Dissertation

Random Bits for Quadrature of SDEs

Lukas Mayer

Vom Fachbereich Mathematik der Technischen Universität Kaiserslautern zur Verleihung des akademischen Grades Doktor der Naturwissenschaften (Doctor rerum naturalium, Dr. rer. nat.) genehmigte Dissertation

Erster Gutachter: Prof. Dr. Klaus Ritter
Zweiter Gutachter: Prof. Dr. Andreas Rößler

Datum der Disputation: 08.02.2019

D 386

Acknowledgements

First I want to express my gratitude to my supervisor Prof. Dr. Klaus Ritter for the interesting topic to work on, many helpful discussions, and for giving me the opportunity to write this thesis.

Furthermore, I want to thank Prof. Dr. Andreas Rößler for serving as a referee for this thesis.

My special thanks are to my co-authors Mario Hefter and Mike Giles for many valuable discussions and the opportunity of joint research.

I want to thank my former colleagues at the TU Kaiserslautern, especially Alexander Hunt, Holger Stroot, Saeed Hadjizadeh, Steffen Omland, André Herzwurm, and Thomas Müller for interesting and helpful discussions, as well as, for their contribution to a positive working atmosphere.

Finally, I want to acknowledge the financial support by the Deutsche Forschungsgemeinschaft (DFG) within the RTG 1932 'Stochastic Models for Innovations in the Engineering Sciences'.

Contents

CHAPTER I

Introduction

I.1. Introduction

In this thesis we study a variant of the quadrature problem for stochastic differential equations (SDEs), namely the approximation of expectations $\mathrm{E}(f(X))$, where $X = (X(t))_{t\in[0,1]}$ is the solution of an SDE and $f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}$ is a functional, mapping each realization of X into the real numbers. The distinctive feature in this work is that we consider randomized (Monte Carlo) algorithms with random bits as their only source of randomness, whereas the algorithms commonly studied in the literature are allowed to sample from the uniform distribution on the unit interval, i.e., they do have access to random numbers from $[0,1]$. By assumption, all other operations like, e.g., arithmetic operations, evaluations of elementary functions, and oracle calls to evaluate f are considered within the real number model of computation, i.e., they are carried out exactly. We will refer to algorithms of this type as restricted randomized or restricted Monte Carlo algorithms, and the approximation of expectations by algorithms of this type will be called random bit quadrature.

In the following, we provide a detailed description of the quadrature problem, namely we are interested in the approximation of

$$S(f) = \mathrm{E}(f(X))$$

for X being the r-dimensional solution of an autonomous SDE of the form

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \quad t \in [0,1], \tag{1}$$

with deterministic initial value

$$X(0) = x_0 \in \mathbb{R}^r,$$

and driven by a d-dimensional standard Brownian motion W. Furthermore, the drift coefficient $a\colon \mathbb{R}^r \to \mathbb{R}^r$ and the diffusion coefficient $b\colon \mathbb{R}^r \to \mathbb{R}^{r\times d}$ are assumed to be globally Lipschitz continuous. These conditions on the SDE are also referred to as standard assumptions and it is well-known that they have a (up to indistinguishability) unique pathwise time-continuous square-integrable solution. Moreover, we are interested in functionals

$$f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}$$

that are assumed to be Lipschitz continuous with respect to either the supremum norm or the L_p-norm, for $1 \le p < \infty$, on $C([0,1],\mathbb{R}^r)$, with a respective Lipschitz constant of at most 1. That is, we are interested in the function classes

$$F_\infty = \bigl\{f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}\colon |f(x) - f(y)| \le \|x-y\|_{\sup}\bigr\}$$

and

$$F_p = \bigl\{f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}\colon |f(x) - f(y)| \le \|x-y\|_{L_p([0,1])}\bigr\}$$

for $1 \le p < \infty$.

Such quadrature problems arise, e.g., in financial mathematics, more precisely, in computational finance. Here, SDEs of the form (1), for instance geometric Brownian motions, are used to model stock prizes, and the functional f can be viewed as a pay-off function, e.g., in the context of European options. Note that our setting covers the case of path-dependent options like, e.g., Asian options, which means that we are in particular considering infinite-dimensional quadrature problems. Moreover, SDEs of the form (1) are used in the context of currency exchange rates and interest rates. Popular models, especially for the latter, involve the Ornstein-Uhlenbeck process, which is, e.g., used in the Vasicek model for interest rates, as well as the Cox-Ingersoll-Ross (CIR) process. The CIR process is the solution to a non-standard SDE, that has been introduced in Cox et al. [11] as a model for short-term interest rates, and is nowadays used to model the volatility process in the well-known Heston model, cf. Heston [37], which is used for option pricing.

One main problem concerning the quadrature problem is that the solution process $X = