c}) \in \mathscr{X}_{1:d_\ell}$$

and

$$((\mathbf{x}_i^{(\ell)})_{1:d_{\ell-1}}, \mathbf{c}) \in \mathscr{X}_{1:d_{\ell-1}},$$

respectively. In particular,

$$C_s^{\mathrm{var}}(\hat{I}) \le n_1 \cdot d_1^s + \sum_{\ell=2}^{L} n_\ell \cdot (d_\ell^s + d_{\ell-1}^s) \le 2 \cdot \sum_{\ell=1}^{L} n_\ell \cdot d_\ell^s \tag{4.10}$$

holds for the cost of the multi-level algorithm $\hat{I}$ in the variable subspace model.

4.3.1 General error bounds. Now we turn to the error analysis of multi-level algorithms. Put

$$b_{1:d}(F) = \sup_{f \in F} |I(f) - I(\Psi_{1:d} f)|,$$

which is the worst case truncation error, and let $B(K_{1:d})$ denote the unit ball in $\mathscr{H}(K_{1:d})$.

Theorem 4.2 *Under the assumptions* (3.1e)*,* (3.1a)*, and* (3.2) *the error of the multi-level algorithm $\hat{I}$ satisfies*

$$\begin{aligned} &e_{wor}^2(\hat{I}, F) \\ &\quad = e_{wor}^2(\hat{I}_1, B(K_{1:d_1})) + \sum_{\ell=2}^{L} \Big(e_{wor}^2(\hat{I}_\ell, B(K_{1:d_\ell})) - e_{wor}^2(\hat{I}_\ell, B(K_{1:d_{\ell-1}})) \Big) \\ &\qquad + b_{1:d_L}^2(F). \end{aligned}$$

Proof 4.2 *Let h denote the representer of I, and let g_ℓ denote the representer of $\hat{I}_\ell$, i.e.,*

$$I(f) = \langle f, h \rangle_{\mathscr{H}(K)}$$

and

$$\hat{I}_\ell(f) = \langle f, g_\ell \rangle_{\mathscr{H}(K)}$$

for every $f \in \mathscr{H}(K)$. Then the representer g of $\hat{I}$ is given by

$$g = \Psi_{1:d_1} g_1 + \sum_{\ell=2}^{L} (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}}) g_\ell \in \mathscr{H}(K_{1:d_L}),$$

which implies $\Psi_{1:d_L} g = g$. Use the orthogonal decomposition (4.8) *for $f = h - g$ to obtain*

$$\begin{aligned} e^2_{wor}(\hat{I}, F) &= \|h - g\|^2_{\mathscr{H}(K)} \\ &= \|\Psi_{1:d_1}(h - g_1)\|^2_{\mathscr{H}(K)} + \sum_{\ell=2}^{L} \|(\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})(h - g_\ell)\|^2_{\mathscr{H}(K)} \\ &\quad + \|h - \Psi_{1:d_L} h\|^2_{\mathscr{H}(K)}. \end{aligned}$$

Moreover, for $\ell > 1$,

$$\begin{aligned} &\|(\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})(h - g_\ell)\|^2_{\mathscr{H}(K)} \\ &\quad = \|\Psi_{1:d_\ell}(h - g_\ell)\|^2_{\mathscr{H}(K)} - \|\Psi_{1:d_{\ell-1}}(h - g_\ell)\|^2_{\mathscr{H}(K)} \\ &\quad = e^2_{wor}(\hat{I}_\ell, B(K_{1:d_\ell})) - e^2_{wor}(\hat{I}_\ell, B(K_{1:d_{\ell-1}})), \end{aligned}$$

and

$$\|\Psi_{1:d_1}(h - g_1)\|^2_{\mathscr{H}(K)} = e^2_{wor}(\hat{I}_1, B(K_{1:d_1})).$$

According to Theorem 4.2 the squared error of the multi-level algorithm $\hat{I}$ can be decomposed into its squared truncation error and differences of squared errors of the quadrature formulas $\hat{I}_\ell$ for integration in dimensions d_ℓ and $d_{\ell-1}$. Note that these differences are always non-negative.

Remark 4.1 *In the particular case $L = 1$, i.e., for a single-level algorithm, Theorem 4.2 yields the decomposition*

$$e^2_{wor}(\hat{I}, F) = b^2_{1:d}(F) + e^2_{wor}(\hat{I}, B(K_{1:d})), \tag{4.11}$$

which is due to [NH10, Thm. 1]. See [HMGNR10, Lemma 8] for a counterpart for randomized algorithms.

Put

$$m = \int_D \int_D K_1(x,y)\,\rho_1(\mathrm{d}x)\,\rho_1(\mathrm{d}y)$$

and note that (3.1e) *implies* $m < \infty$. *Hence*

$$b^2_{1:d}(F) = \prod_{j=1}^{d}(1+\gamma_j m)\left(\prod_{j=d+1}^{\infty}(1+\gamma_j m)-1\right) \preceq \sum_{j=d+1}^{\infty}\gamma_j \tag{4.12}$$

holds for the truncation error, see [NH10].

Remark 4.2 *In the case of equal weight quadrature formulas*

$$\hat{I}_\ell(f) = \frac{1}{n_\ell}\sum_{i=1}^{n_\ell} f(\mathbf{x}_i^{(\ell)})$$

the term $e^2_{wor}(\hat{I}_\ell, B(K_{1:d_\ell})) - e^2_{wor}(\hat{I}_\ell, B(K_{1:d_{\ell-1}}))$ *is the difference between square discrepancies of the same design* $\{\mathbf{x}_1^{(\ell)},\dots,\mathbf{x}_{n_\ell}^{(\ell)}\}$ *with respect to the kernel functions* $K_{1:d_\ell}$ *and* $K_{1:d_{\ell-1}}$.

Our overall goal is to minimize the error of multi-level algorithms $\hat{I}$ subject to a cost bound $C_s^{\mathrm{var}}(\hat{I}) \le N$. While Theorem 4.2 provides an explicit representation of the error, it is technically difficult to directly work with the differences of errors for finite-dimensional integration problems. Thus we are interested in useful upper bounds for these differences. The trivial bound

$$e^2_{\mathrm{wor}}(\hat{I}_\ell, B(K_{1:d_\ell})) - e^2_{\mathrm{wor}}(\hat{I}_\ell, B(K_{1:d_{\ell-1}})) \le e^2_{\mathrm{wor}}(\hat{I}_\ell, B(K_{1:d_\ell}))$$

immediately removes any advantage of the multi-level approach, but a modification of this idea works well. To this end we introduce a suitable kernel function K', which induces a weaker norm than K, such that

$$\|(\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})f\|_{\mathscr{H}(K')} \le \kappa_\ell^{1/2} \cdot \|(\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})f\|_{\mathscr{H}(K)}$$

with suitable numbers $\kappa_\ell \le 1$. For randomized algorithms a counterpart of this approach was developed in [HMGNR10].

For the construction of K' we consider another sequence $\gamma' = (\gamma'_1, \gamma'_2, \ldots)$ of positive weights, which satisfies the conditions

$$\gamma_j \leq \gamma'_j \tag{4.13}$$

for every $j \in \mathbb{N}$ and

$$\sum_{j=1}^{\infty} \gamma'_j < \infty. \tag{4.14}$$

We use these new weights to define

$$K'(\mathbf{x}, \mathbf{y}) = \prod_{j=1}^{\infty} (1 + \gamma'_j K_1(x_j, y_j))$$

for $\mathbf{x}, \mathbf{y} \in \mathscr{X}'$ with

$$\mathscr{X}' = \left\{ \mathbf{x} \in D^{\mathbb{N}} : \sum_{j=1}^{\infty} \gamma'_j K_1(x_j, x_j) < \infty \right\}.$$

The kernel functions K and K' differ only in terms of the different weights used to define them. Hence Lemma 3.6 applies to K' as well, and the projections Ψ_v act on $\mathscr{H}(K')$ in the same way as they act on $\mathscr{H}(K)$.

Let γ'_u etc. be defined in the canonical way. From (4.13) we get $\gamma_u \leq \gamma'_u$ as well as $\mathscr{X}' \subseteq \mathscr{X}$. If $\mathscr{X}' = \mathscr{X}$ then $\mathscr{H}(K) \subseteq \mathscr{H}(K')$. In general,

$$if = f|_{\mathscr{X}'}$$

defines a bounded linear mapping $i : \mathscr{H}(K) \to \mathscr{H}(K')$. Furthermore, we may identify the sets $\mathscr{H}(k_v)$ and $\mathscr{H}(k'_v)$ as well as $\mathscr{H}(K_v)$ and $\mathscr{H}(K'_v)$, since their elements only depend on the variables x_j with $j \in v$.

Theorem 4.3 *Put* $\kappa_1 = 1$ *and let*

$$\kappa_\ell = \max_{d_{\ell-1} < j \leq d_\ell} \frac{\gamma_j}{\gamma'_j}$$

for $2 \leq \ell \leq L$. *Under the assumptions* (3.1e), (3.1a), (4.13), *and* (4.14) *the error of the multi-level algorithm* $\hat{I}$ *satisfies*

$$e^2_{wor}(\hat{I}, F) \leq b^2_{1:d_L}(F) + \sum_{\ell=1}^{L} \kappa_\ell \cdot e^2_{wor}(\hat{I}_\ell, B(K'_{1:d_\ell})).$$

Proof 4.3 *In view of Theorem 4.2 it suffices to show that*

$$e^2_{wor}(\hat{I}_\ell, B(K_{1:d_\ell})) - e^2_{wor}(\hat{I}_\ell, B(K_{1:d_{\ell-1}})) \le \kappa_\ell \cdot e^2_{wor}\hat{I}_\ell, B(K'_{1:d_\ell}))$$

for $\ell > 1$ *and*

$$e^2_{wor}(\hat{I}_1, B(K_{1:d_1})) \le e^2_{wor}(\hat{I}_1, B(K'_{1:d_1})).$$

At first we determine the adjoint i^* *of* i. *Let* $f' = \sum_{u\in\mathbb{U}} f'_u \in \mathscr{H}(K')$ *and* $g = \sum_{u\in\mathbb{U}} g_u \in \mathscr{H}(K)$ *according to Lemma 3.6. Then*

$$f = \sum_{u\in\mathbb{U}} \frac{\gamma_u}{\gamma'_u} f'_u \in \mathscr{H}(K)$$

and

$$\langle f, g\rangle_{\mathscr{H}(K)} = \sum_{u\in\mathbb{U}} \gamma_u^{-1} \left\langle \frac{\gamma_u}{\gamma'_u} f'_u, g_u \right\rangle_{\mathscr{H}(k_u)} = \sum_{u\in\mathbb{U}} (\gamma'_u)^{-1} \langle f'_u, g_u\rangle_{\mathscr{H}(k_u)} = \langle f', g\rangle_{\mathscr{H}(K')},$$

i.e., $i^*(f') = f$.

Put

$$\mathbb{U}_1 = \{u \in \mathbb{U} : \quad u \subseteq 1:d_1\}$$

and

$$\mathbb{U}_\ell = \{u \in \mathbb{U} : \quad u \subseteq 1:d_\ell \text{ and } u \not\subseteq 1:d_{\ell-1}\}$$

for $\ell > 1$. *If* $f'_u = 0$ *for every* $u \notin \mathbb{U}_\ell$, *then*

$$\begin{aligned} \|i^* f'\|^2_{\mathscr{H}(K)} &= \langle f', ii^* f'\rangle_{\mathscr{H}(K')} = \sum_{u\in\mathbb{U}_\ell} \frac{\gamma_u}{\gamma'_u} (\gamma'_u)^{-1} \langle f'_u, f'_u\rangle_{\mathscr{H}(k_u)} \\ &\le \max_{u\in\mathbb{U}_\ell} \frac{\gamma_u}{\gamma'_u} \|f'\|^2_{\mathscr{H}(K')} \le \kappa_\ell \|f'\|^2_{\mathscr{H}(K')}. \end{aligned} \tag{4.15}$$

Now we use the notation and facts from the proof of Theorem 4.2. Let h' *and* g'_ℓ *denote the representers of* I *and* $\hat{I}_\ell$, *respectively, on the space* $\mathscr{H}(K')$. *Put*

$$f'_\ell = (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})(h' - g'_\ell)$$

and

$$f_\ell = (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}})(h - g_\ell)$$

for $\ell > 1$ *as well as*

$$f'_1 = \Psi_{1:d_1}(h' - g'_1)$$

and

$$f_1 = \Psi_{1:d_1}(h - g_1).$$

Note that f'_ℓ *and* f_ℓ *represent the same function, and therefore* $i^* f'_\ell = f_\ell$ *for* $\ell = 1, \ldots, L$. *From* (4.15) *we get*

$$\begin{aligned} e^2_{wor}(\hat{I}_\ell, B(K_{1:d_\ell})) - e^2_{wor}(\hat{I}_\ell, B(K_{1:d_{\ell-1}})) &= \|f_\ell\|^2_{\mathscr{H}(K)} \\ &\leq \kappa_\ell \cdot \|f'_\ell\|^2_{\mathscr{H}(K')} = \kappa_\ell \cdot \left(e^2_{wor}(\hat{I}_\ell, B(K'_{1:d_\ell})) - e^2_{wor}(\hat{I}_\ell, B(K'_{1:d_{\ell-1}})) \right) \\ &\leq \kappa_\ell \cdot e^2_{wor}(\hat{I}_\ell, B(K'_{1:d_\ell})) \end{aligned}$$

if $\ell > 1$, *as claimed. Analogously,*

$$e^2_{wor}(\hat{I}_1, B(K_{1:d_1})) = \|f_1\|^2_{\mathscr{H}(K)} \leq \|f'_1\|^2_{\mathscr{H}(K')} = e^2_{wor}(\hat{I}_\ell, B(K'_{1:d_\ell})).$$

The advantage of introducing the new set of weights γ'_j is an upper bound of a simpler form in Theorem 5.1, which is suitable for optimization under the cost constraint $C^{\mathrm{var}}_s(\hat{I}) \leq N$. The disadvantage is that the upper bound is not necessarily tight. The next section explores the choice of γ'_j, L, d_ℓ, n_ℓ, and $\hat{I}_\ell$ to make the upper bound as small as possible under the given cost constraint.

4.3.2 Error bounds under strong tractability assumptions. In the sequel we strengthen our assumptions on the sequences γ and γ' of weights as well as on the reproducing kernel K_1. Concerning the weights we assume that

$$\gamma_j \preceq j^{-1-2q} \tag{4.16}$$

and

$$\gamma'_j = j^{2(q-q')} \cdot \gamma_j \tag{4.17}$$

with

$$q \geq q' > 0,$$

which implies (3.2), (4.13), and (4.14). Furthermore,

$$b^2_{1:d}(F) \preceq d^{-2q} \tag{4.18}$$

follows from (4.16).

For the finite-dimensional integration problems on the unit balls $B(K'_{1:d})$, we apply the strong tractability result, namely,

$$\sup_{d\in\mathbb{N}} \inf\{e_{\mathrm{wor}}(\hat{I}, B(K'_{1:d})) : \hat{I}\ n\text{-point quadrature formula}\} \preceq n^{-p'} \tag{4.19}$$

with $p' > 0$, see, e.g., [HSW04b, NW01, Woź00, YH06, HN03] as well as Section 4.4. Note that p' typically depends on q' and the kernel K_1.

For a vector $\mathbf{d} \in \mathbb{N}^L$ of increasing dimensions $d_1 \leq \ldots \leq d_L$ and a vector $\mathbf{n} \in \mathbb{N}^L$ of integers $n_1, \ldots, n_L$ we put

$$\mathscr{U}^2(\mathbf{n}, \mathbf{d}) = d_L^{-2q} + \sum_{\ell=1}^{L} n_\ell^{-2p'} \cdot d_{\ell-1}^{-2(q-q')},$$

where $d_0 = 1$, as well as

$$\mathscr{C}(\mathbf{n}, \mathbf{d}) = \sum_{\ell=1}^{L} n_\ell \cdot d_\ell^s.$$

Lemma 4.1 *Assume* (3.1e), (3.1a), (4.16), (4.17), *and* (4.19). *For every $L \in \mathbb{N}$ and all vectors $\mathbf{n} \in \mathbb{N}^L$ of integers and all vectors $\mathbf{d} \in \mathbb{N}^L$ of increasing dimensions there exists a multi-level algorithm $\hat{I}_{\mathbf{n},\mathbf{d}}$ such that*

$$e_{wor}(\hat{I}_{\mathbf{n},\mathbf{d}}, F) \preceq \mathscr{U}(\mathbf{n}, \mathbf{d})$$

and

$$C_s^{\mathrm{var}}(\hat{I}_{\mathbf{n},\mathbf{d}}) \leq 2\mathscr{C}(\mathbf{n}, \mathbf{d}).$$

Proof 4.4 *Due to assumption* (4.19) *there exist quadrature formulas* $\hat{I}_{n_\ell,d_\ell}$ *that use* n_ℓ *knots from the space* $\mathscr{X}_{1:d_\ell}$ *and satisfy*

$$e_{wor}(\hat{I}_{n_\ell,d_\ell}, B(K'_{1:d_\ell})) \preceq n_\ell^{-p'}$$

for $\ell = 1,\dots,L$. *Consider the multi-level algorithm*

$$\hat{I}_{\mathbf{n},\mathbf{d}} = \hat{I}_{n_1,d_1} \circ \Psi_{1:d_1} + \sum_{\ell=2}^{L} \hat{I}_{n_\ell,d_\ell} \circ (\Psi_{1:d_\ell} - \Psi_{1:d_{\ell-1}}).$$

Clearly, (4.10) *yields the cost bound for* $\hat{I}_{\mathbf{n},\mathbf{d}}$. *Furthermore,* (4.16) *and* (4.17) *imply*

$$\kappa_\ell = \max_{d_{\ell-1}<j\le d_\ell} j^{-2(q-q')} = d_{\ell-1}^{-2(q-q')}$$

for $\ell > 1$. *Observe* (4.18) *and apply Theorem 5.1 to obtain the error bound for* $\hat{I}_{\mathbf{n},\mathbf{d}}$.

We minimize the upper error bound $\mathscr{U}(\mathbf{n},\mathbf{d})$ under the cost constraint $2\mathscr{C}(\mathbf{n},\mathbf{d}) \le N$, which leads to an upper bound for the minimal error for variable subspace sampling. The result depends on the parameters q and q', which control the decay of the weights γ_j and γ'_j, on the exponent p' in the tractability assumption (4.19), and on the exponent s, which controls the cost of a single function evaluation.

Theorem 4.4 *Assume* (3.1e), (3.1a), (4.16), (4.17), *and* (4.19). *Then the minimal errors for variable subspace sampling satisfy*

$$e_s^{\mathrm{var}}(N,F) \preceq \begin{cases} N^{-p'\cdot\min\left(1,\frac{q}{p's+q'}\right)} & \text{if } p's+q' \neq q, \\ N^{-p'}\cdot(\log_2 N)^{p'+1/2} & \text{if } p's+q' = q. \end{cases}$$

Proof 4.5 *According to Lemma 4.1*

$$e_s^{\mathrm{var}}(N,F) \preceq \mathscr{U}_N, \tag{4.20}$$

where

$$\mathscr{U}_N = \inf\{\mathscr{U}(\mathbf{n},\mathbf{d}) : 2\mathscr{C}(\mathbf{n},\mathbf{d}) \le N\}.$$

Hence it remains to establish suitable upper bounds for $\mathscr{U}_N$.

Put

$$\eta = q/(p's+q'),$$

and choose

$$\xi \in \begin{cases} \left]\, p'/(q-q'), 1/s \,\right[& \text{if } \eta > 1, \\ \{1/s\} & \text{if } \eta = 1, \\ \left]\, p'/(p's+q'), p'/q \,\right[& \text{if } \eta < 1. \end{cases}$$

Moreover, put

$$\beta = \begin{cases} 1 & \text{if } \eta \geq 1, \\ 1 - \xi q'/p' & \text{otherwise}. \end{cases}$$

We define

$$L = \begin{cases} \lceil \log_2 N \rceil & \text{if } \eta > 1, \\ \lceil \log_2 (N/\log_2 N) \rceil & \text{if } \eta = 1, \\ \lceil \log_2 N/(\xi \cdot (p's+q')/p') \rceil & \text{if } \eta < 1, \end{cases} \tag{4.21}$$

and we define $\mathbf{n}^{(N)} \in \mathbb{N}^L$ and $\mathbf{d}^{(N)} \in \mathbb{N}^L$ by

$$n_\ell = \left\lceil 2^{L-\beta\cdot\ell} \right\rceil, \qquad d_\ell = \left\lceil 2^{\xi\cdot\ell} \right\rceil \tag{4.22}$$

for $\ell = 1, \ldots, L$.

By definition,

$$\mathscr{C}(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp \sum_{\ell=1}^{L} 2^{L-\beta\cdot\ell} \cdot 2^{s\xi\cdot\ell} = 2^L \cdot \sum_{\ell=1}^{L} 2^{(s\xi-\beta)\cdot\ell}$$

and

$$\begin{aligned} \mathscr{U}^2(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) &\asymp 2^{-2L\xi q} + \sum_{\ell=1}^{L} 2^{-2p'\cdot(L-\beta\cdot\ell)} \cdot 2^{-2(q-q')\xi\cdot\ell} \\ &\asymp 2^{-2L\xi q} + 2^{-2Lp'} \cdot \sum_{\ell=1}^{L} 2^{-2((q-q')\xi - p'\beta)\cdot\ell}. \end{aligned}$$

Assume that $\eta > 1$. *Then* $s\xi - \beta = s\xi - 1 < 0$ *and therefore*

$$\mathscr{C}(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp 2^L \asymp N.$$

Furthermore, we have $(q-q')\xi - p'\beta = (q-q')\xi - p' > 0$, *which implies* $q\xi > p'$ *and consequently*

$$\mathscr{U}^2(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp 2^{-2L\xi q} + 2^{-2Lp'} \asymp 2^{-2Lp'} \asymp N^{-2p'}.$$

Next, consider the case $\eta = 1$. *Then* $s\xi - \beta = (q-q')\xi - p'\beta = 0$, *which yields*

$$\mathscr{C}(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp 2^L \cdot L \asymp (N/\log_2 N) \cdot \log_2(N/\log_2 N) \asymp N$$

as well as

$$\begin{aligned}\mathscr{U}^2(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) &\asymp 2^{-2L\xi q} + 2^{-2Lp'} \cdot L \asymp 2^{-2Lp'} \cdot L \\ &\asymp (N/\log_2 N)^{-2p'} \cdot \log_2(N/\log_2 N) \asymp N^{-2p'} \cdot (\log_2 N)^{2p'+1}.\end{aligned}$$

Finally, assume that $\eta < 1$. *Then* $s\xi - \beta = \xi(p's + q')/p' - 1 > 0$, *which implies*

$$\mathscr{C}(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp 2^L \cdot 2^{(s\xi-\beta)\cdot L} = 2^{L\cdot\xi(p's+q')/p'} \asymp N.$$

Moreover, we have $(q-q')\xi - p'\beta = \xi q - p' < 0$ *and*

$$(\log_2 N)p'\eta \le L\xi q \le (\log_2 N)p'\eta + \xi q,$$

which leads to

$$\mathscr{U}^2(\mathbf{n}^{(N)}, \mathbf{d}^{(N)}) \asymp 2^{-2L\xi q} + 2^{-2Lp'} \cdot 2^{2(p'-\xi q)\cdot L} \asymp 2^{-2L\xi q} \asymp N^{-2p'\eta}.$$

Remark 4.3 *The proof of Theorem 4.4 is constructive in the following sense. Assume that* $\gamma'_j \preceq j^{-1-2q'}$ *permits the construction of n-point quadrature formulas* $\hat{I}_{n,d}$ *for all* $n, d \in \mathbb{N}$, *such that* $\hat{I}_{n,d}$ *uses knots from the space* $\mathscr{X}_{1:d}$ *and*

$$\sup_{d\in\mathbb{N}} e_{wor}(\hat{I}_{n,d}, B(K'_{1:d})) \preceq n^{-p'},$$

cf. (4.19). *Then the multi-level algorithms defined by* (4.21) *and* (4.22) *yield the upper bound for* $e_s^{\mathrm{var}}(N,F)$ *in Theorem 4.4. This convergence rate is realized by focusing more sampling effort on the lower dimensions.*

Remark 4.4 *We stress that* (4.20) *is only an upper bound for the minimal error* $e_s^{\mathrm{var}}(N,F)$*, since in its derivation we have imposed a multi-level structure of the quadrature formulas and we have employed auxiliary weights. Nevertheless, we add that the upper bound for* $\mathscr{U}_N$ *from the proof of Theorem 4.4 is sharp, at least if* $q \neq p's+q'$.

For a proof of this fact let $\mathbf{d} \in \mathbb{N}^L$ *with* $d_1 \leq \ldots \leq d_L$ *and* $\mathbf{n} \in \mathbb{N}^L$ *such that* $2\mathscr{C}(\mathbf{n},\mathbf{d}) \leq N$*. Then* $n_1 \leq N$ *and* $n_L d_{L-1}^s \leq n_L d_L^s \leq N$*, and therefore*

$$\begin{aligned}\mathscr{U}^2(\mathbf{n},\mathbf{d}) &\geq n_1^{-2p'} + n_L^{-2p'} \cdot d_{L-1}^{-2(q-q')} + d_L^{-2q} \\ &\geq N^{-2p'} + n_L^{-2p'} (N/n_L)^{-2(q-q')/s} + (N/n_L)^{-2q/s} \\ &= N^{-2p'} + N^{-2p'\eta} \cdot \left(r^{2q(1-1/\eta)/s} + r^{2q/s} \right)\end{aligned}$$

with

$$r = n_L / N^{q'\eta/q}.$$

If $\eta < 1$*, then* $r^{2q(1-1/\eta)/s} + r^{2q/s} \geq 1$ *and therefore* $\mathscr{U}^2(\mathbf{n},\mathbf{d}) \geq N^{-2p'\eta}$*. In any case,* $\mathscr{U}^2(\mathbf{n},\mathbf{d}) \geq N^{-2p'}$.

Together with the upper bound from the proof of Theorem 4.4 this yields

$$\mathscr{U}_N^2 \asymp N^{-2p' \cdot \min\left(1, \frac{q}{p's+q'}\right)}$$

if $q \neq p's+q'$*. In the case* $q = p's+q'$ *we conclude that*

$$N^{-2p'} \preceq \mathscr{U}_N^2 \preceq N^{-2p'} \cdot (\log_2 N)^{2p'+1}.$$

Theorem 4.4 yields

$$\lambda_s^{\mathrm{var}}(F) \geq \tau_s^{\mathrm{var}}(q),$$

where $\lambda_s^{\mathrm{var}}(F)$ is defined in (4.2) and

$$\tau_s^{\mathrm{var}}(q) = \sup \min\left(p', \frac{q}{s+q'/p'}\right) \le \min\left(\sup p', q/s\right). \tag{4.23}$$

Here the suprema are taken over all $q' \le q$ and p' such that the strong tractability assumption (4.19) is satisfied. In some cases, even the upper bound in (4.23) is attainable and $\lambda_s^{\mathrm{var}}(F) = \tau_s^{\mathrm{var}}(q)$, see Section 4.4.

Remark 4.5 *For single-level algorithms, i.e., in the case $L = 1$, there is no advantage in using auxiliary weights γ'_j. Hence we take $q' = q$ so that* (4.19) *becomes a strong tractability assumption for integration on the unit balls $B(K_{1:d})$. Here $p' > 0$ typically depends on q and K_1. Accordingly*

$$\mathscr{U}^2(n,d) = d^{-2q} + n^{-2p'}$$

and

$$\mathscr{C}(n,d) = nd^s.$$

Minimizing $\mathscr{U}^2(n,d)$ with respect to d and n given the constraint $\mathscr{C}(n,d) \le N$ yields the following result. Under the assumptions (3.1e)*,* (3.1a)*,* (4.16)*, and* (4.19) *we have*

$$\lambda_s^{\mathrm{fix}}(F) \ge \frac{p''}{1+p''s/q}$$

for p'' being the supremum over all p' such that the strong tractability assumption (4.19) *is satisfied. See [NH10]. Again the proof is constructive in the sense of Remark 4.3.*

4.3.3 Lower bounds for the minimal errors. To derive lower bounds for the minimal errors $e_s^{\mathrm{fix}}(N,F)$ as well as $e_s^{\mathrm{var}}(N,F)$ for the infinite-dimensional integration problem we consider two extremal cases. Either we only take into account the truncation error and suppose that any finite-dimensional integral can be computed exactly, or we ignore the truncation error and only consider integration with respect to a single variable.

In the latter case we employ the minimal error

$$e(n,B(K_1)) = \inf\{e_{\mathrm{wor}}(\hat{I},B(K_1)) : \hat{I}\ n\text{-point quadrature formula}\}$$

for integration of functions $f : D \to \mathbb{R}$ from the unit ball $B(K_1)$ in the Hilbert space $\mathscr{H}(K_1))$. We assume that

$$m = \int_D \int_D K_1(x,y)\,\rho_1(\mathrm{d}x)\,\rho_1(\mathrm{d}y) > 0, \tag{4.24}$$

which excludes that $\int_D f(x)\,\rho_1(\mathrm{d}x) = 0$ for all $f \in \mathscr{H}(K_1)$.

Theorem 4.5 *Under the assumptions* (3.1e), (3.1a), (3.3), *and* (4.24) *the minimal errors satisfy*

$$e_s^{\mathrm{fix}}(N,F) \succeq \inf_{n\cdot d^s \le N} \left(\left(\sum_{j=d+1}^{\infty} \gamma_j \right)^{1/2} + e(\lfloor N/d^s \rfloor, B(K_1)) \right)$$

and

$$e_s^{\mathrm{var}}(N,F) \succeq \left(\sum_{j=\lfloor N^{1/s} \rfloor+1}^{\infty} \gamma_j \right)^{1/2} + e(N,B(K_1)).$$

Proof 4.6 *At first we derive the lower bound for variable subspace sampling. Consider a quadrature rule $\hat{I}$ with knots $\mathbf{x}_1,\dots,\mathbf{x}_n \in \mathscr{X}$ such that $C_s^{\mathrm{var}}(\hat{I}) \le N$. Then $\hat{I}$ formally is a single-level algorithm with $\{\mathbf{x}_1,\dots,\mathbf{x}_n\} \subseteq \mathscr{X}_{1:d}$ for $d = \lfloor N^{1/s} \rfloor$, and therefore* (4.11) *is applicable. Observe that* (4.24) *imply*

$$b_{1:d}^2(F) \asymp \sum_{j=d+1}^{\infty} \gamma_j.$$

Finally, $n \le N$ and therefore

$$e_{wor}(\hat{I},B(K_{1:d})) \ge \gamma_1^{1/2} \cdot e_{wor}(\hat{I},B(K_{\{1\}})) \ge \gamma_1^{1/2} \cdot e(N,B(K_1)).$$

For fixed subspace sampling $C_s^{\mathrm{fix}}(\hat{I}) \le N$ implies $\{\mathbf{x}_1,\dots,\mathbf{x}_n\} \subseteq \mathscr{X}_{1:d}$ with $n\cdot d^s \le N$. Now we proceed as previously.

In [KSWW10, Thm. 3] and [Gne10, Sec. 4.1] a more sophisticated analysis is used to obtain a lower bound for $e_s^*(N,F)$.

4.4 Examples

We consider the Lebesgue measure ρ_1 on $D = [0,1]$ together with the kernel function

$$K_1(x,y) = \min(x,y)$$

for $x,y \in D$. Clearly (3.1e) is satisfied, and (3.1a) holds for the nominal value $c = 0$. Furthermore, we assume a matching lower bound in (4.16), i.e.,

$$\gamma_j \asymp j^{-1-2q}.$$

It is well known that

$$e(n,B(K_1)) \asymp n^{-1}$$

for the corresponding one-dimensional integration problem.

From Theorem 4.5 we get the following facts, which correspond to lower bounds for minimal errors.

Corollary 4.1 *We have*

$$\lambda_s^{\mathrm{fix}}(F) \le \frac{q}{q+s}$$

and

$$\lambda_s^{\mathrm{var}}(F) \le \min\left(\frac{q}{s}, 1\right).$$

Strong tractability results for the corresponding finite-dimensional integration problems with respect to the uniform distribution on $[0,1]^d$ are established in [HSW04b, YH06], see also [NW10]. We take auxiliary weights γ_j' according to (4.17). Since $\gamma_j' \asymp j^{-1-2q'}$ with $q' > 0$, the strong tractability assumption (4.19) is satisfied for every

$$p' < \min(1, q' + 1/2),$$

and the corresponding n-point quadrature formulas may be chosen as rank-1 lattice rules, see [HSW04b].

Now we turn to the optimization problem (4.23) how to select the parameters of the corresponding multi-level algorithms. In the case

$$0 < q < s/2$$

we choose q' arbitrarily close to zero to obtain p' arbitrarily close to $1/2$ and

$$\tau_s^{\mathrm{var}}(q) = \frac{q}{s}.$$

In the case

$$s/2 \leq q < s + 1/2$$

we choose

$$q' = \frac{q - s/2}{s+1}$$

to obtain p' arbitrarily close to $(q+1/2)/(s+1)$ and finally

$$\tau_s^{\mathrm{var}}(q) \geq \sup_{0<q'\leq q} \min\left(1, q'+1/2, \frac{q}{s+q'/\min(1,q'+1/2)}\right) = \frac{q+1/2}{s+1}.$$

In the case

$$q \geq s + 1/2$$

we choose

$$q' = 1/2$$

to obtain p' arbitrarily close to one and

$$\tau_s^{\mathrm{var}}(q) = 1.$$

The latter analysis and similar arguments for single-level algorithms, see Remark 4.5, imply the following facts, which correspond to upper bounds for the minimal errors. These upper bounds are achieved constructively, see Remark 4.3 and [HSW04b].

Corollary 4.2 *We have*

$$\min\left(\frac{q}{q+s}, \frac{q}{2q/(2q+1)+s}\right) \leq \lambda_s^{\text{fix}}(F)$$

and

$$\min\left(\frac{q}{s}, \frac{q+1/2}{s+1}, 1\right) \leq \lambda_s^{\text{var}}(F).$$

We add that the lower bound for $\lambda_s^{\text{var}}(F)$ in Corollary 5.3 is independently proven in [Gne10, Thm. 4.2] by means of a different approach.

It is interesting to compare Corollary 4.1 and Corollary 5.3 with the following upper and lower bounds for $\lambda_s^*(F)$ from [KSWW10, Thm. 4].

Theorem 4.6 (Kuo et al. (2010)) *We have*

$$\max\left(\frac{q}{q+1}, \min\left(\frac{q}{q+s}, \frac{q}{2q/(2q+1)+s}\right)\right) \leq \lambda_s^*(F) \leq \min\left(\frac{q}{\min(s,1)}, 1\right).$$

Let us consider the particular case $s = 1$. Then we have

$$\lambda_1^{\text{fix}}(F) = \frac{q}{q+1}$$

if $q \geq 1/2$, while we only know that

$$\frac{q}{2q/(2q+1)+1} \leq \lambda_1^{\text{fix}}(F) \leq \frac{q}{q+1}$$

otherwise. Hence the single-level algorithm according to Remark 4.5 is optimal for fixed subspace sampling in the case $q \geq 1/2$ and close to being optimal for $q < 1/2$, since the fraction of the lower and upper bound for $\lambda_s^{\text{fix}}(F)$ is at least 0.93 for $q \in \left]0, 1/2\right[$. For variable subspace sampling

$$\lambda_1^{\text{var}}(F) = \min(q, 1) \tag{4.25}$$

for $|q-1| \geq 1/2$, which shows that the multi-level algorithm according to Remark 4.3 is optimal in this case. For $|q-1| < 1/2$ we only know that

$$\frac{q+1/2}{2} \leq \lambda_1^{\text{var}}(F) \leq \min(q, 1) \tag{4.26}$$

with a significant gap between the upper and the lower bound. Still variable subspace sampling is superior to fixed subspace sampling for all $q > 0$. Finally, by Theorem 4.6,

$$\frac{q}{q+1} \leq \lambda_1^*(F) \leq \min(q,1). \tag{4.27}$$

The corresponding upper bound $N^{-q/(q+1)}$ for the minimal error $e_1^*(N,F)$ is achieved constructively by the so-called changing dimension algorithms introduced in [KSWW10]. The bound is always larger than the corresponding error bound for a suitable multi-level algorithm, and it is close to or even coincides with the upper error bound for a suitable single-level algorithm. Combining (4.25), (4.27), and $\lambda_1^{\mathrm{var}}(F) \leq \lambda_1^*(F)$ we obtain

$$\begin{aligned} \lambda_1^{\mathrm{var}}(F) = \lambda_1^*(F) &= \min(q,1), \quad \text{if } |q-1| \geq 1/2, \\ \frac{q+1/2}{2} \leq \lambda_1^{\mathrm{var}}(F) \leq \lambda_1^*(F) &\leq \min(q,1), \quad \text{if } |q-1| < 1/2. \end{aligned}$$

Thus we have optimality of the multi-level algorithm even in the cost model from [KSWW10] for $|q-1| \geq 1/2$, and in the case $|q-1| < 1/2$ we get an improved lower bound $\lambda_1^*(F) \geq \frac{q+1/2}{2}$ in (4.27). See Figure 4.1 for an illustration.

We stress, however, that for every fixed $q > 0$

$$\lim_{s\to\infty} \lambda_s^{\mathrm{var}}(F) = 0,$$

while

$$\inf_{s>0} \lambda_s^*(F) \geq \frac{q}{q+1} > 0.$$

Consequently, for large values of s, the changing dimension algorithm together with the cost model from [KSWW10] outperforms variable subspace sampling.

Niederreiter (t,d)-nets may serve as well as building blocks for the multi-level construction. In fact, if $q \geq q' > 1/2$ then the corresponding equal weight quadrature formulas yield any exponent

$$p' < \min(1, q'/2 + 1/4)$$

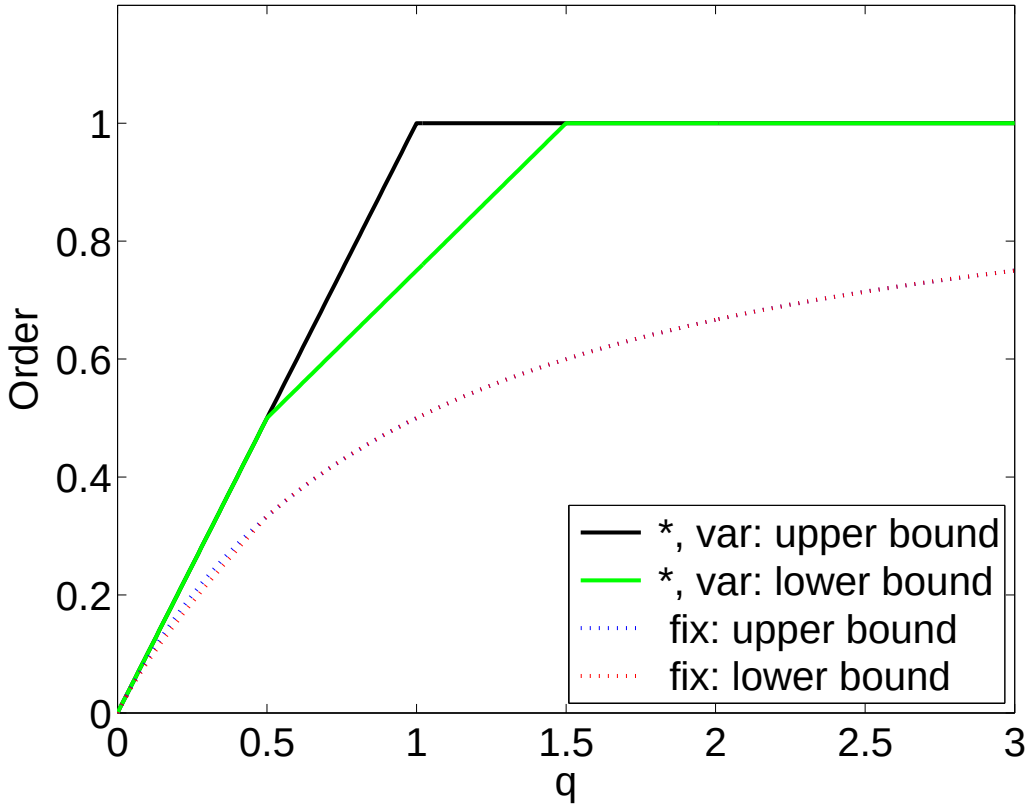


Figure 4.1. Upper and lower bounds for the exponents $\lambda_1^\dagger(F)$ in the case $K = \min$.

in the strong tractability estimate (4.19), see [YH06]. Exploiting this range of parameters in the optimization problem (4.23) we obtain

$$\tau_s^{\mathrm{var}}(q) \geq \min\left(\frac{q}{s+1}, \frac{q+1/2}{s+2}, 1\right).$$

Due to Corollary 5.3 this bound is suboptimal as long as $q < s+3/2$. For larger values of q, however, Niederreiter (T,d)-nets together with the multi-level construction achieve optimality, see Corollary 4.1.

Remark 4.6 *Although the above discussion focuses on ρ_1 as the uniform distribution on $D = [0,1]$, the results derived in the previous sections can also be applied to the case where ρ_1 is the Gaussian distribution on $D = \mathbb{R}$, as suggested by the option pricing example. See [YH06, HSW04a] for relevant kernels, K_1, and strong tractability results for this case.*

Remark 4.7 *The lattice and net designs discussed previously have the advantage of being extensible in both dimension and sample size. This allows one to use parts of one large design for each level of the multi-level algorithm. Specifically, one can remove the super-script (ℓ) labeling the design points in Remark 4.2 and re-arrange the terms to arrive at an equivalent formula that uses fewer arithmetic operations.*

To illustrate this fact assume $n_1 > n_2 > \cdots > n_L > n_{L+1} = 0$, and put $d_0 = 0$ as well as $f(\mathbf{x}_{1:d_0}, \mathbf{c}) = 0$ for every $f \in \mathscr{H}(K)$ and every $\mathbf{x} \in \mathscr{X}$. Then the multi-level algorithm $\hat{I}$ satisfies

$$\begin{aligned}
\hat{I}(f) &= \sum_{\ell=1}^{L} \frac{1}{n_\ell} \sum_{i=1}^{n_\ell} \Big(f((\mathbf{x}_i)_{1:d_\ell}, \mathbf{c}) - f((\mathbf{x}_i)_{1:d_{\ell-1}}, \mathbf{c}) \Big) \\
&= \sum_{\ell=1}^{L} \sum_{k=\ell}^{L} \sum_{i=n_{k+1}+1}^{n_k} \frac{1}{n_\ell} \Big(f((\mathbf{x}_i)_{1:d_\ell}, \mathbf{c}) - f((\mathbf{x}_i)_{1:d_{\ell-1}}, \mathbf{c}) \Big) \\
&= \sum_{k=1}^{L} \sum_{i=n_{k+1}+1}^{n_k} \sum_{\ell=1}^{k} \frac{1}{n_\ell} \Big(f((\mathbf{x}_i)_{1:d_\ell}, \mathbf{c}) - f((\mathbf{x}_i)_{1:d_{\ell-1}}, \mathbf{c}) \Big) \\
&= \sum_{k=1}^{L} \sum_{i=n_{k+1}+1}^{n_k} \left(\frac{1}{n_k} f((\mathbf{x}_i)_{1:d_k}, \mathbf{c}) + \sum_{\ell=1}^{k-1} \left(\frac{1}{n_\ell} - \frac{1}{n_{\ell+1}} \right) f((\mathbf{x}_i)_{1:d_\ell}, \mathbf{c}) \right) \\
&= \sum_{k=1}^{L} \left(\frac{1}{n_k} \sum_{i=n_{k+1}+1}^{n_k} f((\mathbf{x}_i)_{1:d_k}, \mathbf{c}) + \sum_{\ell=1}^{k-1} \frac{n_{\ell+1} - n_\ell}{n_\ell \cdot n_{\ell+1}} \sum_{i=n_{k+1}+1}^{n_k} f((\mathbf{x}_i)_{1:d_\ell}, \mathbf{c}) \right).
\end{aligned}$$

We observe that for each point $\mathbf{x}_i \in \mathscr{X}$ of the design with $n_{\ell+1} < i \leq n_\ell$ one only uses the first d_ℓ components. Moreover,

$$C_s^{\mathrm{var}}(\hat{I}) \leq \sum_{\ell=1}^{L} n_\ell \cdot d_\ell^s,$$

cf. (4.10)*. The savings here do not affect the order of operations required for the multi-level algorithm , but do have an affect on leading constants.*

CHAPTER 5

RESULTS IN THE RANDOMIZED SETTING

This chapter presents the results in the randomized setting following the same structure as in Chapter 4. We first look at cost models and errors defined in the randomized setting, then present the asymptotic error bound in Theorem 5.1, Theorem 5.4 and Theorem 5.5.

5.1 Cost Models and Minimal Errors

5.1.1 The cost models in the randomized setting. We define that the cost per evaluation (oracle call) at a point $\mathbf{x} \in \mathbb{R}^{\mathbb{N}}$ is modeled by a function

$$C : \mathbb{R}^{\mathbb{N}} \to \mathbb{N} \cup \{\infty\}.$$

For *fixed subspace sampling* evaluations are possible only at the points from a finite-dimensional affine subspace

$$\mathscr{X}_{v,c} = \{\mathbf{x} \in \mathbb{R}^{\mathbb{N}} : \ x_j = c \text{ for } j \in \mathbb{N} \setminus v\}$$

for a given (finite) set $\emptyset \neq v \subset \mathbb{N}$ and a given point $c \in D$, and the cost for each oracle call coincides with the dimension $|v|$ of $\mathscr{X}_{v,c}$. Thus,

$$C_v^{\text{fix}}(\mathbf{x}) = \begin{cases} \dim(\mathscr{X}_{v,c}), & \text{if } \mathbf{x} \in \mathscr{X}_{v,c}, \\ \infty, & \text{otherwise.} \end{cases} \tag{5.1}$$

Note that $\mathscr{X}_{v,c} \cap D^{\mathbb{N}} \subseteq \mathscr{X}$.

For *variable subspace sampling* we consider a sequence of finite-dimensional affine subspaces

$$\mathscr{X}_{v_1,c} \subset \mathscr{X}_{v_2,c} \subset \dots$$

for a given increasing sequence $\mathbf{v} = (v_i)_{i\in\mathbb{N}}$ of (finite) sets $\emptyset \neq v_i \subset \mathbb{N}$ and a point $c \in D$, and the cost function is defined by

$$C_{\mathbf{v}}^{\mathrm{var}}(\mathbf{x}) = \inf\{\dim(\mathscr{X}_{v_i,c}) : \mathbf{x} \in \mathscr{X}_{v_i,c}\}, \tag{5.2}$$

with $\inf\emptyset = \infty$ as usual. These sampling regimes and corresponding cost models have been introduced in [CDMGR09] in the context of integration of functions on separable Banach spaces with arbitrary finite-dimensional linear subspaces. In the present setting a generalization of the model, where $C(\mathbf{x})$ depends in any way on the number of components of $\mathbf{x}$ that are different from c, is studied in [KSWW10].

We consider randomized algorithms for integration of functions $f : \mathscr{X} \to \mathbb{R}$, and we refer to [TWW88, CDMGR09] for a formal definition and some rather mild measurability assumptions involved.

We define the cost of a computation as the sum of the cost of all oracle calls that are made during the computation. For a randomized algorithm $\hat{I}$, the cost defines a random variable, which may also depend on f, and this random variable is henceforth denoted by $C(\hat{I}, f)$. Let $\mathscr{C}^{\mathrm{fix}}$ denote the set of all cost functions given by (5.1) with any finite-dimensional affine subspace $\mathscr{X}_v$, and let $\mathscr{C}^{\mathrm{var}}$ denote the set of all cost functions given by (5.2) with any increasing sequence of finite-dimensional affine subspaces $\mathscr{X}_{v_i,c}$. The *worst-case cost* of $\hat{I}$ on a class F of integrands is defined by

$$C^{\mathrm{fix}}(\hat{I}, F) = \inf_{C\in\mathscr{C}^{\mathrm{fix}}} \sup_{f\in F} \mathbb{E}(C(\hat{I}, f))$$

in the fixed subspace model and by

$$C^{\mathrm{var}}(\hat{I}, F) = \inf_{C\in\mathscr{C}^{\mathrm{var}}} \sup_{f\in F} \mathbb{E}(C(\hat{I}, f))$$

in the variable subspace model. Clearly $C^{\mathrm{var}}(\hat{I}, F) \leq C^{\mathrm{fix}}(\hat{I}, F)$.

Let us look at the particular case of a *randomized quadrature formula*

$$\hat{I}(f) = \sum_{\ell=1}^{n} b_\ell f(X_\ell)$$

with deterministic weights $b_\ell \in \mathbb{R}$ and random elements X_ℓ taking values in $\mathscr{X}$. If $\hat{I}$ satisfies the sampling constraint $X_1, \dots, X_n \in \mathscr{X}_{v,c}$ for some finite-dimensional affine subspace $\mathscr{X}_{v,c}$, then

$$C^{\mathrm{fix}}(\hat{I}, F) \leq n \cdot |v|.$$

If $\hat{I}$ satisfies the sampling constraint $X_\ell \in \mathscr{X}_{v_{i_\ell},c} \setminus \mathscr{X}_{v_{i_\ell - 1},c}$ for an increasing sequence of finite-dimensional subspaces $\mathscr{X}_{v_i,c}$ with $\mathscr{X}_{v_{i_0},c} = \emptyset$, then

$$C^{\mathrm{var}}(\hat{I}, F) \leq \sum_{\ell=1}^{n} |v_{i_\ell}|,$$

while

$$C^{\mathrm{fix}}(\hat{I}, F) \leq n \cdot \max_{\ell=1,\dots,n} |v_{i_\ell}|.$$

5.1.2 Defining the randomized error. A randomized algorithm $\hat{I}$ that terminates for every integrand $f \in F$ induces a family $(\hat{I}(f))_{f \in F}$ of random variables, which yield the random outputs of the algorithm for inputs f. The *randomized error* of $\hat{I}$ on the class F is defined by

$$e_{\mathrm{ran}}(\hat{I}, F) = \sup_{f \in F} \left(\mathbb{E}(I(f) - \hat{I}(f))^2 \right)^{1/2}.$$

For $N \in \mathbb{N}$ we introduce the *N-th minimal errors*

$$e^{\mathrm{fix}}(N, F) = \inf\{ e_{\mathrm{ran}}(\hat{I}, F) : C^{\mathrm{fix}}(\hat{I}, F) \leq N \}$$

and

$$e^{\mathrm{var}}(N, F) = \inf\{ e_{\mathrm{ran}}(\hat{I}, F) : C^{\mathrm{var}}(\hat{I}, F) \leq N \}.$$

Clearly we have $e^{\mathrm{var}}(N, F) \leq e^{\mathrm{fix}}(N, F)$. We drop the notation *ran* here for simplicity.

It is worth mentioning that the deterministic algorithm, which deterministically chooses sampling points $\mathbf{x}_i$, can be considered as a special example of the randomized algorithm. Please refer to [TWW88, CDMGR09] for a formal definition and some rather mild measurability assumptions involved.

5.2 The Error Analysis of The Single-Level Algorithm

The analysis of the fixed subspace sampling is motivated by a common approach to infinite-dimensional integration as follows. Let $c \in D$. We use $\mathbf{c}$ to denote the constant sequence in $D^{\mathbb{N}}$ with coordinates c. Furthermore, for a (finite) set $\emptyset \neq v \subset \mathbb{N}$ and $\mathbf{y} \in D^v$, we use $(\mathbf{y}, \mathbf{c})$ to denote the sequence $\mathbf{x} \in D^{\mathbb{N}}$ with $x_j = y_j$ for $j \in v$ and $x_j = c$ otherwise. Moreover, μ_v denotes the product of the measure ρ_1 on D^v. Commonly, the integral $I(f)$ is approximated by

$$\int_{\mathscr{X}} f(\mathbf{x}_v, \mathbf{c})\, \mu(\mathrm{d}\mathbf{x}) = \int_{D^v} f(\mathbf{y}, \mathbf{c})\, \mu_v(\mathrm{d}\mathbf{y}),$$

and for computation of the latter one uses a randomized algorithm $\hat{I}_v$ for integration on $D^{|v|}$ with respect to μ_v. In this way one gets a randomized algorithm $\hat{I}$ with random output

$$\hat{I}(f) = \hat{I}_v(f(\cdot, \mathbf{c})) \tag{5.3}$$

for any integrable function $f : \mathscr{X} \to \mathbb{R}$. Clearly $\hat{I}$ is based on evaluation of f at points from the finite-dimensional affine subspace $\mathscr{X}_{v,c}$, and therefore $C^{\mathrm{fix}}(\hat{I}, f)$ is given as the product of $|v|$ and the number of evaluations of f, which is a random variable and may depend on f. In particular, if $\hat{I}_v$ is c randomized quadrature formula with n evaluations, then $C^{\mathrm{fix}}(\hat{I}, F) \leq n \cdot |v|$ for every class F of integrands.

5.2.1 Preliminaries. For v and c as previously we define

$$(\Psi_{v,c} f)(\mathbf{x}) = f(\mathbf{x}_v, \mathbf{c}), \qquad \mathbf{x} \in \mathscr{X}.$$

Obviously (5.3) implies

$$\hat{I}(f) = \hat{I}(\Psi_{v,c} f). \tag{5.4}$$

We use $B_r(K)$ and $B_r(K_v)$ to denote the closed centered balls with radius r in the spaces $\mathscr{H}(K)$ and $\mathscr{H}(K_v)$, respectively. Furthermore, the unit balls are denoted by F and $B(K_v)$, respectively. We show that the maximal error of $\hat{I}$ on F can essentially be

decomposed into its maximal error on $B(K_v)$ and the quantity

$$b_{v,c} = \sup_{f \in F} \left| I(f) - I(\Psi_{v,c} f) \right|.$$

If $\hat{I}$ is given by (5.3) with an unbiased algorithm $\hat{I}_v$ for integration on $D^{|v|}$, then $b_{v,c}$ is the worst case bias of $\hat{I}$ for integration on $\mathscr{X}$.

Lemma 5.1 *The mapping $\Psi_{v,c}$ maps the unit ball F onto the closed centered ball $B_r(K_v)$ in $\mathscr{H}(K_v)$ with radius $r = r_{v,c}$ given by*

$$r_{v,c}^2 = \sum_{w \subset \mathbb{N} \setminus v} \gamma_w \left(K_1(c,c) \right)^{|w|}.$$

Furthermore, we have $r_{v,c} \geq 1$ and $\sup_{v \neq \emptyset} r_{v,c} < \infty$ as well as $\lim_{d \to \infty} r_{1:d,c} = 1$.

Proof 5.1 *Lemma A.1 from the Appendix with $E_1 = D^{|v|}$, $E_2 = \{\mathbf{x} \in D^{\mathbb{N} \setminus v} : \sum_{j \notin v} \gamma_j K_1(\mathbf{x}_j, \mathbf{x}_j) < \infty\}$, $e_2 = \mathbf{c}$, $\Psi = \Psi_{v,c}$, and*

$$J(\mathbf{x}, \mathbf{y}) = K((\mathbf{x}_v, \mathbf{c}), (\mathbf{y}_v, \mathbf{c})), \qquad \mathbf{x}, \mathbf{y} \in \mathscr{X},$$

yields

$$\Psi_{v,c}(F) = \{g \in \mathscr{H}(J) : \|g\|_{\mathscr{H}(J)} \leq 1\}.$$

Note that

$$J(\mathbf{x}, \mathbf{y}) = \sum_u \gamma_u \prod_{j \in u \cap v} K_1(x_j, y_j) \prod_{j \in u \setminus v} K_1(c,c) = K_v(\mathbf{x}, \mathbf{y}) \cdot r_{v,c}^2.$$

Hence

$$\{g \in \mathscr{H}(J) : \|g\|_{\mathscr{H}(J)} \leq 1\} = \{f \in \mathscr{H}(K_v) : \|f\|_{K_v} \leq r_{v,c}\}.$$

Take $w = \emptyset$ to get $r_{v,c} \geq 1$, and $\sup_v r_{v,c} < \infty$ as well as $\lim_{d \to \infty} r_{1:d,c} = 1$ are due to (3.2).

Lemma 5.2 *Assume that* (5.4) *is satisfied for every $f \in F$. Then*

$$\max(b_{v,c}/(1 + r_{v,c}), e_{ran}(\hat{I}, B(K_v))) \leq e_{ran}(\hat{I}, F) \leq b_{v,c} + e_{ran}(\hat{I}, B_{r_{v,c}}(K_v)).$$

Proof 5.2 *For $f \in F$ we use* (5.4) *to obtain*

$$\left(\mathbb{E}\left(I(f)-\hat{I}(f)\right)^2\right)^{1/2} \le |I(f)-I(\Psi_{v,c}f)| + \left(\mathbb{E}\left(I(\Psi_{v,c}f)-\hat{I}(\Psi_{v,c}f)\right)^2\right)^{1/2}.$$

Due to Lemma 5.1,

$$\sup_{f\in F}\mathbb{E}\left(I(\Psi_{v,c}f)-\hat{I}(\Psi_{v,c}f)\right)^2 \le \sup_{f\in B_{r_{v,c}}(K_v)}\mathbb{E}\left(I(f)-\hat{I}(f)\right)^2,$$

which completes the proof of the upper bound.

Let $f \in F$ and consider the function $g = (1+r_{v,c})^{-1}\cdot(f-\Psi_{v,c}f)$. Then $g \in F$ by Lemma 5.1, and $\Psi_{v,c}g = \Psi_{v,c}(-g) = 0$. Hence

$$\begin{aligned} e^2_{ran}(\hat{I},F) &\ge \max\left(\mathbb{E}\left(I(g)-\hat{I}(\Psi_{v,c}g)\right)^2, \mathbb{E}\left(I(-g)-\hat{I}(\Psi_{v,c}(-g))\right)^2\right) \\ &\ge |I(g)|^2 = (1+r_{v,c})^{-2}\cdot|I(f)-I(\Psi_{v,c}f)|^2, \end{aligned}$$

which yields $e_{ran}(\hat{I},F) \ge b_{v,c}/(1+r_{v,c})$. Furthermore, $e_{ran}(\hat{I},F) \ge e_{ran}(\hat{I},B(K_v))$, since $F \supset B(K_v)$, which completes the proof of the lower bound.

For every w the mapping $f = \sum_u f_u \mapsto I(f_w)$ defines a bounded linear functional on $\mathscr{H}(K)$, and its representer $g_w \in \mathscr{H}_w$ is given by

$$g_w(\mathbf{x}) = \gamma_w \int_{\mathscr{X}} k_w(\mathbf{x},\mathbf{y})\,\mu(\mathrm{d}\mathbf{y}), \qquad \mathbf{x}\in\mathscr{X}.$$

Put

$$A(v,c) = \sum_{\emptyset\neq w\subset\mathbb{N}\setminus v} \|g_w - \gamma_w k_w(\cdot,\mathbf{c})\|^2_{\mathscr{H}(K)}.$$

Recall that the representer h of $f \mapsto I(f)$ is given by (3.14).

Lemma 5.3 *We have*

$$A(v,c) \le b^2_{v,c} \le A(v,c)\cdot\|h\|^2_{\mathscr{H}(K)}$$

and

$$r^2_{v,c} \le 2\,(A(v,c)+\|h\|^2_{\mathscr{H}(K)}).$$

Proof 5.3 *Use Lemma 5.1 to conclude that $f \mapsto I(\Psi_{v,c} f)$ defines a bounded linear functional on $\mathscr{H}(K)$. Its representer is*

$$h_{v,c}(\mathbf{x}) = \int_{D^v} K(\mathbf{x}, (\mathbf{y}, \mathbf{c}))\, \mu_v(\mathrm{d}\mathbf{y}), \qquad \mathbf{x} \in \mathscr{X}. \tag{5.5}$$

We have

$$h = \sum_u g_u, \qquad h_{v,c} = \sum_u g_{u\cap v} \cdot \gamma_{u\setminus v} k_{u\setminus v}(\cdot, \mathbf{c}). \tag{5.6}$$

Since $g_{u\cap v} \cdot \gamma_{u\setminus v} k_{u\setminus v}(\cdot, \mathbf{c}) \in \mathscr{H}_u$ we obtain

$$b_{v,c}^2 = \|h - h_{v,c}\|_{\mathscr{H}(K)}^2 = \sum_u \|g_u - g_{u\cap v} \cdot \gamma_{u\setminus v} k_{u\setminus v}(\cdot, \mathbf{c})\|_{\mathscr{H}(K)}^2.$$

Note that

$$\|g_u - g_{u\cap v} \cdot \gamma_{u\setminus v} k_{u\setminus v}(\cdot, \mathbf{c})\|_{\mathscr{H}(K)} = \|g_{u\cap v}\|_{\mathscr{H}(K)} \cdot \|g_{u\setminus v} - \gamma_{u\setminus v} k_{u\setminus v}(\cdot, \mathbf{c})\|_{\mathscr{H}(K)},$$

and therefore

$$b_{v,c}^2 = A(v,c) \sum_{u\subseteq v} \|g_u\|_{\mathscr{H}(K)}^2.$$

By definition,

$$r_{v,c}^2 = \Big\| \sum_{w\subset \mathbb{N}\setminus v} \gamma_w k_w(\cdot, \mathbf{c}) \Big\|_{\mathscr{H}(K)}^2 \le 2A(v,c) + 2 \sum_{w\subset\mathbb{N}\setminus v} \|g_w\|_{\mathscr{H}(K)}^2.$$

Use $\|g_\emptyset\|_{\mathscr{H}(K)} = 1$ and $\sum_u \|g_u\|_{\mathscr{H}(K)}^2 = \|h\|_{\mathscr{H}(K)}^2$ to derive the estimates for $b_{v,c}^2$ and $r_{v,c}^2$ as claimed.

We provide further estimates for $b_{v,c}$, if the kernel K_1 satisfies one of the following two conditions, both of which imply condition (3.1a), namely,

(A2a) $\int_D K_1(x,y)\, \rho_1(\mathrm{d}y) = 0$ holds for every $x \in D$,

(A2b) there exists a point $c^* \in D$ such that $K_1(c^*, c^*) = 0$.

Remark 5.1 *If K_1 satisfies both conditions (A2a) and (A2b) and if we take $c = c^*$, then $I(f) = f(\mathbf{c})$ for every $f \in \mathscr{H}(K)$, and the quadrature problem is trivial. In fact, (A2a) implies $h = 1$ for the representer of integration in $\mathscr{H}(K)$, while (A2b) implies $K(\cdot, \mathbf{c}) = 1$.*

In the case (A2b) the mapping Ψ_{v,c^*} is the orthogonal projection onto $\mathscr{H}(K_v)$, and therefore $r_{v,c^*} = 1$ in Lemma 5.1, and (A2b) with $c^* = c$ is called the anchored case in the literature.

Define $g \in \mathscr{H}(K_1)$ by

$$g(x) = \int_D K_1(x,y)\,\rho_1(\mathrm{d}y), \qquad x \in D,$$

and put

$$\kappa(c) = \|g - K_1(\cdot, c)\|^2_{\mathscr{H}(K_1)}$$

for $c \in D$.

Lemma 5.4 *Let $c \in D$ and assume that K_1 satisfies* (A2a) *or* (A2b) *with $c^* = c$. Then*

$$\kappa(c) \cdot \sum_{j \notin v} \gamma_j \le b^2_{v,a} \le \kappa(c) \cdot \sum_{j \notin v} \gamma_j \cdot \exp\left(\kappa(c) \cdot \sum_{j \notin v} \gamma_j\right) \cdot \|h\|^2_{\mathscr{H}(K)}$$

for every v.

Proof 5.4 *We apply Lemma 5.3 and derive a corresponding estimate for $A(v,c)$. Since (A2a) implies $g_{\{j\}} = 0$ for every $j \in \mathbb{N}$, and (A2b) with $c^* = c$ implies $k_{\{j\}}(\cdot, \mathbf{c}) = 0$ for every $j \in \mathbb{N}$, we have*

$$g_w - \gamma_w k_w(\cdot, \mathbf{c}) = \prod_{j \in w} g_{\{j\}} - \prod_{j \in w} \gamma_j k_{\{j\}}(\cdot, \mathbf{c}) = \prod_{j \in w} \Big(g_{\{j\}} - \gamma_j k_{\{j\}}(\cdot, \mathbf{c}) \Big),$$

and therefore

$$\begin{aligned} \|g_w - k_w(\cdot, \mathbf{c})\|_{\mathscr{H}(K)} &= \prod_{j \in w} \|g_{\{j\}} - \gamma_j k_{\{j\}}(\cdot, \mathbf{c})\|_{\mathscr{H}(K)} \\ &= \prod_{j \in w} \gamma_j^{1/2} \|g - K_1(\cdot, c)\|_{\mathscr{H}(K_1)} = \gamma_w^{1/2} \left(\kappa(c)\right)^{|w|/2}. \end{aligned}$$

Hence

$$A(v,c) = \sum_{\emptyset \neq w \subset \mathbb{N} \setminus v} \gamma_w \, (\kappa(c))^{|w|} = \sum_{\emptyset \neq w \subset \mathbb{N} \setminus v} \prod_{j \in w} \kappa(c) \, \gamma_j$$

and therefore

$$\kappa(c) \sum_{j \notin v}^{\infty} \gamma_j \leq A(v,c) \leq \exp\left(\kappa(c) \sum_{j \notin v}^{\infty} \gamma_j \right) - 1.$$

5.2.2 Upper and lower bounds. For the proof of upper bounds and the construction of algorithms we consider a family of randomized algorithms $\hat{I}_{n,1:d}$ with $n,d \in \mathbb{N}$ for finite-dimensional integration on $D^{1:d}$ as well as the corresponding randomized algorithms $\hat{I}_{n,d,c} = \hat{I}_{n,1:d} \circ \Psi_{1:d,c}$ for infinite-dimensional integration, see (5.3). Typically, $\hat{I}_{n,1:d}$ is a randomized quadrature formula with n evaluations, and then we might assume that an upper bound for the maximal error of $\hat{I}_{n,1:d}$ on the unit ball in $\mathscr{H}(K_{1:d})$ is available that only depends on n. In general, we have to consider the ball of radius $r_{1:d,c}$, see Lemma 5.1 and Remark 5.1.

Theorem 5.1 *Let $c \in D$. Assume that*

(i) K_1 *satisfies* (A2a) *or* (A2b) *with* $c^* = c$,

(ii) $\gamma_j \preceq j^{-2q-1}$ *with* $q > 0$,

(iii) there exist $p,c > 0$ *such that*

$$e_{ran}(\hat{I}_{n,s,c}, B_{r_{1:d,c}}(K_{1:d})) \leq c \cdot n^{-p}$$

and

$$C^{\mathrm{fix}}(\hat{I}_{n,d,c}, B_{r_{1:d,c}}(K_{1:d})) \leq n \cdot d$$

hold for all $n,d \in \mathbb{N}$.

Choose

$$n_N \asymp N^{\frac{q}{p+q}}$$

and

$$d_N \asymp N^{\frac{p}{p+q}}$$

for $N \in \mathbb{N}$. Then the sequence of randomized algorithms $\hat{I}_N = \hat{I}_{n_N,d_N,c}$ satisfies

$$e_{ran}(\hat{I}_N, F) \preceq N^{-\frac{pq}{p+q}},$$

and

$$C^{\text{fix}}(\hat{I}_N, F) \preceq N.$$

Proof 5.5 *Assumption (iii) together with Lemma 5.2 yields*

$$e_{ran}(\hat{I}_{n_N,d_N,c}, F) \leq b_{1:d_N,c} + c \cdot n_N^{-p}$$

for every $N \in \mathbb{N}$. Use assumptions (i) and (ii) together with Lemma 5.4 to conclude

$$e_{ran}(\hat{I}_{n_N,d_N,c}, F) \preceq d_N^{-q} + n_N^{-p} \asymp N^{-\frac{pq}{p+q}}.$$

By assumption (iii) and Lemma 5.1

$$C^{\text{fix}}(\hat{I}_N, F) \leq C^{\text{fix}}(\hat{I}_{n_N,d_N,c}, B_{r_{1:d_N,c}}(K_{1:d_N})) \leq n_N \cdot d_N,$$

and, clearly, $n_N \cdot d_N \asymp N$.

Now we establish a lower bound, which matches the upper bound from Theorem 5.1 if the minimal errors for one-dimensional integration on the unit ball in the space $\mathscr{H}(K_{\{1\}})$ are of order p, too.

Theorem 5.2 *Assume that*

(i) $\gamma_j \succeq j^{-2q-1}$ *with* $q > 0$,

(ii) there exist $p, c > 0$ *such that*

$$e^{\text{fix}}(N, B(K_{\{1\}})) \geq c \cdot N^{-p}$$

for all $N \in \mathbb{N}$.

Then the minimal errors for integration on the unit ball F using fixed subspace sampling satisfy

$$e^{\mathrm{fix}}(N,F) \succeq N^{-\frac{pq}{p+q}}.$$

Proof 5.6 *Consider any randomized algorithm $\hat{I}$ with $C^{\mathrm{fix}}(\hat{I},F) \le N$. Hence there exists a set $v \subset \mathbb{N}$ and a point $c \in D$ such that $\mathbb{E}(C(\hat{I},f)) \le N+1$ holds for every $f \in F$. Hence, for every $f \in F$, the expected number of evaluations by $\hat{I}$ is at most $(N+1)/|v|$ and (with probability one) these evaluations are made at points from $\mathscr{X}_{v,c}$.*

Due to the latter fact, (5.4) *holds for every $f \in F$, and Lemma 5.2 yields*

$$e_{ran}(\hat{I},F) \ge \frac{b_{v,c}}{1+r_{v,c}}.$$

Since

$$\frac{b_{v,c}^2}{(1+r_{v,c})^2} \ge \frac{b_{v,c}^2}{2\,(1+2b_{v,c}^2+2\|h\|_{\mathscr{H}(K)}^2)},$$

which follows from Lemma 5.3, we derive a lower bound for $b_{v,c}$. Clearly,

$$e^{\mathrm{fix}}(1,B(K_{\{1\}})) \le \sup_{f\in B(K_{\{1\}})} |I(f)-f(\mathbf{c})|.$$

For $f \in F$ we have $I(\Psi_{\{1\},v}f)-f(\mathbf{c}) = \langle h_{\{1\},c}-K(\cdot,\mathbf{c}),f\rangle_{\mathscr{H}(K)}$, see (5.5)*. Moreover,*

$$h_{\{1\},c}-K(\cdot,\mathbf{c}) = \Big(g_{\{1\}}-\gamma_1 k_{\{1\}}(\cdot,\mathbf{c})\Big)\sum_{1\in u}\gamma_{u\setminus\{1\}}k_{u\setminus\{1\}}(\cdot,\mathbf{c})$$

due to (5.6)*, and therefore*

$$\begin{aligned}|I(\Psi_{\{1\},v}f)-f(\mathbf{c})|^2 &\le \|h_{\{1\},c}-K(\cdot,\mathbf{c})\|_{\mathscr{H}(K)}^2\\ &= \|g-K_1(\cdot,c)\|_{\mathscr{H}(K_1)}^2\sum_{1\in u}\gamma_u\,(K_1(c,c))^{|u|-1} = \kappa(c)\,\gamma_1\,r_{\{1\},c}^2.\end{aligned}$$

Let $f \in B(K_{\{1\}})$. Then $r_{\{1\},v}f \in \Psi_{\{1\},c}(F)$ due to Lemma 5.1, and we obtain

$$|I(f)-f(\mathbf{c})|^2 \le \gamma_1\,\kappa(c).$$

Hence $\kappa(c) \ge c/\gamma_1$ follows from assumption (ii), and Lemma 5.3 implies

$$b_{v,c}^2 \ge A(v,c) \ge \sum_{j\notin v}\|g_{\{j\}}-\gamma_j k_{\{j\}}(\cdot,\mathbf{c})\|_{\mathscr{H}(K)}^2 = \kappa(c)\sum_{j\notin v}\gamma_j \ge c/\gamma_1\sum_{j\notin v}\gamma_j \succeq |v|^{-2q}.$$

We conclude that

$$e^2_{ran}(\hat{I},F) \succeq \frac{|v|^{-2q}}{2\,(1+2|v|^{-2q}+2\|h\|^2_{\mathscr{H}(K)})} \succeq |v|^{-2q}.$$

On the other hand we have

$$e^2_{ran}(\hat{I},F) \geq e^2_{ran}(\hat{I},B(K_{\{1\}})) \geq c\,((N+1)/|v|)^{-2p}$$

due to assumption (ii). It remains to observe that

$$((N+1)/|v|)^{-2p} + |v|^{-2q} \succeq N^{-\frac{pq}{p+q}}.$$

5.2.3 Examples. We apply Theorem 5.1 and Theorem 5.2 in the case of ρ_1 being the uniform distribution on $D = [0,1]$ and for the kernels given by (3.16) and (3.20).

First, we consider the kernel K_1 given by (3.16), which satisfies assumption (A2a). For integration of functions $f : [0,1]^d \to \mathbb{R}$ we employ scrambled quasi-Monte Carlo rules. Scrambling, which is a randomization technique that preserves good discrepancy properties of point sets, was introduced by [Owe97b]. Here we rely on a result from [YH05], which has analyzed randomized quadrature formulas

$$\hat{I}_{b,m,1:d}(f) = \frac{1}{b^m}\sum_{i=1}^{b^m} f(X_i)$$

that use base b scrambling of a Niederreiter (t,m,s)-net in base b. In particular, $\hat{I}_{b,m,1:d}$ is unbiased for every integrable function f. Henceforth we fix b and we choose any $c \in [0,1]$. The methods

$$\hat{I}_{n,s,c} = \hat{I}_{b,\lfloor \log_b(n) \rfloor,1:d} \circ \Psi_{1:d,c} \tag{5.7}$$

with $n,d \in \mathbb{N}$ are called scrambled QMC rules. Note that $\hat{I}_{n,d,c}$ satisfies the cost bound in assumption (iii) of Theorem 5.1.

Assume that

$$\sum_{j=1}^{\infty} \gamma_j\,(j\log j)^3 < \infty. \tag{5.8}$$

Then for every $\varepsilon > 0$ there exists a constant $c_\varepsilon > 0$ such that the scrambled QMC rules $\hat{I}_{n,s,c}$ satisfy

$$e_{\mathrm{ran}}(\hat{I}_{n,s,c}, B_{r_{1:d,c}}(K_{1:d})) \leq c_\varepsilon \cdot n^{-(3/2-\varepsilon)} \tag{5.9}$$

for every $n \in \mathbb{N}$ and every dimension s, see [YH05, Thm. 4.(i)].

Corollary 5.1 *Assume that K_1 is given by* (3.16)*. Let $\varepsilon > 0$, and let assumption (ii) from Theorem 5.1 be satisfied with $q > 3/2$. Choose*

$$n_N \asymp N^{\frac{2q}{2q+3-\varepsilon}}$$

and

$$d_N \asymp N^{\frac{3-q}{2q+3-\varepsilon}}$$

for $N \in \mathbb{N}$. Then, for $\hat{I}_N = \hat{I}_{n_N,d_N,c}$,

$$e_{ran}(\hat{I}_N, F) \preceq N^{-\frac{q(3-\varepsilon)}{2q+3-\varepsilon}}$$

and

$$C^{\mathrm{fix}}(\hat{I}_N, F) \preceq N.$$

Proof 5.7 *Apply Theorem 5.1 with $c = c_{\varepsilon/2}$ according to* (5.9) *and $p = 3/2 - \varepsilon/2$, and note that $\hat{I}_N$ uses*

$$b^{\lfloor \log_b(n_N) \rfloor} \asymp n_N$$

function evaluations in $\mathscr{X}_{1:d_N,c}$ and $n_N \cdot d_N \asymp N$.

Next we turn to K_1 given by (3.20), which satisfies assumption (A2b) with $c^* = 0$. Consider the classical Monte Carlo method $\hat{I}_{n,1:d}$ for integration of functions $f : [0,1]^{1:d} \to \mathbb{R}$, i.e.,

$$\hat{I}_{n,1:d}(f) = \frac{1}{n} \sum_{i=1}^{n} f(X_i),$$

where $X_1, \dots, X_n$ are independent and uniformly distributed on $[0,1]$. The methods

$$\hat{I}_{n,d,0} = \hat{I}_{n,1:d} \circ \Psi_{1:d,0} \tag{5.10}$$

clearly satisfy the cost bound in assumption (iii) of Theorem 5.1. From [SW02] or [Was04, Theorem. 1.1] we infer that there exists a constant $c_0 > 0$ such that

$$e_{\mathrm{ran}}(\hat{I}_{n,d,0}, B(K_{1:d})) \leq c_0\, n^{-1/2} \sum_{j=1}^{d} \gamma_j \tag{5.11}$$

holds for all $n, s \in \mathbb{N}$.

Henceforth, we refer to the methods $\hat{I}_{n,d,0}$ as classical MC rules.

Corollary 5.2 *Assume that K_1 is given by* (3.20)*, and let assumption (ii) from Theorem 5.1 be satisfied. Choose*

$$n_N \asymp N^{\frac{2q}{2q+1}}$$

and

$$d_N \asymp N^{\frac{1}{2q+1}}$$

for $N \in \mathbb{N}$. Then the sequence of classical MC rules $\hat{I}_N = \hat{I}_{n_N,d_N,0}$ satisfies

$$e_{ran}(\hat{I}_N, F) \preceq N^{-\frac{q}{2q+1}}$$

and

$$C^{\mathrm{fix}}(\hat{I}_N, F) \preceq N.$$

Proof 5.8 *Apply Theorem 5.1 with $p = 1/2$ according to* (5.11)*.*

Corollary 5.3 *Assume that K_1 is given by* (3.16) *or by* (3.20)*, and let the assumption (i) from Theorem 5.2 be satisfied with $q > 0$. Then*

$$e^{\mathrm{fix}}(N, F) \succeq N^{-\frac{3q}{2q+3}}.$$

Proof 5.9 *For both kernels, the Sobolev space $W_2^1([0,1])$ is continuously embedded in the space $\mathscr{H}(K_{\{1\}})$, see Section 3.3, and the minimal errors on $W_2^1([0,1])$ are of the order $p = 3/2$, see [Nov88, Sec. 2.2.9]. Hence the result follows from Theorem 5.2.*

Remark 5.2 *Obviously Corollaries 5.1 and 5.2 provide upper bounds for the respective minimal errors $e_{N,\mathrm{fix}}(F)$, while lower bounds are provided by Corollary 5.3. In order to slightly simplify the results we define*

$$\lambda^{\mathrm{fix}} = \sup\{\chi > 0 : \sup_{N\in\mathbb{N}} e^{\mathrm{fix}}(N,F)\cdot N^{\chi} < \infty\}.$$

If K_1 is given by (3.16) *and $\gamma_j \asymp j^{-2q-1}$ with $q > 1/4$, then*

$$\lambda^{\mathrm{fix}} = \frac{3q}{2q+3}.$$

Clearly, $\lim_{q\to 3/2+} \lambda^{\mathrm{fix}} = 3/4$ and $\lim_{q\to\infty} \lambda^{\mathrm{fix}} = 3/2$. In particular, Theorem 5.1 and 5.2 lead to sharp bounds for the minimal error and scrambled QMC rules with appropriately chosen dimensions are almost optimal in the fixed subspace model.

In the case that K_1 is given by (3.20) *and $\gamma_j \asymp j^{-2q-1}$ with $q > 0$ we only get*

$$\frac{q}{2q+1} \le \lambda^{\mathrm{fix}} \le \frac{3q}{2q+3}$$

from Corollaries 5.2 and 5.3. A better lower bound

$$\lambda^{\mathrm{fix}} \ge \begin{cases} q(2q+1)/(4q+1), & \text{if } 0 < q < 1/2, \\ q/(q+1), & \text{if } q \ge 1/2, \end{cases}$$

is due to [KSWW10], and we stress that this bound is already achieved by suitable deterministic algorithms. It is unknown to us whether the latter bound can further be improved if the classical MC rule is replaced by a different randomized algorithm in Corollary 5.2.

5.3 The Error Analysis of the Multi-Level Algorithm

The analysis of variable subspace sampling is motivated by the multi-level approach to infinite-dimensional integration. The latter is based on a sequence of finite-dimensional affine subspaces

$$\mathscr{X}_{v_1,c} \subset \cdots \subset \mathscr{X}_{v_L,c} \tag{5.12}$$

with a point $c \in D$ and an increasing sequence

$$v_1 \subset \cdots \subset v_L$$

of (finite) non-empty subsets of $\mathbb{N}$. For the finite-dimensional integral $I(\Psi_{v_L,c}f)$, which serves as an approximation to $I(f)$ as in Section 4.4, we have

$$I(\Psi_{v_L,c}f) = \sum_{\ell=1}^{L} I(\Psi_{v_\ell,c}f - \Psi_{v_{\ell-1},c}f),$$

where

$$\Psi_{v_0,c}f = 0.$$

In the multi-level approach each of the integrals $I(\Psi_{v_\ell,c}f - \Psi_{v_{\ell-1},c}f)$ is approximated separately by means of independent randomized algorithms, and sampling of f in $\mathscr{X}_{v_\ell,c}$ is used at level ℓ. Clearly, the cost per evaluation of f is increasing with ℓ. Provided that the error for integration of $\Psi_{v_\ell,c}f - \Psi_{v_{\ell-1},c}f$ is decreasing with ℓ at a certain rate, we properly balance these effects.

Remark 5.3 *Consider an increasing sequence of sets $v_\ell \subset \mathbb{N}$ with $\bigcup_{\ell\in\mathbb{N}} v_\ell = \mathbb{N}$. Since*

$$\lim_{\ell\to\infty} \|f - \Psi_{v_\ell,c}f\|_{\mathscr{H}(K)} = 0$$

for every $f \in \mathscr{H}(K)$, which is easily verified, we have strong convergence of $\Psi_{v_\ell,c} - \Psi_{v_{\ell-1},c}$ towards zero. However,

$$\inf_{\ell\in\mathbb{N}} \sup_{f\in F} \|\Psi_{v_\ell,c}f - \Psi_{v_{\ell-1},c}f\|_{\mathscr{H}(K)} > 0.$$

The latter obviously holds true in the case (A2b) with $c^* = c$, since $\Psi_{v,c}$ is the orthogonal projection onto $\mathscr{H}(K_v)$ in this case. To cover the general case we take $y \in D$ such that $K_1(y,y) > 0$. Let $d \in \mathbb{N}$. Put $f(\mathbf{x}) = \sqrt{\gamma_d} K_1(x_d, y)$. Then $f \in \mathscr{H}_{\{d\}}$ with $\|f\|_{\mathscr{H}(K_{1:d})} = \sqrt{K_1(y,y)}$ and $\Psi_{1:d,c} f = f$. Moreover, $\Psi_{1:d-1,c}\Psi_{1:d,c} f \in \mathscr{H}_{\emptyset}$, so that

$$\left\|\Psi_{1:d,c} f - \Psi_{1:d-1,c} f\right\|^2_{\mathscr{H}(K_{1:d})} = \left\|\Psi_{1:d,c} f\right\|^2_{\mathscr{H}(K_{1:d})} + \left\|\Psi_{1:d-1,c} f\right\|^2_{\mathscr{H}(K_{1:d})} = K_1(y,y) + \gamma_d \left(K_1(c,y)\right)^2.$$

We conclude that $\sup_{f\in F} \left\|\Psi_{1:d,c} f - \Psi_{1:d-1,c} f\right\|_{\mathscr{H}(K_{1:d})}$ *does not converge to zero as* $d \to \infty$.

Because of Remark 5.3 we take another sequence of real numbers γ'_j that satisfies

(A4′) $\gamma'_1 \geq \gamma'_2 \geq \cdots > 0$, $\sum_{j=1}^{\infty} \gamma'_j < \infty$, and $\gamma_j / \gamma'_j \leq 1$,

and we define

$$\gamma'_u = \prod_{j\in u} \gamma'_j$$

for every u. The associated kernels are denoted by K', etc., and Lemma 3.6 implies that $\mathscr{H}(K) \subseteq \mathscr{H}(K')$ with

$$\|f\|_{\mathscr{H}(K')} \leq \|f\|_{\mathscr{H}(K)}, \qquad f \in \mathscr{H}(K).$$

5.3.1 Preliminaries. Fix $c \in D$ and let $v \subset w \subset \mathbb{N}$. Recall that $\Psi_{v,c} f \in \mathscr{H}(K_v)$ for every $f \in \mathscr{H}(K)$ by Lemma 5.1. We establish estimates for

$$\Psi_{w,c} f - \Psi_{v,c} f = (\mathrm{id} - \Psi_{v,c})(\Psi_{w,c} f) \in \mathscr{H}(K_w), \tag{5.13}$$

where we consider the norm $\|\cdot\|_{\mathscr{H}(K'_w)}$.

Lemma 5.5 *We have*

$$\sup_{f\in F} \left\|\Psi_{w,c} f - \Psi_{v,c} f\right\|_{\mathscr{H}(K'_w)} \asymp \sup_{f\in B(K_w)} \left\|f - \Psi_{v,c} f\right\|_{\mathscr{H}(K'_w)}$$

uniformly in v and w with $v \subset w$.

Proof 5.10 *Use Lemma 5.1 together with* (5.13)*.*

For the impact of $\Psi_{v,c}$ on each of the terms in an orthogonal decomposition (3.7) the following holds true.

Lemma 5.6 *For $f \in \mathscr{H}_u$ we have*

$$\Psi_{v,c} f \in \mathscr{H}_{u\cap v}$$

and

$$\left\|\Psi_{v,c} f\right\|_{\mathscr{H}(k_{u\cap v})} \leq (K_1(c,c))^{|u\setminus v|/2} \left\|f\right\|_{\mathscr{H}(k_u)}.$$

Moreover, if $u \subseteq v$ then $\Psi_{v,c} f = f$.

Proof 5.11 *Let $f \in \mathscr{H}_u$. Then $\Psi_{v,c} f = \Psi_{u\cap v,c} f$ due to Lemma 3.2, and in particular $\Psi_{v,c} f = \Psi_{u,c} f = f$ in the case $u \subseteq v$. Put*

$$J(\mathbf{x},\mathbf{y}) = (K_1(c,c))^{|u\setminus v|} \prod_{j\in u\cap v} K_1(x_j,y_j) = (K_1(c,c))^{|u\setminus v|}\, k_{u\cap v}(\mathbf{x},\mathbf{y}).$$

We get $\Psi_{u\cap v,c} f \in \mathscr{H}(J) \subseteq \mathscr{H}_{u\cap v}$ and a norm estimate as claimed from Lemma A.1 from the Appendix.

Lemma 5.7 *Let $f \in \mathscr{H}(K_w)$. If K_1 satisfies (A2b) with $c^* = c$ or if $|w \setminus v| = 1$, then*

$$\left\|f - \Psi_{v,c} f\right\|^2_{\mathscr{H}(K'_w)} \leq (1 + \gamma'_1 K_1(c,c)) \cdot \sum_{u\subseteq w, u\setminus v\neq\emptyset} (\gamma'_u)^{-1} \left\|f_u\right\|^2_{\mathscr{H}(k_u)}.$$

Proof 5.12 *Let $f = \sum_{u\subseteq w} f_u$ with $f_u \in \mathscr{H}_u$, see Lemma 3.4. Use Lemma 5.6 to obtain*

$$f - \Psi_{v,c} f = \sum_{u\subseteq w, u\setminus v\neq\emptyset} \left(f_u - \Psi_{u\cap v,c} f_u\right)$$

and

$$\begin{aligned}\left\|f - \Psi_{v,c} f\right\|^2_{\mathscr{H}(K'_w)} &= \sum_{u\subseteq w, u\setminus v\neq\emptyset} \sum_{u'\subseteq w, u'\setminus v\neq\emptyset} \left\langle f_u - \Psi_{u\cap v,c} f_u, f_{u'} - \Psi_{u'\cap v,c} f_{u'}\right\rangle_{\mathscr{H}(K'_w)} \\ &= \sum_{u\subseteq w, u\setminus v\neq\emptyset} (\gamma'_u)^{-1} \left\|f_u\right\|^2_{\mathscr{H}(k_u)} + \sum_{(u,u')\in M} (\gamma'_{u\cap v})^{-1} \left\langle \Psi_{u\cap v,c} f_u, \Psi_{u'\cap v,c} f_{u'}\right\rangle_{\mathscr{H}(k_{u\cap v})}\end{aligned}$$

with

$$M = \{(u,u') : u,u' \subseteq w, u \setminus v \neq \emptyset, u' \setminus v \neq \emptyset, u \cap v = u' \cap v\}.$$

Assume that K_1 satisfies (A2b) with $c^* = c$. Then $\Psi_{u\cap v,c}$ is the orthogonal projection onto $\mathscr{H}(K_{u\cap v})$, and we have $\Psi_{u\cap v,c} f_u = 0$ for every $u \subseteq w$ with $u \setminus v \neq \emptyset$.

On the other hand, if $|w \setminus v| = \{\ell\}$ with $\ell \in \mathbb{N}$ then $M = \{(u,u) : \ell \in u \subseteq w\}$, and it remains to observe that

$$(\gamma'_{u\cap v})^{-1} \left\|\Psi_{u\cap v,c} f_u\right\|^2_{\mathscr{H}(k_{u\cap v})} \leq \gamma'_1 (\gamma'_u)^{-1} K_1(c,c) \|f_u\|^2_{\mathscr{H}(k_u)}$$

due to Lemma 5.6 and (3.4).

Theorem 5.3 *Assume that K_1 satisfies (A2b) with $c^* = c$ or that $|w \setminus v| = 1$. We have*

$$\sup_{f\in F} \left\|\Psi_{w,c} f - \Psi_{v,c} f\right\|_{\mathscr{H}(K'_w)} \preceq \max_{j\in w\setminus v} \sqrt{\gamma_j/\gamma'_j}$$

uniformly in v and w.

Proof 5.13 *Put $c = 1 + \gamma'_1 K_1(c,c)$ and use Lemma 5.7 to obtain*

$$\left\|f - \Psi_{v,c} f\right\|^2_{\mathscr{H}(K'_w)} \leq c \sum_{u\subseteq w, u\setminus v\neq\emptyset} \frac{\gamma_u}{\gamma'_u} \gamma_u^{-1} \|f_u\|^2_{\mathscr{H}(k_u)} \leq c \max_{j\in w\setminus v} \frac{\gamma_j}{\gamma'_j} \cdot \|f\|^2_{\mathscr{H}(K_w)}$$

for $f \in \mathscr{H}(K_w)$. It remains to apply Lemma 5.5.

We do not know whether a result similar to the estimate from Theorem 5.3 is valid under the assumption (A2a) if $|w \setminus v|$ is large.

5.3.2 Upper bounds for multi-level algorithms. We consider an independent family of unbiased randomized algorithms $\hat{I}_{n,1:d}$ for finite-dimensional integration on $D^{1:d}$, and for the construction of multi-level methods we take $c \in D$ and we employ the corresponding independent randomized algorithms $\hat{I}_{n,d,c} = \hat{I}_{n,1:d} \circ \Psi_{1:d,c}$ for infinite-dimensional integration, see (5.3).

For $L \in \mathbb{N}$ and two sequences $n_1, \dots, n_L$ and $d_1, \dots, d_L \in \mathbb{N}$ of positive integers with $d_\ell < d_{\ell+1}$ we put

$$\hat{I}_\ell = \hat{I}_{n_\ell, d_\ell, c}$$

as well as

$$\Psi_\ell = \Psi_{1:d_\ell, c}$$

and we define a multi-level algorithm by

$$\hat{I}(f) = \sum_{\ell=1}^{L} \hat{I}_\ell(f - \Psi_{\ell-1} f), \tag{5.14}$$

where

$$\Psi_0 = 0.$$

Note that

$$\hat{I}_\ell(f - \Psi_{\ell-1} f) = \hat{I}_\ell(\Psi_\ell f - \Psi_{\ell-1} f).$$

Hence $\hat{I}$ uses variable subspace sampling based on the subspaces (5.12) with $v_\ell = 1 : d_\ell$.

In the analysis of the cost of $\hat{I}$ we accordingly take $C = C^{\mathrm{var}}_{\mathbf{v}}$ with $\mathbf{v} = (v_\ell)_{\ell \in \mathbb{N}}$, see (5.2). $\hat{I}_\ell$ is based on function values at points from the subspace $\mathscr{X}_{v_\ell, c}$, see (5.1), i.e., every function evaluation is charged with cost $d_\ell = \dim(\mathscr{X}_{v_\ell, c})$. As in Section 5.2.2, $\hat{I}_{n,1:d}$ typically is an unbiased randomized quadrature formula with n evaluations, and then we clearly have

$$C^{\mathrm{var}}(\hat{I}, F) \le 2 \cdot \sum_{\ell=1}^{L} n_\ell \cdot d_\ell.$$

To cover the general case we put

$$r^* = 2\, r_{v_1, c},$$

see Lemma 5.1 and Remark 5.1

Lemma 5.8 *For the cost of $\hat{I}$ we have*

$$C^{\mathrm{var}}(\hat{I}, F) \le 2 \cdot \sum_{\ell=1}^{L} \sup_{f \in B_{r^*}(K_{v_\ell})} \mathbb{E}(C(\hat{I}_\ell, f))$$

in the variable subspace model.

Proof 5.14 *Note that*

$$C^{\mathrm{var}}(\hat{I},F) \le \sup_{f\in F} \mathbb{E}(C(\hat{I},f)) \le \sum_{\ell=1}^{L} \sup_{f\in F} \mathbb{E}(C(\hat{I}_\ell \circ (\Psi_\ell - \Psi_{\ell-1}),f)).$$

Moreover,

$$C_{C_\ell}(\hat{I}_\ell \circ (\Psi_\ell - \Psi_{\ell-1}),f) \le 2 \cdot C_{C_\ell}(\hat{I}_\ell, \Psi_\ell f - \Psi_{\ell-1} f)$$

and $\Psi_\ell f - \Psi_{\ell-1} f \in B_{r^*}(K_{v_\ell})$ *for* $f \in F$, *see Lemma 5.1.*

For the error of $\hat{I}$ we obtain

$$\mathbb{E}(I(f) - \hat{I}(f))^2 = \big(I(f) - I(\Psi_L f)\big)^2 + \mathrm{Var}(\hat{I}(f)) \tag{5.15}$$

with

$$\mathrm{Var}(\hat{I}(f)) = \sum_{\ell=1}^{L} \mathrm{Var}\left(\hat{I}_\ell(f - \Psi_{\ell-1} f)\right). \tag{5.16}$$

If the building blocks $\hat{I}_{n,1:d}$ are unbiased randomized quadrature formulas, then, in the error analysis for the multi-level algorithm, we might assume that an upper bound for the maximal error of $\hat{I}_{n,1:d}$ is available that only depends on n. However, the maximal error is taken on the unit ball in $\mathscr{H}(K'_{1:d})$ instead of $\mathscr{H}(K_{1:d})$. In general, we have to consider the ball of radius r^* in $\mathscr{H}(K_{1:d})$ and to provide an error bound in terms of the norm in $\mathscr{H}(K'_{1:d})$.

We first study the case of a kernel that satisfies (A2a), where we assume that $d_{\ell+1} = d_\ell + 1$ because of the limitation in Theorem 5.3.

Theorem 5.4 *Let* $c \in D$, *and assume that*

(i) K_1 *satisfies* (A2a),

(ii) $\gamma_j \preceq j^{-2q-1}$ *with* $q > 0$,

(iii) $\gamma'_j \asymp j^{-2q'-1}$ *with* $0 < q' < q$,

(iv) there exist $p, c > 0$ such that

$$\mathrm{Var}(\hat{I}_{n,d,c}(f)) \le c \, \|f\|^2_{\mathscr{H}(K'_{1:d})} \, n^{-2p}$$

and

$$\mathbb{E}(C(\hat{I}_{n,d,c}, f)) \le c \, n \, d$$

for all $n, d \in \mathbb{N}$ and every $f \in B_{r^*}(K_{1:d})$.

Put

$$\rho_1 = \frac{q}{p}, \qquad \rho_2 = \frac{2q - 2q' - 1}{2p}.$$

For $N \ge 2$ we choose

$$L = \begin{cases} \left\lceil N^{\frac{1}{\rho_1}} \right\rceil, & \text{if } \rho_2 > 2, \\ \left\lceil (N/\ln N)^{\frac{1}{\rho_1}} \right\rceil, & \text{if } \rho_2 = 2, \\ \left\lceil N^{\frac{1}{\rho_1 + 2 - \rho_2}} \right\rceil, & \text{if } \rho_2 < 2, \end{cases} \tag{5.17}$$

as well as

$$d_\ell = \ell \tag{5.18}$$

and

$$n_\ell = \lceil d_\ell^{-\rho_2} L^{\rho_1} \rceil = \begin{cases} \lceil \ell^{-\rho_2} N \rceil, & \text{if } \rho_2 > 2, \\ \lceil \ell^{-\rho_2} N / \ln N \rceil, & \text{if } \rho_2 = 2, \\ \left\lceil \ell^{-\rho_2} N^{\frac{\rho_1}{\rho_1 + 2 - \rho_2}} \right\rceil, & \text{if } \rho_2 < 2, \end{cases} \tag{5.19}$$

for $\ell = 1, \dots, L$. Then the corresponding multi-level algorithm $\hat{I}_N$ given by (5.14) *satisfies*

$$e_{ran}(\hat{I}_N, F) \preceq (\ln N)^{1/2} \cdot \begin{cases} N^{-p}, & \text{if } q - q' > 2p + 1/2, \\ (N/\ln N)^{-p}, & \text{if } q - q' = 2q + 1/2, \\ N^{-p \frac{2q}{2q' + 4p - 1}}, & \text{if } q - q' < 2q + 1/2, \end{cases}$$

as well as

$$C^{\mathrm{var}}(\hat{I}_N, F) \preceq N.$$

Proof 5.15 *Throughout the following we do not indicate the dependence of the numbers d_ℓ, n_ℓ and L on N. Assumptions (i), (iii), and (iv) together with Theorem 5.3 yield*

$$\mathrm{Var}\Big(\hat{I}_{n_\ell,d_\ell,c}(f-\Psi_{1:d_{\ell-1},c}f)\Big) \le c\,\|\Psi_{1:d_\ell,c}(f-\Psi_{1:d_{\ell-1},c}f)\|^2_{\mathscr{H}(K'_{1:d_\ell})}\cdot n_\ell^{-2p} \preceq d_\ell^{-2(q-q')}\cdot n_\ell^{-2p}$$

for every $f\in F$. Use assumptions (i) and (ii) together with Lemma 5.4 to get

$$b^2_{1:d_L,c} \preceq d_L^{-2q}.$$

Hence, by (5.15) *and* (5.16)*,*

$$e^2_{ran}(\hat{I}_N,F) \preceq \sum_{\ell=1}^{L} d_\ell^{-2(q-q')}\,n_\ell^{-2p} + d_L^{-2q},$$

and Lemma 5.8 together with assumption (iv) implies

$$C^{\mathrm{var}}(\hat{I}_N,F) \preceq \sum_{\ell=1}^{L} n_\ell\cdot d_\ell.$$

Consequently,

$$\begin{aligned} e^2_{ran}(\hat{I}_N,F) &\preceq \sum_{\ell=1}^{L} \ell^{-2(q-q')}\,n_\ell^{-2p} + L^{-2q} \\ &\asymp L^{-2p\rho_1}\sum_{\ell=1}^{L}\ell^{2p\rho_2-2(q-q')} \asymp L^{-2p\rho_1}\,(\ln L). \end{aligned}$$

Furthermore, since $\rho_1>\rho_2$,

$$\begin{aligned} C^{\mathrm{var}}(\hat{I}_N,F) &\preceq L^2 + L^{\rho_1}\sum_{\ell=1}^{L}\ell^{1-\rho_2} \\ &\preceq L^2 + \begin{cases} L^{\rho_1}, & \text{if } \rho_2>2,\\ L^{\rho_1}(\ln L), & \text{if } \rho_2=2,\\ L^{\rho_1+2-\rho_2}, & \text{if } \rho_2<2, \end{cases} \quad \asymp \begin{cases} L^{\rho_1}, & \text{if } \rho_2>2,\\ L^{\rho_1}(\ln L), & \text{if } \rho_2=2,\\ L^{\rho_1+2-\rho_2}, & \text{if } \rho_2<2, \end{cases} \end{aligned}$$

and it remains to observe that $\ln L \asymp \ln N$.

Now we consider the anchored case, where a better estimate, compared to the one from Theorem 5.4, is obtained, since we may analyze any progression of the dimensions d_ℓ.

Theorem 5.5 *Let $c \in D$. Assume that K_1 satisfies* (A2b) *with $c^* = c$ and that the assumptions (ii)–(iv) from Theorem 5.4 are satisfied. Put*

$$\rho_1 = \frac{q}{p}, \qquad \rho_3 = \frac{q - q'}{p}.$$

For $N \geq 2$ we choose

$$L = \begin{cases} \left\lceil \ln N / \rho_1 \right\rceil, & \text{if } \rho_3 \geq 1, \\ \left\lceil \ln N / (\rho_1 + 1 - \rho_3) \right\rceil, & \text{if } \rho_3 < 1, \end{cases} \tag{5.20}$$

as well as

$$d_\ell = 2^\ell \tag{5.21}$$

and

$$n_\ell = \begin{cases} \lceil d_\ell^{-\rho_3} d_L^{\rho_1} \rceil, & \text{if } \rho_3 \neq 1, \\ \lceil d_\ell^{-1} d_L^{\rho_1} / L \rceil, & \text{if } \rho_3 = 1, \end{cases} \tag{5.22}$$

for $\ell = 1, \ldots, L$. Then the corresponding multi-level algorithm $\hat{I}_N$ given by (5.14) *satisfies*

$$e_{ran}(\hat{I}_N, F) \preceq (\ln N)^{1/2} \cdot \begin{cases} N^{-p}, & \text{if } q - q' > p, \\ (N / \ln N)^{-p}, & \text{if } q - q' = p, \\ N^{-p \frac{q}{q'+p}}, & \text{if } q - q' = p, \end{cases}$$

as well as

$$C^{\text{var}}(\hat{I}_N, F) \preceq N.$$

Proof 5.16 *We proceed as in the proof of Theorem 5.4 to obtain*

$$e_{ran}^2(\hat{I}_N, F) \preceq \sum_{\ell=1}^{L} d_\ell^{-2(q-q')} n_\ell^{-2p} + d_L^{-2q}$$

as well as

$$C^{\text{var}}(\hat{I}_N, F) \preceq \sum_{\ell=1}^{L} n_\ell \cdot d_\ell.$$

Assume $\rho_3 \neq 1$. *Then* $d_\ell^{-2(q-q')} \cdot n_\ell^{-2p} \leq d_L^{-2q}$ *and consequently,*

$$e_{ran}^2(\hat{I}_N, F) \preceq (L+1) \cdot d_L^{-2q} \preceq (\ln N) \cdot \begin{cases} N^{-2p}, & \text{if } \rho_3 > 1, \\ N^{\frac{-2p\rho_1}{\rho_1+1-\rho_3}}, & \text{if } \rho_3 < 1. \end{cases}$$

Furthermore, we have $d_\ell \cdot n_\ell \leq d_\ell^{1-\rho_3} \cdot d_L^{\rho_1} + d_\ell$ *and* $\rho_1 > \rho_3$, *which yields*

$$C^{\text{var}}(\hat{I}_N, F) \preceq \sum_{\ell=1}^{L} \left(d_\ell^{1-\rho_3} \cdot d_L^{\rho_1} + d_\ell \right) \preceq d_L^{\rho_1} + d_L \preceq d_L^{\rho_1} \preceq N$$

in the case $\rho_3 > 1$, *and*

$$C^{\text{var}}(\hat{I}_N, F) \preceq d_L^{\rho_1+1-\rho_3} + d_L \preceq d_L^{\rho_1+1-\rho_3} \preceq N$$

in the case $\rho_3 < 1$.

Now consider the case $\rho_3 = 1$. *Then* $d_\ell^{-2(q-q')} \cdot n_\ell^{-2p} \leq d_L^{-2q} \cdot L^{2p}$ *and we obtain*

$$e_{ran}^2(\hat{I}_N, F) \preceq (L^{2p+1} + 1) \cdot d_L^{-2q} \preceq (\ln N)^{2p+1} \cdot N^{-2p}.$$

Moreover, $d_\ell \cdot n_\ell \leq d_L^{\rho_1}/L + d_\ell$ *and* $\rho_1 \geq 1$, *and we conclude*

$$C^{\text{var}}(\hat{I}_N, F) \preceq \sum_{\ell=1}^{L} \left(d_L^{\rho_1}/L + d_\ell \right) \preceq d_L^{\rho_1} \preceq N,$$

which finishes the proof.

5.4 Examples

As in Section 5.2.3 we study the case of ρ being the uniform distribution on $D = [0,1]$ and K_1 given by (3.16) or by (3.20). The building blocks of the multi-level algorithms are the ones that we have already considered in Section 5.2.3, namely, scrambled QMC rules for the kernel (3.16) and classical MC rules for the kernel (3.20).

Corollary 5.4 *Assume that* K_1 *is given by* (3.16) *and that*

$$\gamma_j \asymp j^{-2q-1}$$

for any $q > 3/2$. *Let* $0 < \varepsilon < \min(6, 2q-3)$ *and put*

$$\rho_1 = \frac{2q}{3-\varepsilon/2}, \qquad \rho_2 = \frac{2q-4-\varepsilon}{3-\varepsilon/2}.$$

Choose L, d_ℓ *and* n_ℓ *according to* (5.17), (5.18), *and* (5.19), *respectively, and let* $c \in [0,1]$. *Take the corresponding multi-level algorithm* $\hat{I}_N$ *according to* (5.14) *based on the scrambled QMC rules* $\hat{I}_{n,d,c}$ *provided by* (5.7). *Then*

$$e_{ran}(\hat{I}_N, F) \preceq \begin{cases} N^{-(3-\varepsilon)/2}, & \text{if } q \geq 5, \\ N^{-(3-\varepsilon)\frac{q}{10}}, & \text{if } q < 5, \end{cases}$$

and

$$C^{var}(\hat{I}_N, F) \preceq N.$$

Proof 5.17 *Consider the weights* $\gamma'_j = j^{-(4+\varepsilon)}$ *and apply Theorem 5.4 with the constant* $c = c_{\varepsilon/4}$ *and* $p = 3/2 - \varepsilon/4$ *according to* (5.9) *to obtain* $C^{var}(\hat{I}_N, F) \preceq N$ *as well as*

$$e_{ran}(\hat{I}_N, F) \preceq (\ln N)^{1/2} \cdot \begin{cases} N^{-(3/2-\varepsilon/4)}, & \text{if } q > 5, \\ (N/\ln N)^{-(3/2-\varepsilon/4)}, & \text{if } q = 5, \\ N^{-(3/2-\varepsilon/4)\frac{q}{5}}, & \text{if } q < 5. \end{cases}$$

Clearly, the latter bound implies the error bound in the corollary.

Corollary 5.5 *Assume that* K_1 *is given by* (3.20) *and that*

$$\gamma_j \asymp j^{-2q-1}$$

for any $q > 0$. *Let*

$$\varepsilon \in \begin{cases}]0, 2q[, & \text{if } q \leq 1/2, \\ ]0, 2q-1[, & \text{if } q > q/2, \end{cases}$$

and put

$$\rho_1 = 2q, \qquad \rho_3 = 2q - \varepsilon/2.$$

Choose L, d_ℓ and n_ℓ according to (5.20), (5.21) *and* (5.22)*, respectively. Take the corresponding multi-level algorithm $\hat{I}_N$ according to* (5.14) *based on the classical MC rules $\hat{I}_{n,d,0}$ given by* (5.10)*. Then*

$$e_{ran}(\hat{I}_N, F) \preceq \begin{cases} (\ln N)^{1/2} \cdot N^{-1/2} & \text{if } q > 1/2, \\ N^{-\frac{q}{1+\varepsilon}} & \text{if } q \leq q/2, \end{cases}$$

and

$$C^{\mathrm{var}}(\hat{I}_n, F) \preceq N.$$

Proof 5.18 *Consider the weights $\gamma'_j = j^{-(1+\varepsilon/2)}$ and apply Theorem 5.5 with $c = 0$ and $p = 1/2$ according to* (5.11) *to obtain $C^{\mathrm{var}}(\hat{I}_n, F) \preceq N$ and*

$$e_{ran}(\hat{I}_N, F) \preceq (\ln N)^{1/2} \cdot \begin{cases} N^{-1/2} & \text{if } q > 1/2, \\ N^{-\frac{q}{1+\varepsilon/2}} & \text{if } q \leq 1/2. \end{cases}$$

The latter bound clearly implies the error bound in the corollary.

Remark 5.4 *For both kernels,* (3.16) *and* (3.20)*, a comparison of fixed and variable subspace sampling can be based on the lower bound from Corollary 5.3 and the respective upper bounds from Corollaries 5.2 and 5.4. Similar to Remark 5.2 we take a slightly simplified view and we define*

$$\lambda^{\mathrm{var}} = \sup\{\chi > 0 : \sup_{N \in \mathbb{N}} e^{\mathrm{var}}(N, F) \cdot N^{\chi} < \infty\}.$$

Clearly, $\lambda^{\mathrm{var}} \geq \lambda^{\mathrm{fix}}$.

If K_1 is given by (3.16)*, and $\gamma_j \asymp j^{-2q-1}$ with $q > 3/2$, then*

$$\lambda^{\mathrm{var}} \geq \begin{cases} 3/2, & \text{if } q \geq 5, \\ 3/2 \cdot q/5, & \text{if } 7/2 < q < 5, \\ 3/2 \cdot 2q/(2q+3) & \text{if } 0 < q \leq 7/2. \end{cases}$$

We conclude that variable subspace sampling is superior to fixed subspace sampling (at least) if $q > 7/2$. Moreover, the multi-level algorithm according to Corollary 5.4 is almost optimal (at least) if $q \geq 5$, since a classical result for one-dimensional integration implies $\lambda^{\mathrm{var}} \leq 3/2$, see the proof of Corollary 5.3. For small values of q, however, our analysis of variable subspace sampling suffers from the limitations in Theorem 5.3.

In the case of K_1 given by (3.20) *and $\gamma_j \asymp j^{-2q-1}$ with $q > 0$, we have*

$$\lambda^{\mathrm{var}} \geq \begin{cases} 1/2, & \text{if } q > 1/2, \\ 1/2 \cdot 2q, & \text{if } 0 < q \leq 1/2, \end{cases}$$

which shows that variable subspace sampling is superior to fixed subspace sampling (at least) if $0 < q < 3/4$. A better lower bound

$$\lambda^{\mathrm{var}} \geq \lambda^{\mathrm{fix}} \geq \frac{q}{q+1}, \qquad q > 1,$$

which is due to [KSWW10], was already discussed in Remark 5.2. It would be interesting to know whether suitable multi-level Monte Carlo algorithms outperform deterministic algorithms that use fixed subspace sampling for $q > 1$.

CHAPTER 6

NUMERICAL EXPERIMENTS

In this section, we report numerical results concerning the pricing of different kinds of financial options, introduced in Section 1.1, by means of the single-level and multi-level algorithm described in Section 2.3.1.2 and Section 2.3.1.1. This chapter first illustrates the stochastic models and options used in the numerical experiment, then the results of the single-level and multi-level algorithms.

6.1 Introduction

6.1.1 Geometric Brownian motions. A stochastic process $S(t)$ is a geometric Brownian motion if $\log(S(t))$ is a Brownian motion with initial value $\log(S(0))$. In other words, a geometric Brownian motion is simply the exponential of a Brownian motion $B(t)$. The advantage of the geometric Brownian motion is that it's always positive because the exponential function takes only positive values, which makes it a favorable tool for modeling the stock price and other financial assets. Mathematically, the stochastic process $S(t)$ is specified as

$$\mathrm{d}S(t) = \mu S(t)\,\mathrm{d}t + \sigma S(t)\,\mathrm{d}B(t), \tag{6.1}$$

μ is the drift of $S(t)$, which measures the average growth rate for $S(t)$, σ is the volatility parameter of $S(t)$. By Itô formula,

$$\mathrm{d}\log(S(t)) = (\mu - \frac{1}{2}\sigma^2)\,\mathrm{d}t + \sigma\,\mathrm{d}B(t),$$

and the solution of (6.1) is

$$S(t) = S(0)\exp\left((\mu - \frac{1}{2}\sigma^2)t + \sigma B(t)\right). \tag{6.2}$$

The Black-Scholes-Merton differential equation is a specific geometric Brownian motion which is introduced to value the option. The key property of the Black-Scholes-Merton differential equation is that it does not involve any variables affected by the risk

preferences in investors. Compared to (6.1), μ is replaced by r, where r is the risk-free rate of interest. From [BS73], the stock price dynamics are

$$\mathrm{d}S(t) = rS(t)\,\mathrm{d}t + \sigma S(t)\,\mathrm{d}B(t), \qquad S(t) = S(0)\exp\left((r-\frac{1}{2}\sigma^2)t + \sigma B(t)\right).$$

In the case of the European call option, whose price is given by $\mathbb{E}\left[e^{-rT}\max(S(T)-K,0)\right]$, this expectation can be evaluated explicitly because $B(T)$ is normally distributed. This yields the famous Black-Scholes formula

$$\text{fair price of European call option} = S(0)N(d_1) - Ke^{-rT}N(d_2), \tag{6.3}$$

where N is the cumulative probability distribution function of a standard normal distribution, and

$$\begin{aligned} d_1 &= \frac{\log(S(0)/K) + (r+\sigma^2/2)T}{\sigma\sqrt{T}}, \\ d_2 &= \frac{\log(S(0)/K) + (r-\sigma^2/2)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}. \end{aligned}$$

6.1.2 Path-dependent options. A path-dependent option is an option whose payoff depends on the past history of the underlying asset price and its price at exercise or expiration. The common path-dependent options include American options, Asian options, Barrier options and lookback options. We focus on the Asian options and Barrier options.

Asian options are path-dependent exotic options whose payoffs depend on the history of the stochastic processes of the underlying asset prices via some kinds of average. One such option is the arithmetic mean Asian option, where call and put payoffs are $\max(\bar{S}-K,0)$ and $\max(K-\bar{S},0)$ respectively, where K is a constant and

$$\bar{S} = \frac{1}{m}\sum_{i=1}^{m} S(t_i) \tag{6.4}$$

is the arithmetic average of the underlying asset over the discrete monitoring dates $t_1,\ldots,t_m$. There is no explicit formula for the price of this type of option because the distribution of

$\bar{S}$ is intractable. Another type of Asian options is the geometric mean option, where $\bar{S}$ in (6.4) is defined as

$$\left(\prod_{i=1}^{m} S(t_i)\right)^{1/m}.$$

Such kind of options are useful as test tools for computational algorithms, because the geometric average of joint lognormal random variables is still lognormal. From (6.2),

$$\left(\prod_{i=1}^{m} S(t_i)\right)^{1/m} = S(0)\exp\left((r-\frac{1}{2}\sigma^2)\frac{1}{m}\sum_{i=1}^{m} t_i + \frac{\sigma}{m}\sum_{i=1}^{m} B(t_i)\right).$$

Options on geometric average can be easily priced in closed form using Black-Scholes formula (6.3). For a geometric average option defined on $[0,T]$, it follows from the Black-Scholes formula (6.3) that

$$\bar{S} := \left(\prod_{i=1}^{m} S(t_i)\right)^{1/m} = S(0)\exp\left((\bar{r}-\bar{\sigma}^2/2)\bar{T} + \bar{\sigma}\sqrt{\bar{T}}X\right),$$

where $\bar{T} = T(1+1/m)/2$, $\bar{\sigma}^2 = \sigma^2(2+1/m)/3 \le \sigma^2$, $\bar{r} = r + (\bar{\sigma}^2 - \sigma^2)/2 \le r$, and

$$\begin{aligned} & \text{fair geometric mean option price}(r,\sigma,T) \\ = \;& \text{fair European option price}(\bar{r},\bar{\sigma},\bar{T}) \times \exp(\bar{r}\bar{T} - rT). \end{aligned} \tag{6.5}$$

Barrier options are options where payoffs depend on whether the underlying asset price reaches a preset level during a specific period of time. Barrier options are classified as either knock-out options or knock-in options. Knock-out options expire when the underlying asset price reaches a certain barrier; whereas knock-in options come into existence only when the underlying asset price reach certain levels. A specific example is a down-and-out call with barrier B, strike K, and expiration T with payoff

$$\max(S(T)-K,0) \qquad \text{if} \quad \max_{0\le t\le T} S(t) > B.$$

Barrier options generally do not admit closed form solutions and require approximate computational procedures.

6.1.3 Implementation of Monte Carlo methods in option pricing. Monte Carlo methods are widely used in pricing options. In Section 2.3.1, we reviewed the basic principle and steps of Monte Carlo methods. Here we briefly describe how to use Monte Carlo methods to price options. This section is focusing on the approximation of the arithmetic mean Asian call option.

In order to price an option with Monte Carlo methods, one simulates the stochastic process of the underlying asset prices and then compute the option price for each scenario. By the central limit theorem, the statistical estimate with sufficient simulations yields a good approximation of the option price. Since we are able to simulate the asset price process over the whole life of the option, Monte Carlo type methods can be used to price path-dependent options. However, it is more difficult to price American-style options via Monte Carlo simulations because the option can be exercised early. See [LS01] for a regression based Monte Carlo simulation methodology.

We demonstrate the steps of pricing the arithmetic mean Asian call option defined in (6.4), the value of the option in the risk-neutral setting is

$$\mathbb{E}\left[\exp(-rT)\max(\bar{S}-K,0)\right].$$

Following the outline of Monte Carlo simulations given in Section 2.3.1, we implement a Monte Carlo method for pricing the Asian call option as follows:

1. Produce a random series $\mathbf{x}_i, i = 1,\ldots,n$ with standard normal distribution. Refer to Section 2.2 for the possible choice of low discrepancy sampling.

2. Generate Brownian motions $B(t)$ using approaches introduced in Section 2.1.

3. Compute $S(t_i), i = 1,\ldots m$ according to (6.2).

4. Valuate the arithmetic average $\bar{S}_i$ via (6.4) for each simulation path $i, i = 1,\ldots,n$.

5. Approximate the option price by $\frac{1}{n}\sum_{i=1}^{n}\max\left(\bar{S}_i - K, 0\right)$.

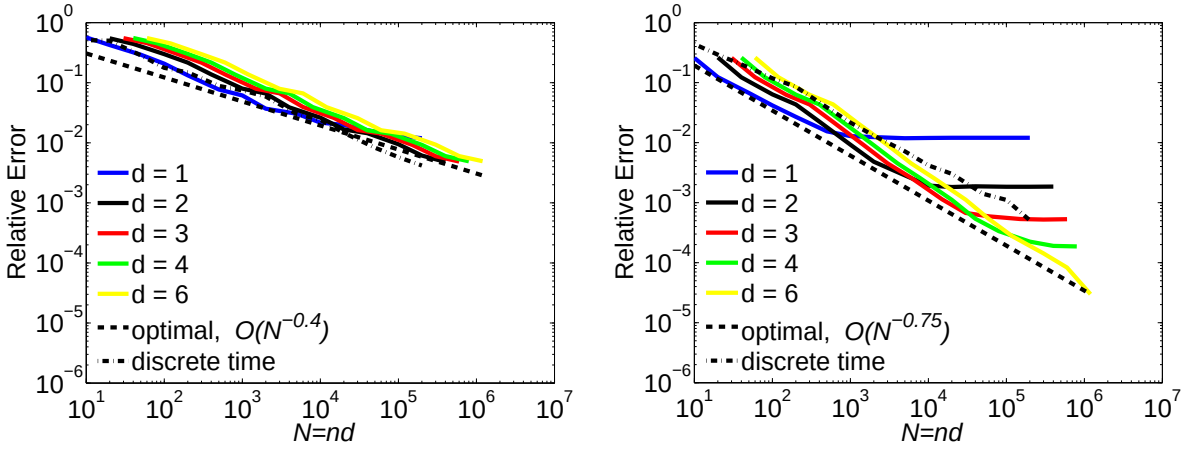
Note that in step 2, the Brownian motion $B(t)$ can be simulated by the discrete time method introduced in Section 2.1.1. To perform this framework, m random numbers are needed to simulate one option payoff, so nm random variables are needed in total of n simulations. The more accurate and efficient ways are the Karhunen-Loève Expansion in Section 2.1.2.1 and the Brownian Bridge method in Section 2.1.2.2. All those method require the sampling of standard normal random variables, where the dimension of problem is decided by the truncation of the infinite-dimensional expansion.

6.2 The Single-Level Algorithm

In this section, we implement the single-level algorithm introduced in Section 2.3.1.1 to price a continuously monitored geometric mean Asian option. As in Section 2.3.1.1, the payoff function of options involves the max function, which typically make the integrand non-smooth. Finding a proper kernel K and the corresponding Hilbert space to match realistic option payoff functions is an open problem, even in the case of discretely monitored options (finite d). However, most kernels used in the analysis of quasi-Monte Carlo algorithm assume moderate smoothness. Follow the setting introduced in Section 2.3.1.1, n is the number of simulation paths for the stock prices and d is the truncated dimension arising from the expansion of Brownian motion $B(t)$. The numerical experiment given here shows how both d and n affect the approximation error. The explicit analytical formula for the geometric mean Asian option makes the exact computation of the error possible, see (6.5).

The parameters of the model are: $S(0) = 100, K = 100, T = 1, r = 0.03, \sigma = 0.3$. The simulation algorithm for this pricing problem is from (2.20). We use the Karhunen-Loève expansion to approximate the Brownian motion $B(t)$. To compare the Karhunen-Loève expansion and the discrete time algorithm for generating Brownian motions, their values at times kT/m, $k = 1, \ldots, m$ are generated by both methods. These values are then

used to approximate the payoff function. A mid-point rule is used to approximate the integral with respect to time above. A choice of $m = 52$ corresponds to weekly sampling and this makes the error of approximating the integral with respect to time negligible compared to the other errors for n up to at least 10^5 and d up to at least 6. This can be seen from Figure 6.1 where the relative RMSE continues to decrease as n increases for $d = 6$. Note that the dimension d corresponds to the space dimension, which represents the truncated dimension for $B(t)$, while m is the time dimension, introduced to approximate the integrand in the payoff function.



(a) Simple Random Sequences

(b) Scrambled Sobol' Sequences

Figure 6.1. The root mean square error for the single-level algorithm approximating the Geometric mean Asian call option.

In both plots, the truncated dimension is chosen to be $d = 1,2,3,4,6$ for the pricing algorithms. The $x-$axis is the computational cost $N = nd$. In the MC case, a pseudo-random sequence is used, while in the QMC case, a scrambled Sobol' sequence is used. At

each stage, R replications are implemented to compute the root mean square error,

$$\begin{aligned}\hat{I} &= \frac{1}{R}\sum_{r=1}^{R}\underbrace{\frac{1}{n}\sum_{i=1}^{n}\text{payoff}\left(\hat{B}_d\left(\mathbf{x}_{i,1:d}^{(r)}\right)\right)}_{\hat{I}_r},\\ \text{RMSE}(\hat{I}) &= \frac{1.96}{\sqrt{R}}\sqrt{\frac{1}{R-1}\sum_{r=1}^{R}\left(\hat{I}-\hat{I}_r\right)^2}.\end{aligned}$$

Here, 1.96 means we take 95% confidence interval. In addition, the discrete time algorithm is implemented with $N = n$ to facilitate the comparison with the Karhunen-Loève expansion method.

For a Karhunen-Loève expansion with fixed d, the relative RMSE using either i.i.d. or scrambled Sobol' sampling initially decreases as N increases, but eventually reaches the $n = \infty$ limit corresponding to the bias. The Sobol' sampling scheme reaches this limit for smaller n (or N) than is the case for i.i.d. sampling because the former is a superior sampling scheme. It is found empirically by the method described at the end of Section 2.3.1.1 that the convergence rate using the optimal choices of n and d with $N = nd$ in the simple random sampling case is approximately $\mathcal{O}(N^{-0.4})$, while in the scrambled Sobol' sequence case, the optimal convergence rate is close to $\mathcal{O}(N^{-0.8})$.

Sobol' sampling combined with the Karhunen-Loève expansion yields a smaller error than Sobol' sampling combined with the time discretization algorithm. This is because the time discretization algorithm requires a Sobol' sequence of dimension equalling the number of discrete times ($m = 52$), whereas the Karhunen-Loève algorithm requires a Sobol' sequence of dimension equalling the number of terms in the expansion, here corresponding to $d \leq 6$. The Karhunen-Loève expansion concentrates the low frequency behavior of the Brownian motion in the early terms of the expansion, where the Sobol' sequence has especially good equi-distribution properties, something that the time discretization algorithm cannot do.

In this numerical experiment, we choose a range of truncated dimension d to iden-

tify the convergence rate of the single-level algorithm. However, how to choose the optimal d and n in practical computation is still an open problem. Following from the main result from Section 4.2, n and d yield the asymptotic relationship given that $N = nd$ and the worst-case error is minimized. In other words, given a tolerance for the approximation and computation constraint, we are able to identify the relationship between n and d, largely due to the possibility of writing the error bound explicitly. For Monte Carlo sampling, the root mean square error can be easily illustrated because the random variables are identically independent distributed. The quasi-Monte Carlo sampling utilizes the low discrepancy sequence with deterministic property, so the explicit formula for the approximation error is untractable. Here we are using the randomized quasi-Monte Carlo sampling to achieve an approximation of the error, which may not be the optimal case.

6.3 The Multi-Level Algorithm

In this section, we demonstrate the implementation of the multi-level algorithm in practice. Numerical results of applying the multi-level algorithm to the geometric mean Asian call option and Barrier call option are presented. The multi-level algorithm combined with the low discrepancy sampling yields a higher convergence order than the single-level algorithm.

First of all, we follow the multi-level algorithm discussed in Section 2.3.1.2. The increasing sequence of truncation dimensions are chosen as $0 = d_0 < d_1 < \cdots < d_{L+1} = \infty$, and sample paths at each level are $n_1 > n_2 > \ldots > n_L$, where L is the level of the algorithm. The option price is modeled as $I = \mathbb{E}[f(X_1, X_2, \ldots)]$, where $(X_1, X_2, \ldots)$ is the standard normal random sequence. As in (2.22),

$$\hat{I}(f) = \sum_{l=1}^{L} \hat{I}_l(f),$$

where

$$\hat{I}_l(f) = \frac{1}{n_l} \sum_{i=1}^{n_l} \left[f(\mathbf{x}_{i,1:d_\ell}^{(l)}, \mathbf{0}) - f(\mathbf{x}_{i,1:d_{\ell-1}}^{(l)}, \mathbf{0}) \right], \quad l = 1, \ldots, L. \tag{6.6}$$

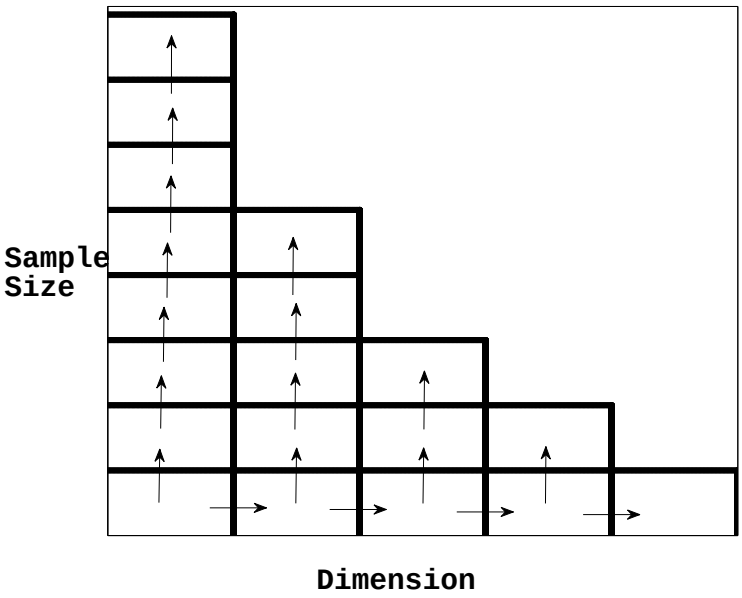


Figure 6.2. From the left to right, search for the optimal L. From the bottom to top, search for the optimal n_l

Here we introduce an adapted algorithm to perform (6.6). Denote $h(\mathbf{x}_{i,1:d_l}) = f(\mathbf{x}^{(l)}_{i,1:d_\ell}, \mathbf{c}) - f(\mathbf{x}^{(l)}_{i,1:d_{\ell-1}}, \mathbf{c})$, then introduce $S_j = \sum_{i=1}^{m_j} h(\mathbf{x}_{i+m_j,1:d_l})$ with $m_j = m_0 2^j, j = 0,1,2,\ldots$. Then, the approximation formula (6.6) can be rewritten as

$$\hat{I}_l = \frac{1}{m_1} S_1 + \frac{1}{m_2}(S_2 - S_1) + \cdots + \frac{1}{m_{J_l}}(S_{J_l} - S_{J_{l-1}}), \quad \text{where } n_l = m_{J_l}. \tag{6.7}$$

The advantage of this adapted algorithm is it allows us to gradually search for the optimal n_l used at each level l. More specifically, we evenly assign the total tolerance of the algorithm, ε, to each level l, and then continuously increase the sample size by the power of 2 at each iteration until the tolerance of the level l approximation is reached. Figure 6.2 describes the flow chart. Starting at level 1, we first use m_1 samples to approximate $\hat{I}_1$, the error is measured in the same way as used in the single-level algorithm, i.e., run R

replications to compute the root mean square error,

$$\begin{aligned} \hat{I}_l &= \frac{1}{R}\sum_{r=1}^{R}\underbrace{\frac{1}{n_l}\sum_{i=1}^{n_l} h(\mathbf{x}_{i,1:d_l}^{(r)})}_{\hat{I}_{l,r}}, \\ \widehat{\mathrm{err}}_{l,n_l} &= \frac{1.96}{\sqrt{R}}\sqrt{\frac{1}{R-1}\sum_{r=1}^{R}\left(\hat{I}_l-\hat{I}_{l,r}\right)^2}. \end{aligned} \tag{6.8}$$

If the error tolerance is not met after the first try, we then increase the sample size to m_2, repeat the above steps until the tolerance is reached. After the first level, the same framework is applied for the next level, and so on. Another important task is to find the optimal L given the tolerance and computational cost N. This algorithm allow us to determine the optimal L given the tolerance ε. It stops when

$$\widehat{\mathrm{err}}_{1,n_1}+\widehat{\mathrm{err}}_{2,n_2}+\cdots+\widehat{\mathrm{err}}_{L,n_L}+\hat{\mu}_L<\varepsilon,$$

and each term in the above inequality yields the same tolerance $\varepsilon/(L+1)$. Hence, the estimator $\hat{\mu}_L$ determines the optimal L.

Even though this algorithm provide a dynamic way to search for the optimal n_l and L given the tolerance ε and cost N, it does not address the issue for choosing d_l, which requires more theoretical work. In the numerical experiments, we choose d_l as a power function of l.

We first show the results for the geometric mean Asian call option. The parameters are set as follows:

$$\begin{aligned} \text{price} &= \$7.09 & S(0) &= \text{initial asset price} = \$100 \\ r &= \text{risk-less interest rate} = 3\% & \sigma &= \text{asset volatility} = 30\% \\ K &= \text{strike price} = \$100 & T &= \text{time to expiry} = 1 \text{ year} \end{aligned}$$

The accurate price of the option is computed by the approach introduced in Section 6.1.2, which allows us to measure the true approximation error. A sequence of tolerances is

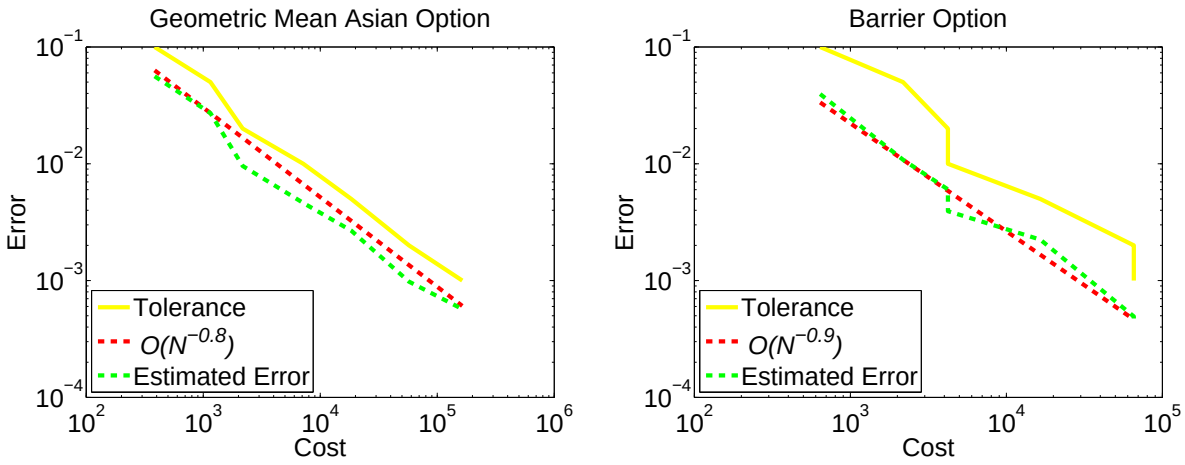
chosen to optimally decide the number of levels L and samples n_l required at each level l. The Karhunen-Loève expansion is used to approximate the Brownian motion $B(t)$. For the sample average, we use scrambled Sobol' sequences, and run 30 replications at each level to compute the estimated error following (6.8). The truncation dimension sequence is chosen as $d_l = 2^l, l = 1, \ldots$. The sample sequence is initialized at $m_0 = 32$. Implementing the algorithm described in (6.7) enables us to achieve the convergence rate $\mathcal{O}(N^{-0.8})$ for the estimation error, see Figure 6.3(a). This result is superior to the single level result. Note that we adopt the same approach as in the single level algorithm to approximate the time integral seen in the payoff function, i.e., choose $m = 52$ to approximate the integral by a mid-point rule. The error of approximating the integral with respect to time is neglected in the theoretical work because we focus on the truncation error and the sampling error.

Table 6.1. The comparison of time consumed for the discrete time method, the single-level algorithm and the multi-level algorithm. Times in seconds.

ε	Discrete Time	Single Level(d =6)	Single Level (d=4)	Multi-Level
0.1	0.5	0.3	0.3	< 0.1
0.05	0.6	0.6	0.5	< 0.1
0.01	1.3	1.1	0.9	0.7
0.005	4.1	1.9	1.6	1.1
0.001	9.5	5.9	5.0	3.6

Table 6.1 compares the computation time of various methods for a sequence of accuracy tolerance. The discrete time method corresponds to discretizing the time interval to 52 equal subintervals, the Brownian motion is sampled by (2.1). We record the time consumed in the single-level experiment performed in Section 6.2. Single-level algorithms with truncations $d = 4$ and $d = 6$ are also The last piece is the multi-level algorithm described above, the Karhunen-Loève Expansion method in (2.7) is used for the Brownian motion simulation. Note that Sobol' sequences are used in all three methods for simulation.

For the barrier option, similar parameters are chosen as in the geometric mean Asian option and the barrier is chosen to be 110. We adopt the same methods as in the geometric mean Asian call option except that the Brownian motion is approximated by the Brownian bridge method. Using the Brownian bridge method yields a better empirical result than the Karhunen-Loève expansion. We remark that the choice of $d_l = 2^l$ here fits the Brownian bridge construction since the Brownian bridge is constructed by continuously refining the time meshs by half. Note that the exact solution for the Barrier option is unreachable, so the estimated error is obtained by running 30 replications at each level. Figure 6.3(b) shows the multi-level algorithm yields the convergence rate $\mathcal{O}(N^{-0.9})$.

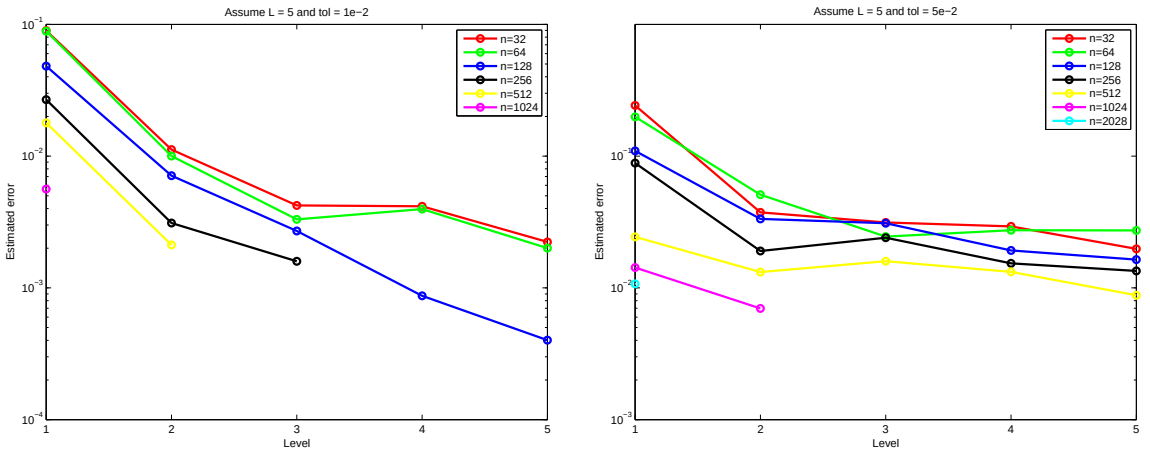


(a) (b)

Figure 6.3. The error bound for the multi-level algorithm. The left panel is using Karhunen-Loève expansion for Brownian motion approximation, the right panel is using Brownian bridge method.

In addition to show the empirical convergence rate, we also perform a numerical test to demonstrate the advantage of the multi-level sampling regime. Given a specific tolerance $\varepsilon = 10^{-2}$, we run the calculation in (6.7) through 5 levels, i.e., $L = 5$. At each level, the algorithm dynamically searches for the required sample paths, this is done by increasing the sample paths by the power of 2. For example, at the first level of Figure 6.4(a), the attempt samples path are 32, 64, 128, 256, 512, 1024. This means the required sample paths at

level 1 is 1024 because the tolerance is met by using 1024 sample paths. Similarly, 512 sample paths are needed at level 2, 256 at level 3, and so on. This shows that the number of samples used decreases as the level increases. This is because the cost for evaluating each term in (6.7) increases with the level l, on the other hand, these differences get smaller with l, which makes the estimation easier with increasing l. This scheme means sampling more frequently in lower dimensions. In addition, Figure 6.4 also shows that the estimated error with the same sample paths decreases with the level, which is consistent with the theoretical result.



(a) Geometric mean Asian call option

(b) Barrier call option

Figure 6.4. The figures demonstrates the required sample paths at each level, which decrease as the level increases.

CHAPTER 7

QUASI-STANDARD ERROR

This chapter investigates an alternative way to estimate the integration error studied in previous chapters. It is widely accepted that estimating the error of QMC from data is problematic, due to the difficulty in measuring the variation of the integrand function and the intractability of the discrepancy. In Chapter 6, we measure the approximation error of QMC by implementing the schemes of randomizing QMC points, which has been extensively studied in [Owe97a, LL02]. This approach performs certain randomized replications of QMC points, then computes the root mean square error of the replicated samples. The quasi-standard error (QSE) was proposed in [War02] and was studied in [Hal05, Owe05]. That error measure is treated as the sample standard error of certain non-randomized quasi-Monte Carlo algorithm, and tends to be a deterministic replication treated as if it were a random replication. This method was investigated by [Owe05] empirically and it showed that the QSE closely tracks the actual error in a one-dimensional integration problem, and is at least as accurate as the standard error on a random replication. However, [Owe05] claims that the QSE can severely underestimate the true error for some QMC points, and it is not appropriate to use the QSE until a more solid theoretical support is developed.

We study the QSE theoretically by investigating the difference between the true integration error and the QSE in the worst-case setting and randomized setting. We represent the difference as a matrix, whose entries are determined by the representers of function evaluation in the setting of reproducing kernel Hilbert spaces. In the worst-case setting, we are able to achieve the lower bound and upper bound of the difference in form of eigenvalues of the matrix, and in the average-case setting, the difference is simply the trace of the matrix.

7.1 Introduction

We follow the approach from [Owe05] to estimate the integral

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

The quasi-Monte Carlo integration approximates the integral by the average

$$\hat{I}(f, \mathscr{P}) = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_i), \tag{7.1}$$

where $\mathscr{P} = \{\mathbf{x}_i\}_{i=1}^n$ is the point set well chosen from $[0,1]^d$. Choose m disjoint groups of such design points $\mathscr{P}_1, \ldots, \mathscr{P}_m$ with equal size n, and denote

$$\mathscr{P}_M = \mathscr{P}_1 \bigcup \cdots \bigcup \mathscr{P}_m$$

as the combination of all the design points. The basic idea is to generate nm design points in $[0,1]^d$, take the first m points as $\mathscr{P}_1$, the second m points as $\mathscr{P}_2$, e.t.c. For each subset design points, there is a QMC estimator

$$\hat{I}(f, \mathscr{P}_l) = \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_{i,l}), \quad l = 1, \ldots m, \quad x_{i,l} \in \mathscr{P}_l.$$

Then the estimator for (7.1) is

$$\hat{I}(f, \mathscr{P}_M) = \frac{1}{nm}\sum_{i=1}^{nm} f(\mathbf{x}_i) = \frac{1}{m}\sum_{l=1}^{m} \hat{I}(f, \mathscr{P}_l). \tag{7.2}$$

We study two quantities here. One is the true error defined as

$$\left| I(f) - \hat{I}(f, \mathscr{P}_M) \right|^2 \tag{7.3}$$

which is immediately available if the true value of $I(f)$ is known. The other one is the QSE which is computed as

$$\mathrm{QSE}(f, \mathscr{P}_M) = \frac{c}{m}\sum_{l=1}^{m} \left(\hat{I}(f, \mathscr{P}_l) - \hat{I}(f, \mathscr{P}_M) \right)^2, \tag{7.4}$$

where c is some constant. Recall that in the randomized QMC error estimate in Chapter 6, we choose the tuning parameter to be 1.96 representing the threshold for a 95% confidence interval. Here, c performs as a tuning parameter to define the QSE.

Our interest here is to see how close are (7.3) and (7.4). Hence, we introduce the difference of the squares:

$$L(f) = \left|I(f) - \hat{I}(f, \mathscr{P}_M)\right|^2 - \frac{c}{m}\sum_{l=1}^{m}\left(\hat{I}(f, \mathscr{P}_l) - \hat{I}(f, \mathscr{P}_M)\right)^2. \tag{7.5}$$

In the following sections, we study the difference $L(f)$ in the worst-case setting and the randomized setting in the reproducing kernel Hilbert space $\mathscr{H}(K)$.

7.2 The Error Difference

We assume the function f resides in the reproducing kernel Hilbert space $\mathscr{H}(K)$ introduced in Chapter 3. Using the reproducing property, we are able to represent the following function valuation in form of inner product, the representers are defined correspondingly:

$$\begin{aligned} I(f) &= \int_{[0,1]^d} f(\mathbf{x})\,\mathrm{d}\mathbf{x} = \langle \eta_0, f\rangle_{\mathscr{H}(K)}, \quad \eta_0(\mathbf{x}) = \int_{[0,1]^d} K_{1:d}(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{y}, \\ \hat{I}(f, \mathscr{P}_l) &= \frac{1}{n}\sum_{i=1}^{n} f(\mathbf{x}_{i,l}) = \langle \eta_l, f\rangle_{\mathscr{H}(K)}, \quad \eta_l(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K_{1:d}(\mathbf{x}, \mathbf{x}_{i,l}), \end{aligned} \tag{7.6}$$

for $l = 1, 2, \ldots m$, and $K_{1:d}$ is the $d-$dimensional kernel function. In addition, the representer for the estimator (7.2) is defined correspondingly as

$$\hat{I}(f, \mathscr{P}_l) = \langle \eta_M, f\rangle_{\mathscr{H}(K)}, \quad \eta_M(\mathbf{x}) = \frac{1}{m}\sum_{l=1}^{m} \eta_l(\mathbf{x}).$$

Theorem 7.1 *Given the sampling regime in the case of QSE defined in* (7.4)*, the difference between the true error* (7.3) *and QSE satisfies*

$$\sup_{\|f\|^2_{\mathscr{H}(K)} \le 1} |L(f)| = \max\left(|\lambda(\mathbf{A}(c))|\right),$$

where λ is the eigenvalue of the $m+1$ by $m+1$ matrix defined in (7.10) *below. More specifically,*

$$\lambda_{\min}(\mathbf{A}(c)) \leq \frac{L(f)}{\|f\|_{\mathscr{H}(K)}} \leq \lambda_{\max}(\mathbf{A}(c)), \tag{7.7}$$

where $\lambda_{\min}(\mathbf{A}(c))$ is the smallest eigenvalue of $\mathbf{A}(c)$ and $\lambda_{\max}(\mathbf{A}(c))$ is the largest.

Proof 7.1 *Given the representations in* (7.6)*, let us reorganize $L(f)$ in* (7.5) *as*

$$\begin{aligned} L(f) &= \left|\langle \eta_0 - \eta_M, f\rangle_{\mathscr{H}(K)}\right|^2 - \frac{c}{m}\sum_{l=1}^{m}\left|\langle \eta_l - \eta_M, f\rangle_{\mathscr{H}(K)}\right|^2 \\ &= \alpha^T \mathbf{H}^T \operatorname{diag}\left(1, -\frac{c}{m}, \cdots, -\frac{c}{m}\right)\mathbf{H}\alpha \end{aligned} \tag{7.8}$$

where $\alpha = (\alpha_0, \alpha_1, \cdots, \alpha_m)^T$. We define the matrix

$$\begin{aligned} \mathbf{H} &= \left(h_{i,j}\right)_{i,j=1}^{m+1} \\ &= \begin{pmatrix} \langle \eta_0 - \eta_M, \eta_0 - \eta_M\rangle_{\mathscr{H}(K)} & \left(\langle \eta_0 - \eta_M, \eta_l - \eta_M\rangle_{\mathscr{H}(K)}\right)_{l=1}^{m} \\ \left(\langle \eta_l - \eta_M, \eta_0 - \eta_M\rangle_{\mathscr{H}(K)}\right)_{l=1}^{m} & \left(\langle \eta_i - \eta_M, \eta_l - \eta_M\rangle_{\mathscr{H}(K)}\right)_{i,l=1}^{m} \end{pmatrix}_{(m+1)\times(m+1)}. \end{aligned}$$

Then, based on the representation in (7.8)*, we have $f = (\eta_0 - \eta_M)\alpha_0 + \sum_{l=1}^{m}(\eta_l - \eta_M)\alpha_l + f_\perp$, where $\langle f_\perp, \eta_l\rangle_{\mathscr{H}(K)} = 0, l = 0, \ldots, m$. In addition, $\|f\|^2_{\mathscr{H}(K)} = \alpha^T\mathbf{H}\alpha$.*

Correspondingly, we have

$$\begin{aligned} h_{1,1} &= \|\eta_0 - \eta_M\|^2_{\mathscr{H}(K)} \\ &= \int_{[0,1]^d}\int_{[0,1]^d} K_{1:d}(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y} - \frac{2}{nm}\sum_{i=1}^{nm}\int_{[0,1]^d} K_{1:d}(\mathbf{x}_i,\mathbf{y})\,\mathrm{d}\mathbf{y} \\ &+ \frac{1}{(nm)^2}\sum_{i,j=1}^{nm} K_{1:d}(\mathbf{x}_i,\mathbf{x}_j), \end{aligned} \tag{7.9}$$

$$\begin{aligned} h_{1,l+1} = h_{l+1,1} &= \langle \eta_l - \eta_M, \eta_0 - \eta_M\rangle_{\mathscr{H}(K)} \\ &= \frac{1}{n}\sum_{i=1}^{n}\int_{[0,1]^d} K_{1:d}(\mathbf{x}_{i,l},\mathbf{y})\,\mathrm{d}\mathbf{y} - \frac{1}{nm}\sum_{i=1}^{nm}\int_{[0,1]^d} K_{1:d}(\mathbf{x}_i,\mathbf{y})\,\mathrm{d}\mathbf{y} \\ &- \frac{1}{n^2 m}\sum_{i,j=1}^{n,nm} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_j) + \frac{1}{(nm)^2}\sum_{i,j=1}^{nm} K_{1:d}(\mathbf{x}_i,\mathbf{x}_j), \\ \textit{where} \quad & \mathbf{x}_{i,l} \in \mathscr{P}_l, \quad l = 1\ldots, m. \end{aligned}$$

$$
\begin{aligned}
h_{k+1,l+1} &= \langle \eta_k - \eta_M, \eta_l - \eta_M \rangle_{\mathscr{H}(K)} \\
&= \frac{1}{n^2} \sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,k}, \mathbf{x}_{j,l}) - \frac{1}{n^2 m} \sum_{i,j=1}^{n,nm} K_{1:d}(\mathbf{x}_{i,k}, \mathbf{x}_j) \\
&- \frac{1}{n^2 m} \sum_{i,j=1}^{n,nm} K_{1:d}(\mathbf{x}_{i,l}, \mathbf{x}_j) + \frac{1}{(nm)^2} \sum_{i,j=1}^{nm} K_{1:d}(\mathbf{x}_i, \mathbf{x}_j), \\
\textit{where} \quad & \mathbf{x}_{i,k} \in \mathscr{P}_k, \quad \mathbf{x}_{i,l} \in \mathscr{P}_l, \quad k,l = 1,\ldots,m.
\end{aligned}
$$

We then use the Cholesky decomposition to denote $\mathbf{H}$ *as* $\mathbf{H} = \mathbf{B}^T\mathbf{B}$. *Let* $\beta = \mathbf{B}\alpha$, *so* $\mathbf{H}\alpha = \mathbf{B}^T\beta$, *thus*

$$
L(f) = \beta^T \mathbf{B}\,\mathrm{diag}\left(1, -\frac{c}{m}, \cdots, -\frac{c}{m}\right)\mathbf{B}^T\beta = \beta^T \mathbf{A}(c)\beta,
$$

where

$$
\mathbf{A}(c) = \mathbf{B}\,\mathrm{diag}\left(1, -\frac{c}{m}, \cdots, -\frac{c}{m}\right)\mathbf{B}^T. \tag{7.10}
$$

In addition, we can see that $\|f\|^2_{\mathscr{H}(K)} = \beta^T\beta$ *such that*

$$
\sup_{\|f\|^2_{\mathscr{H}(K)} \le 1} |L(f)| = \|\mathbf{A}(c)\|_{\mathscr{H}(K)} = \lambda_{\max}(\mathbf{A}(c)).
$$

Next, we study the difference in the randomized setting. Before proceeding, the following lemma related to the trace of the matrix $\mathbf{A}(c)$ is introduced to facilitate the derivation of the expectation of the difference (7.5) appearing in Theorem 7.2.

Lemma 7.1 *The trace of the matrix* $\mathbf{A}(c)$ *is*

$$
\mathrm{Trace}(\mathbf{A}(c)) = h(1,1) - \frac{c}{m}\sum_{i=1}^{m} \|\eta_i\|^2_{\mathscr{H}(K)} + \frac{c}{m^2}\sum_{i,j=1}^{m} \langle \eta_i, \eta_j \rangle_{\mathscr{H}(K)}
$$

Proof 7.2 *Given the definition of* $\mathbf{A}(c)$ *in* (7.10)*, the trace of the matrix is computed as*

$$\begin{aligned}
\text{Trace}(\mathbf{A}(c)) &= \text{Trace}\left(\mathbf{B}\,\text{diag}\left(1,-\frac{c}{m},\cdots,-\frac{c}{m}\right)\mathbf{B}^T\right)\\
&= \text{Trace}\left(\text{diag}\left(1,-\frac{c}{m},\cdots,-\frac{c}{m}\right)\mathbf{B}^T\mathbf{B}\right)\\
&= \text{Trace}\left(\text{diag}\left(1,-\frac{c}{m},\cdots,-\frac{c}{m}\right)\mathbf{H}\right)\\
&= h(1,1)-\frac{c}{m}\sum_{i=2}^{m+1}h(i,i).
\end{aligned}$$

$h(1,1)$ *is known from the proof of Theorem 7.1. For* $hi, i, i = 2,\ldots m+1$*:*

$$\begin{aligned}
h(i,i) &= \|\eta_i-\eta_M\|^2_{\mathscr{H}(K)} = \left\|\frac{1}{m}\sum_{j=1}^{m}(\eta_i-\eta_j)\right\|^2_{\mathscr{H}(K)}\\
&= \frac{1}{m^2}\sum_{j,k=1}^{m}\left[\langle\eta_i,\eta_i\rangle_{\mathscr{H}(K)}-2\langle\eta_i,\eta_j\rangle_{\mathscr{H}(K)}+\langle\eta_j,\eta_k\rangle_{\mathscr{H}(K)}\right]\\
&= \|\eta_i\|^2_{\mathscr{H}(K)}-\frac{2}{m}\sum_{j=1}^{m}\langle\eta_i,\eta_j\rangle_{\mathscr{H}(K)}+\frac{1}{m^2}\sum_{j,k=1}^{m}\langle\eta_j,\eta_k\rangle_{\mathscr{H}(K)}
\end{aligned}$$

So, the trace of $\mathbf{A}(c)$ *is*

$$\begin{aligned}
\text{Trace}(\mathbf{A}(c)) &= h(1,1)-\frac{c}{m}\sum_{i=1}^{m}\left[\|\eta_i\|^2_{\mathscr{H}(K)}-\frac{2}{m}\sum_{j=1}^{m}\langle\eta_i,\eta_j\rangle_{\mathscr{H}(K)}+\frac{1}{m^2}\sum_{j,k=1}^{m}\langle\eta_j,\eta_k\rangle_{\mathscr{H}(K)}\right]\\
&= h(1,1)-\frac{c}{m}\sum_{i=1}^{m}\|\eta_i\|^2_{\mathscr{H}(K)}+\frac{c}{m^2}\sum_{i,j=1}^{m}\langle\eta_i,\eta_j\rangle_{\mathscr{H}(K)}
\end{aligned}$$

Theorem 7.2 *Given the QSE defined in* (7.4)*, the difference between the true error* (7.3) *and QSE satisfies*

$$\mathbb{E}(L(f)) = \text{Trace}(\mathbf{A}(c)).$$

where $\text{Trace}(\mathbf{A}(c))$ *is trace of the* $m+1$ *by* $m+1$ *matrix defined in* (7.10)*.*

Proof 7.3 *The proof is straightforward by working out each term in the following expectation*

$$\mathbb{E}(L(f)) = \underbrace{\mathbb{E}[I(f)-\hat{I}(f,\mathscr{P}_M)]^2}_{A}-\frac{c}{m}\sum_{l=1}^{m}\underbrace{\mathbb{E}[\hat{I}(f,\mathscr{P}_l)-\hat{I}(f,\mathscr{P}_M)]^2}_{B_l} = A-\frac{c}{m}\sum_{l=1}^{m}B_l.$$

Here we use the notations $A, B_l, l = 1,2,\ldots,m$ for simplicity.

For A,

$$A = \mathbb{E}\left[\frac{1}{m}\sum_{l=1}^{m}\left(I(f)-\hat{I}(f,\mathscr{P}_l)\right)\right]^2 = \frac{1}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\mathbb{E}\left[\left(I(f)-\hat{I}(f,\mathscr{P}_k)\right)\left(I(f)-\hat{I}(f,\mathscr{P}_l)\right)\right]$$
$$= \mathbb{E}[I(f)]^2 + \frac{2}{m}\sum_{k=1}^{m}\mathbb{E}[I(f)\hat{I}(f,\mathscr{P}_k)] + \frac{1}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\mathbb{E}\left[\hat{I}(f,\mathscr{P}_k)\hat{I}(f,\mathscr{P}_l)\right].$$

For $B_l, l = 1,2,\ldots,m$,

$$B_l = \mathbb{E}[\hat{I}(f,\mathscr{P}_l)-\hat{I}(f,\mathscr{P}_M)]^2 = \mathbb{E}\left[\frac{1}{m}\sum_{k=1}^{m}\left(\hat{I}(f,\mathscr{P}_l)-\hat{I}(f,\mathscr{P}_k)\right)\right]^2$$
$$= \frac{1}{m^2}\mathbb{E}\left[\sum_{j=1}^{m}\sum_{k=1}^{m}\left[\left(\hat{I}(f,\mathscr{P}_l)-\hat{I}(f,\mathscr{P}_k)\right)\left(\hat{I}(f,\mathscr{P}_l)-\hat{I}(f,\mathscr{P}_j)\right)\right]\right]$$
$$= \mathbb{E}[\hat{I}(f,\mathscr{P}_l)]^2 - \frac{2}{m}\sum_{k=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_l)\hat{I}(f,\mathscr{P}_k)] + \frac{1}{m^2}\sum_{j=1}^{m}\sum_{k=1}^{m}\mathbb{E}\left[\hat{I}(f,\mathscr{P}_j)\hat{I}(f,\mathscr{P}_k)\right]$$

Hence,

$$\mathbb{E}(L(f)) = A - \frac{c}{m}\sum_{l=1}^{m}B_l$$
$$= \mathbb{E}[I(f)]^2 + \frac{2}{m}\sum_{k=1}^{m}\mathbb{E}[I(f)\hat{I}(f,\mathscr{P}_k)] + \frac{1+2c}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_l)\hat{I}(f,\mathscr{P}_k)]$$
$$- \frac{c}{m^2}\sum_{j=1}^{m}\sum_{k=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_j)\hat{I}(f,\mathscr{P}_k)] - \frac{c}{m}\sum_{l=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_l)]^2$$
$$= \mathbb{E}[I(f)]^2 + \frac{2}{m}\sum_{k=1}^{m}\mathbb{E}_f[I(f)\hat{I}(f,\mathscr{P}_k)] + \frac{1+c}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_l)\hat{I}(f,\mathscr{P}_k)]$$
$$- \frac{c}{m}\sum_{l=1}^{m}\mathbb{E}[\hat{I}(f,\mathscr{P}_l)]^2.$$

Adopting the inner product representation as in (7.6)

$$\mathbb{E}[\hat{I}(f,\mathscr{P}_l)I(f)] = \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}f(x_{i,l})\int_{[0,1]^d}f(\mathbf{x})\,\mathrm{d}\mathbf{x}\right] = \frac{1}{n}\sum_{i=1}^{n}\int_{[0,1]^d}K_{1:d}(\mathbf{x}_{i,l},y)\,\mathrm{d}\mathbf{y},$$
$$E[\hat{I}(f,\mathscr{P}_l)\hat{I}(f,\mathscr{P}_k)] = E\left[\frac{1}{n}\sum_{i=1}^{n}f(\mathbf{x}_{i,l})\frac{1}{n}\sum_{i=1}^{n}f(\mathbf{x}_{i,k})\right] = \frac{1}{n^2}\sum_{i,j=1}^{n}K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,k}),$$
$$E[I(f)]^2 = E\left[\int_{[0,1]^d}f(\mathbf{x})\,\mathrm{d}\mathbf{x}\int_{[0,1]^d}f(\mathbf{x})\,\mathrm{d}\mathbf{x}\right] = \int_{[0,1]^d}\int_{[0,1]^d}K_{1:d}(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y}.$$

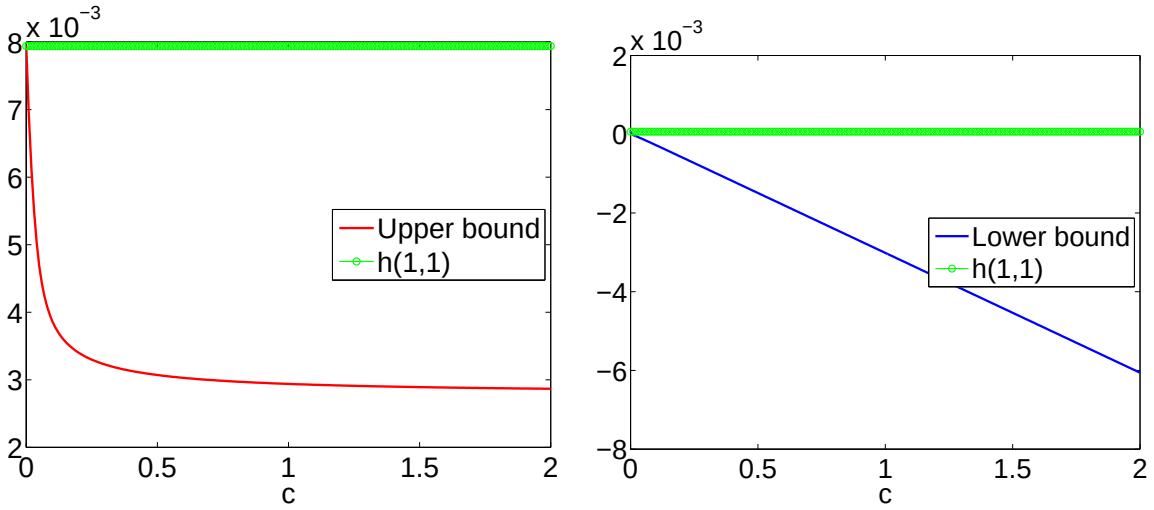
Continue from the above,

$$\begin{aligned}
\mathbb{E}(L(f)) &= \int_{[0,1]^d}\int_{[0,1]^d} K_{1:d}(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y} + \frac{2}{m}\sum_{k=1}^{m}\frac{1}{n}\sum_{i=1}^{n}\int_{[0,1]^d} K_{1:d}(\mathbf{x}_{i,k},\mathbf{y})\,\mathrm{d}\mathbf{y} \\
&+ \frac{1+c}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\frac{1}{n^2}\sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,k}) - \frac{c}{m}\sum_{l=1}^{m}\frac{1}{n^2}\sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,l}) \\
&= \int_{[0,1]}\int_{[0,1]} K_{1:d}(\mathbf{x},\mathbf{y})\,\mathrm{d}\mathbf{x}\,\mathrm{d}\mathbf{y} + \frac{2}{m}\sum_{k=1}^{m}\frac{1}{n}\sum_{i=1}^{n}\int_{[0,1]^d} K_{1:d}(\mathbf{x}_{i,k},\mathbf{y})\,\mathrm{d}\mathbf{y} \\
&+ \frac{1}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\frac{1}{n^2}\sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,k}) \\
&+ \frac{c}{m^2}\sum_{l=1}^{m}\sum_{k=1}^{m}\frac{1}{n^2}\sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,k}) - \frac{c}{m}\sum_{l=1}^{m}\frac{1}{n^2}\sum_{i,j=1}^{n} K_{1:d}(\mathbf{x}_{i,l},\mathbf{x}_{j,l}) \\
&= h(1,1) - \frac{c}{m}\sum_{i=1}^{m}\|\eta_i\|^2_{\mathscr{H}(K)} + \frac{c}{m^2}\sum_{i,j=1}^{m}\langle\eta_i,\eta_j\rangle_{\mathscr{H}(K)} \\
&= \mathrm{Trace}(A(c)),
\end{aligned}$$

as from Lemma 7.1

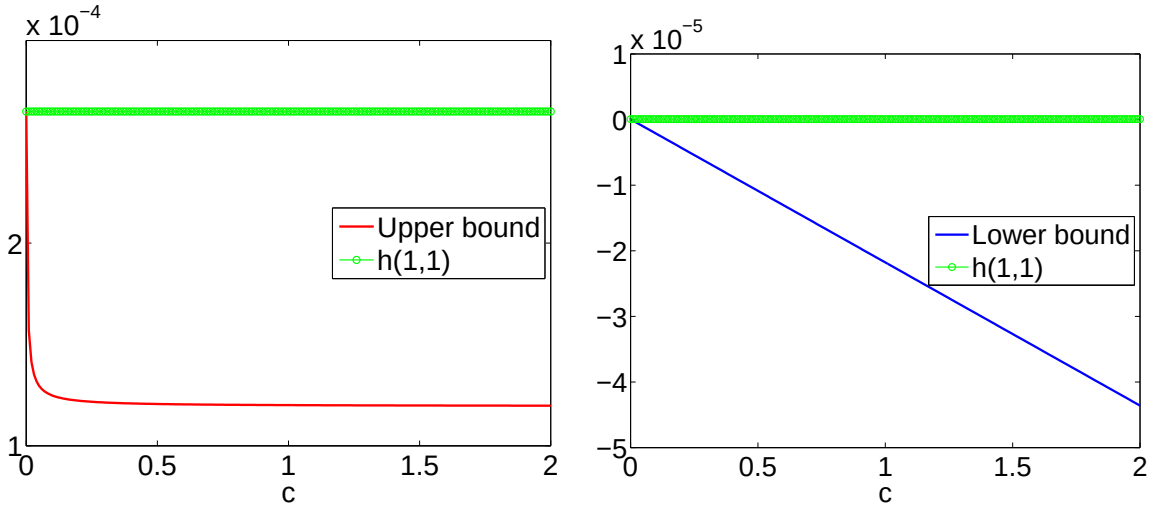
7.3 Numerical Examples

We study the simple one-dimensional integration problem defined on $[0,1]$, and use the kernel $K_1(x,y) = \min(x,y)$. Here we choose $m = 15, n = 256$ introduced in (7.2). Follow the steps in the proof of Theorem 7.1, each entry of matrix $\mathbf{H}$ can be computed explicitly. Figure 7.1 shows the use of simple random sequence and Sobol' sequence for the computation of the upper bound and the lower bound for $L(f)$ as defined in (7.7), for various choices of c. It can be observed from the figures that as c increases, the upper bound of $L(f)$ converges to the asymptotic value while the lower bound continues to decrease as c increases. In addition, using Sobol' sequences yields a much smaller magnitude of $L(f)$. Note that when $c = 0$, both the upper bound and the lower bound equal to the squared discrepancy, measured by $h(1,1)$ defined in (7.9). See the green flat lines in both charts. The Sobol' sequence yields a much smaller discprepancy, which is due to the superior distribution property of Sobol' sequence in the lower dimensions.



(a) The upper bound and the squared discrepancy (b) The lower bound and the squared discrepancy

Figure 7.1. The simple random sequence



(a) The upper bound and the squared discrepancy (b) The lower bound and the squared discrepancy

Figure 7.2. The Sobol' sequence

CHAPTER 8

CONCLUSION

In this dissertation, we study algorithms for numerical integration with respect to a product probability measure on the infinite-dimensional sequence. Both the single-level and multi-level algorithm in the sense of space dimension are constructed and the approximation errors for them are analyzed either in the worst-case setting or the randomized setting. This chapter summarizes the main results and brings out the future work to be pursued.

8.1 Summary

Infinite-dimensional quadratures motivated by the option pricing problem can be well modeled as the expectation of functions of infinitely many random variables. Either the Karhunen-Loève expansion or the Brownian bridge expansion allows us to take the series expansion $B(t) = \sum_{j=1}^{\infty} X_j \cdot e_j(t)$ as in (2.6). We briefly discuss the classes F of integrands. The basic idea is to consider the infinite-dimensional integration as the limiting case of the high-dimensional integration, and thus we rely on error bounds for finite-dimensional integration that are fully dimension independent, which are provided in the study of tractability of high-dimensional problems. Most frequently, tensor products of weighted reproducing kernel Hilbert spaces are employed in the tractability analysis. In the case of product weights this construction is based on a sequence of weights $\gamma_j > 0$ and a reproducing kernel K_1 for real-valued functions on D. In the dissertation we study the limiting case, namely the reproducing kernel

$$K(\mathbf{x}, \mathbf{y}) = \sum_{u} \prod_{j \in u} \gamma_j K_1(x_j, y_j),$$

where u varies over all finite subsets of $\mathbb{N}$ and $\mathbf{x}$ and $\mathbf{y}$ belong to a subset of $D^{\mathbb{N}}$ with μ-measure one. The class F of integrands is the unit ball F in the Hilbert space $\mathscr{H}(K)$ with reproducing kernel K.

For the single-level algorithm, a common computational approach truncates the series expansion of the Brownian motion $B(t)$ and the infinite-dimensional integral is approximated by a finite-dimensional integral with a suitably chosen dimension. The latter integral is then approximated by means of a deterministic or randomized (Monte Carlo) algorithm. Accordingly, the integrand is sampled (evaluated) at a finite number of deterministically or randomly chosen points from a fixed finite-dimensional affine subspace $\mathrm{span}\{e_1,\dots,e_m\}+e$, which amounts to sampling of the integrand at points from the finite-dimensional subspace $\{\mathbf{x}\in\mathbb{R}^{\mathbb{N}} : x_{m+1}=x_{m+2}=\cdots=c\}$. Any sampling regime of this kind is called fixed subspace sampling. In contrast to the single-level algorithm, a multi-level algorithm evaluates the integrand at points from a hierarchy of finite-dimensional subspaces, and this sampling regime is called the variable subspace sampling. It has turned out to be superior to fixed subspace sampling for a number integration problems. Here superiority refers to a comparison of specific algorithms based on numerical experiments or upper bounds for their error and cost, or a comparison based on the analysis of minimal errors, i.e., on the study of upper and lower bounds.

We study the minimal worst case errors $e_s^{\mathrm{fix}}(N,F)$ and $e_s^{\mathrm{var}}(N,F)$ that can be achieved by deterministic algorithms that use fixed or variable subspace sampling, respectively, with worst-case cost at most N. We derive upper and lower bounds for these quantities, which depend on the decay of the weights γ_j and on respective upper and lower bounds for finite-dimensional integration on the unit balls $B(K_{1:d})$, where

$$K_{1:d}(\mathbf{x},\mathbf{y})=\sum_{u\subseteq\{1,\dots,d\}}\prod_{j\in u}\gamma_j K_1(x_j,y_j)$$

is a reproducing kernel for functions of the variables $x_1,\dots,x_d\in D$. The upper bounds for $e_s^{\mathrm{var}}(N,F)$ are achieved by suitable multi-level algorithms, and in the corresponding analysis auxiliary weights γ_j' such that $\lim_{j\to\infty}\gamma_j/\gamma_j'=0$ are employed.

To give a flavor of our results, consider the uniform distribution ρ_1 on $D=[0,1]$

and the kernel

$$K_1(x,y) = \min(x,y), \qquad x,y \in [0,1],$$

and assume $\gamma_j = j^{-1-2q}$ with $q > 0$. Furthermore, let $s = 1$ in the definition of the cost. In order to simplify the presentation we put

$$\lambda^{\mathrm{var}} = \sup\{\chi > 0 : \sup_{N\in\mathbb{N}} e_1^{\mathrm{var}}(N,F) \cdot N^{\chi} < \infty\},$$

and we use λ^{fix} to denote the corresponding quantity for fixed subspace sampling. Roughly speaking, λ^{var} and λ^{fix} are the best orders of convergence that can be achieved by any sequence of algorithms using variable or fixed subspace sampling, respectively. Clearly, $\lambda^{\mathrm{var}} \geq \lambda^{\mathrm{fix}}$. We have

$$\lambda^{\mathrm{fix}} = \frac{q}{q+1}$$

if $q \geq 1/2$, and otherwise the rather tight bounds

$$\frac{q}{2q/(2q+1)+1} \leq \lambda^{\mathrm{fix}} \leq \frac{q}{q+1}$$

hold true. For variable subspace sampling

$$\lambda^{\mathrm{var}} = \min(q,1)$$

if $|q-1| \geq 1/2$, while otherwise we only know that

$$\frac{q+1/2}{2} \leq \lambda^{\mathrm{var}} \leq \min(q,1)$$

with a significant gap between the upper and the lower bound. Still we conclude that variable subspace sampling is superior to fixed subspace sampling for all $q > 0$. Moreover, the lower bounds for λ^{var} are achieved by multi-level algorithms based on rank-1 lattice rules, so that we have optimality for these multi-level algorithms in the case $|q-1| \geq 1/2$. The proof of the lower bound for λ^{var} relies on a tractability result from [HSW04b] for rank-1 lattice rules. We add that the lower bound for λ^{var} is independently proven in [Gne10] by means of a different approach.

We then analyze the minimal worst case errors $e_{N,\mathrm{fix}}(F)$ that can be achieved by randomized algorithms that use fixed subspace sampling with worst case cost at most N. We derive upper and lower bounds for these quantities, which depend on the decay of the weights γ_j and on respective upper and lower bounds for finite-dimensional integration on the unit balls $B(K_{1:d})$, where

$$K_{1:d}(\mathbf{x},\mathbf{y}) = \sum_{u \subseteq 1:d} \prod_{j \in u} \gamma_j K_1(x_j, y_j)$$

is a reproducing kernel for functions on $D^{1:d}$. See Theorems 5.1 and 5.2.

For variable subspace sampling we only have upper bounds for the respective minimal errors $e_{N,\mathrm{var}}(F)$ as non-trivial results. For the analysis we choose a suitable sequence of auxiliary weights γ'_j such that

$$\lim_{j\to\infty} \gamma_j/\gamma'_j = 0,$$

and we consider the corresponding counterparts K' and $K'_{1:d}$ of the kernels K and $K_{1:d}$, respectively. Our upper bounds for $e_{N,\mathrm{var}}(F)$ depend on the decay of the weights γ_j and γ'_j and on upper bounds for finite-dimensional integration on the unit balls $B(K'_{1:d})$. See Theorems 5.4 and 5.5. These bounds are achieved by multi-level algorithms, where a proper decay of variances results from embedding $\mathscr{H}(K)$ into $\mathscr{H}(K')$.

To give a flavor of our results, consider the uniform distribution ρ on $D = [0,1]$ and the kernel

$$K_1(x,y) = 1/3 + (x^2+y^2)/2 - \max(x,y), \quad x,y \in [0,1],$$

and assume $\gamma_j = j^{-\alpha}$ with $\alpha > 1$. In order to simplify the presentation we put

$$\lambda_{\mathrm{var}} = \sup\{\chi > 0 : \sup_{N\in\mathbb{N}} e_{N,\mathrm{var}}(F) \cdot N^{\chi} < \infty\}$$

and we use λ_{fix} to denote the corresponding quantity for fixed subspace sampling. Roughly speaking, λ_{var} and λ_{fix} are the best orders of convergence that can be achieved by any

sequence of algorithms using variable or fixed subspace sampling, respectively. Clearly, $\lambda_{\text{var}} \geq \lambda_{\text{fix}}$. We have

$$\lambda_{\text{fix}} = 3/2\,(\alpha-1)/(\alpha+2),$$

see Corollary 5.3, and

$$\lambda_{\text{var}} \geq 3/2\,\min((\alpha-1)/10,1) > \lambda_{\text{fix}}$$

if $\alpha > 8$, see Corollary 5.4, where the order $3/2\,\min((\alpha-1)/10,1)$ is (almost) achieved with suitable multi-level algorithms based on scrambled QMC rules as building blocks. Scrambling is a randomization technique that preserves good discrepancy properties of point sets, and was introduced by [Owe97a]. In the present case we use a result from [YH05], which studied randomized quadrature formulas that use base b scrambling of a Niederreiter (t,m,s)-net in base b. Consequently, variable subspace sampling is superior to fixed subspace sampling (at least) if $\alpha > 8$. Moreover, we have almost optimality for the multi-level algorithm (at least) if $\alpha \geq 11$ due to a classical result for one-dimensional integration, which implies $\lambda_{\text{var}} \leq 3/2$.

8.2 Future Work

As seen from the numerical experiments, we are able to optimally select the number of levels L and samples n_l given a specific choice of d_l. In those numerical examples, d_l is chosen in our favor. The error analysis enables us to construct the single-level and multi-level algorithm optimally with the computation constraint and tolerance theoretically, however how to implement those algorithms in a more general sense is still under exploration. We believe this could be solved if there is an efficient solution to the error measure for the randomized algorithms. For example, at each level l of the multilevel algorithm, if the error can be written in some function of the truncated dimension explicitly, the control of d_l is straightforward. In the current work, we take certain number of replications at each level to estimate the error by computing the root mean square error, this could either overestimate or underestimate the true error. This can be seen from the separate work in Chapter 7.

APPENDIX A

AUXILIARY RESULTS

Suppose that $E = E_1 \times E_2$ with $E_1, E_2 \neq \emptyset$, fix $e_2 \in E_2$, and let K denote a reproducing kernel on $E \times E$. Consider the linear mapping $\Psi : \mathscr{R}^E \to \mathscr{R}^E$ given by

$$(\Psi f)(x_1, x_2) = f(x_1, e_2), \qquad x_j \in E_j,$$

and the reproducing kernel J on $E \times E$ defined by

$$J((x_1, x_2), (y_1, y_2)) = K((x_1, e_2), (y_1, e_2)).$$

Note that $J \neq 0$ iff there exists a point $x_1 \in E_1$ such that $K((x_1, e_2), (x_1, e_2)) \neq 0$. In particular, $J = 0$ might hold for a kernel $K \neq 0$.

Lemma A.1 *We have*

$$\{\Psi f : f \in \mathscr{H}(K),\ \|f\|_{\mathscr{H}}(K) \leq 1\} = \{g \in \mathscr{H}(J) : \|g\|_J \leq 1\}.$$

Proof A.1 *Consider the closed subspaces*

$$\mathscr{H}_0 = \{f \in \mathscr{H}(K) : f|_{E_1 \times \{e_2\}} = 0\}$$

and

$$\mathscr{H}_0^{\perp} = \overline{\text{span}}\{K(\cdot, x) : x \in E_1 \times \{e_2\}\}$$

of $\mathscr{H}(K)$. For $f = \sum_{i=1}^n a^{(i)} K(\cdot, (y_1^{(i)}, e_2))$ with $a^{(i)} \in \mathscr{R}$ and $y_1^{(i)} \in E_1$ we have

$$(\Psi f)(x_1, x_2) = \sum_{i=1}^n a^{(i)} K((x_1, e_2), (y_1^{(i)}, e_2)) = \sum_{i=1}^n a^{(i)} J((x_1, x_2), (y_1^{(i)}, e_2)),$$

which implies $\Psi f \in \mathscr{H}(J)$ and, by definition, $\|\Psi f\|_{\mathscr{H}}(J) = \|f\|_{\mathscr{H}}(K)$. The same conclusions hold for every $f \in \mathscr{H}_0^{\perp}$, and furthermore $\Psi(\mathscr{H}_0^{\perp}) = \mathscr{H}(J)$.

Let P denote the orthogonal projection onto $\mathscr{H}_0^{\perp}$. Clearly $\Psi f = \Psi P f$ for $f \in \mathscr{H}(K)$, so that $\Psi f \in \mathscr{H}(J)$ and $\|\Psi f\|_{\mathscr{H}}(J) = \|Pf\|_{\mathscr{H}}(K) \leq \|f\|_{\mathscr{H}}(K)$.

We also consider the reproducing kernel L on $E_1 \times E_1$ that is given by

$$L(x_1, y_1) = K((x_1, e_2), (y_1, e_2)).$$

Lemma A.2 *We have*

$$\mathscr{H}(J) = \{f : E \to \mathscr{R} : \exists g \in \mathscr{H}(L) \forall x_2 \in E_2 : \ f(\cdot, x_2) = g\}.$$

Proof A.2 *Let H denote the set on the right-hand side in Lemma A.2. We define an inner product on H by*

$$\langle f, f' \rangle = \langle f(\cdot, e_2), f'(\cdot, e_2) \rangle_L,$$

which turns H into a Hilbert space. Obviously, $J(\cdot, (y_1, y_2)) \in H$ and

$$\langle f, J(\cdot, (y_1, y_2)) \rangle = \langle f(\cdot, e_2), L(\cdot, y_1) \rangle_L = \langle f(\cdot, y_2), L(\cdot, y_1) \rangle_L = f(y_1, y_2)$$

for all $(y_1, y_2) \in E_1 \times E_2$ and $f \in \mathscr{H}$.

BIBLIOGRAPHY

[Adl90] R. J. Adler. *An Introduction to Continuity, Extrema, and Related Topics for General Gaussian Processes*. Institute of Mathematical Statistics, Hayward, California, 1990.

[BS73] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81:637—654, 1973.

[CDMGR09] J. Creutzig, S. Dereich, T. Müller-Gronbach, and K. Ritter. Infinite-dimensional quadrature and approximation of distributions. *Found. Comput. Math.*, 9:391–429, 2009.

[CM96] R. E. Caflisch and W. Morokoff. Valuation of mortgage backed securities using the quasi-Monte Carlo method. In *International Association of Financial Engineers First Annual Computational Finance Conference*, 1996.

[CMO97] R. E. Caflisch, W. Morokoff, and A. Owen. Valuation of mortgage backed securities using Brownian bridges to reduce effective dimension. *J. Comput. Finance.*, 1:27–46, 1997.

[DP10] J. Dick and F. Pillichshammer. *Digital Nets and Sequences*. Cambridge University Press, New York, 2010.

[DR84] P. J. Davis and P. Rabinowitz. *Methods of Numerical Integration*. Academic Press, New York, second edition, 1984.

[Gil] M. B. Giles. Improved multilevel Monte Carlo convergence using the Milstein scheme. In *Monte Carlo and Quasi-Monte Carlo Methods 2006*, pages 343–358.

[Gil08] M. B. Giles. Multilevel monte carlo path simulation. *Oper. Res.*, 56:607–617, 2008.

[Gne10] Michael Gnewuch. Infinite-dimensional integration on weighted hilbert spaces. *Submitted for publication*, 2010.

[GW09] M. B. Giles and B. J. Waterhouse. Multilevel quasi-Monte Carlo path simulation. *Radon Series Comp. Appl. Math*, 8:1–18, 2009.

[Hal05] J. H. Halton. Quasi-probability: Why quasi-Monte-Carlo methods are statistically valid and how their errors can be estimated statistically. *Monte Carlo Methods and Appl.*, 11:203–350, 2005.

[Hei98] S. Heinrich. Monte carlo complexity of global solution of integral equations. *J. Complexity.*, 14:151–175, 1998.

[HH99] F. J. Hickernell and H. S. Hong. The asymptotic efficiency of randomized nets for quadrature. *Math. Comp.*, 68:767–791, 1999.

[HHLL00] F. J. Hickernell, H. S. Hong, P. L'Écuyer, and C. Lemieux. Extensible lattice sequences for quasi-Monte Carlo quadrature. *SIAM J. Sci. Comput.*, 22:1117–1138, 2000.

[HHY01] S. Heinrich, F. J. Hickernell, and R. X. Yue. Integration of multivariate Haar wavelet series. In Y. Y. Tang, V. Wickerhauser, P. C. Yuen, and C. H. Li, editors, *Wavelet Analysis and Its Applications*, number 2251 in Lecture Notes in Computer Science, pages 99–106. Springer-Verlag, New York, 2001.

[Hic98] F. J. Hickernell. A generalized discrepancy and quadrature error bound. *Math. Comp.*, 67(221):299–322, 1998.

[Hig01] D. J. Higham. An algorithmic introduction to numerical simulation of stochastic differential equations. *SIAM Review, Education Section*, 43:525–546, 2001.

[Hla61] E. Hlawka. Funktionen von beschrankter variation in der theorie der gleichverteilung. *Ann. Mat. Pura Appl.*, 54:325–333, 1961.

[HMGNR10] F. J. Hickernell, T. Müller-Gronbach, B. Niu, and K. Ritter. Multi-level Monte Carlo algorithms for infinite-dimensional integration on $\mathbb{R}^{\mathbb{N}}$. *J. Complexity.*, 26(3):229–254, 2010.

[HN03] F. J. Hickernell and H. Niederreiter. The existence of good extensible rank-1 lattices. *J. Complexity.*, 19:286–300, 2003.

[HSW04a] F. J. Hickernell, I. H. Sloan, and G. W. Wasilkowski. On tractability of weighted integration for certain Banach spaces of functions. In Niederreiter [Nie04], pages 51–71.

[HSW04b] F. J. Hickernell, I. H. Sloan, and G. W. Wasilkowski. The strong tractability of multivariate integration using lattice rules. In Niederreiter [Nie04], pages 259–273.

[HW00] F. J. Hickernell and H. Woźniakowski. Integration and approximation in arbitrary dimensions. *Adv. Comput. Math.*, 12:25–58, 2000.

[HW02] F. J. Hickernell and X. Wang. The error bounds and tractability of quasi-Monte Carlo algorithms in infinite dimension. *Math. Comp.*, 71(240):1641–1661, 2002.

[HY00] F. J. Hickernell and R. X. Yue. The mean square discrepancy of scrambled (t,s)-sequences. *SIAM J. Numer. Anal.*, 38:1089–1112, 2000.

[KS91] Ioannis Karatzas and Steven E. Shreve. *Brownian Motion and Stochastic Calculus (Graduate Texts in Mathematics)*. Springer, August 1991.

[KSWW10] F. Y. Kuo, I. H. Sloan, G. W. Wasilkowski, and H. Woźniakowski. Liberating the dimension. *J. Complexity.*, 26(5):422–454, 2010.

[LL00] P. L'Ecuyer and C. Lemieux. Variance reduction via lattice rules. *Management Science*, 46:1214–1235, 2000.

[LL02] P. L'Ecuyer and C. Lemieux. Recent advances in randomized quasi-Monte Carlo methods. In M. Dror, P. L'Ecuyer, and F. Szidarovszki, editors, *Modeling Uncertainty: An Examination of Stochastic Theory, Methods, and Applications*, pages 419–474. Kluwer Academic Publishers, Boston, 2002.

[LS01] F. A. Longstaff and E. S. Schwartz. Valuing american options by simulation: A simple least-squares approach. *Review of Financial Studies*, 14:113–147, 2001.

[Mck69] H. P. Mckean. *Stochastic Integrals*, volume 1733. Academic Press, New York, 1969.

[Mor98] W. J. Morokoff. Generating quasi-random paths for stochastic processes. *SIAM Rev.*, 40:765–788, 1998.

[NH10] B. Niu and F. J. Hickernell. Monte Carlo simulation of stochastic integrals when the cost of function evaluation is dimension dependent. In P. L'Ecuyer and A. Owen, editors, *Monte Carlo and Quasi-Monte Carlo Methods 2008*, pages 545–560. Springer-Verlag, Berlin, 2010.

[NHMGR10] B. Niu, F. J. Hickernell, T. Müller-Gronbach, and K. Ritter. Deterministic multi-level Monte Carlo algorithms for infinite-dimensional integration on $\mathbb{R}^{\mathbb{N}}$. *J. Complexity.*, In Press, Corrected Proof, 2010.

[Nie87] H. Niederreiter. Point sets and sequences with small discrepancy. *Monatsh. Math.*, 194:273–337, 1987.

[Nie92] H. Niederreiter. *Random number generation and Quasi-Monte Carlo methods*, volume 63 of *CBMS-NSF Regional Conference Series in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1992.

[Nie04] H. Niederreiter, editor. *Monte Carlo and Quasi-Monte Carlo Methods 2002*. Springer-Verlag, Berlin, 2004.

[Nie10] H. Niederreiter. Quasi-Monte Carlo methods. *Encyclopedia of Quantitative Finance*, online, 2010.

[Nov88] E. Novak. *Deterministic and stochastic error bounds in numerical analysis*, volume 1349 of *Lecture Notes in Mathematics*. Springer-Verlag, Berlin, 1988.

[NW01] E. Novak and H. Woźniakowski. When are integration and discrepancy tractable? In *Foundations of Computational Mathematics*, volume 284 of *London Math. Soc. Lecture Note Ser.*, pages 211–266. Cambridge University Press, Cambridge, 2001.

[NW08] E. Novak and H. Woźniakowski. *Tractability of Multivariate Problems Volume 1: Linear Information*. Number 6 in EMS Tracts in Mathematics. European Mathematical Society, 2008.

[NW10] E. Novak and H. Woźniakowski. *Tractability of Multivariate Problems Volume 2: Standard Information for Functionals*, volume 12 of *EMS Tracts in Mathematics*. European Mathematical Society, 2010.

[Owe94] A. B. Owen. Lattice sampling revisited: Monte Carlo variance of means over randomized orthogonal arrays. *Ann. Stat.*, 22:930–945, 1994.

[Owe97a] A. B. Owen. Monte Carlo variance of scrambled net quadrature. *SIAM J. Numer. Anal.*, 34:1884–1910, 1997.

[Owe97b] A. B. Owen. Scrambled net variance for integrals of smooth functions. *Ann. Stat.*, 25:1541–1562, 1997.

[Owe98] A. B. Owen. Latin supercube sampling for very high dimensional simulations. *ACM Trans. Model. Comput. Simul.*, 8:71–102, 1998.

[Owe05] Art B. Owen. On the warnock-halton quasi-standard error, 2005.

[Shr04] S. E. Shreve. *Stochastic Calculus for Finance. Volume II - Continuous Time Models*. Springer-Verlag, New York, 2004.

[Sob67] I.M. Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Journal of Computational Mathematics and Mathematical Physics (English translation)*, 7:784–802, 1967.

[SW98] I. H. Sloan and H. Woźniakowski. When are quasi-Monte Carlo algorithms efficient for high dimensional integrals. *J. Complexity.*, 14:1–33, 1998.

[SW01] I. H. Sloan and H. Woźniakowski. Tractability of multivariate integration for weighted Korobov classes. *J. Complexity.*, 17:697–721, 2001.

[SW02] I. H. Sloan and H. Woźniakowski. When does monte carlo depend polynomially on the number of variables? In H. Niederreiter (Ed.), editor, *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 407–437. Springer-Verlag, Berlin, 2002.

[TWW88] J. F. Traub, G. W. Wasilkowski, and H. Woźniakowski. *Information-Based Complexity*. Academic Press, Boston, 1988.

[Wan03] X. Wang. Strong tractability of multivariate integration using quasi-Monte Carlo algorithms. *Math. Comp.*, 72:823–838, 2003.

[War02] T. Warnock. Effective error estimates for quasi-monte carlo computations, 2002.

[Was04] G. W. Wasilkowski. On polynomial-time property for a class of randomized quadratures. *J. Complexity.*, 20:624–637, 2004.

[Woź00] H. Woźniakowski. Efficiency of quasi-Monte Carlo algorithms for high dimensions. In H. Niederreiter and J. Spanier, editors, *Monte Carlo and Quasi-Monte Carlo Methods 1998*, pages 114–136. Springer-Verlag, Berlin, 2000.

[YH01] R. X. Yue and F. J. Hickernell. Integration and approximation based on scramble sampling in arbitrary dimensions. *J. Complexity.*, 17:881–897, 2001.

[YH05] R. X. Yue and F. J. Hickernell. Strong tractability of integration using scrambled Niederreiter points. *Math. Comp.*, 74:1871–1893, 2005.

[YH06] R. X. Yue and F. J. Hickernell. Strong tractability of quasi-Monte Carlo quadrature using nets for certain Banach spaces. *SIAM J. Numer. Anal.*, 44:2559–2583, 2006.

[Yue99] R. X. Yue. Variance of quadrature over scrambled unions of nets. *Statist. Sinica.*, 9:451–473, 1999.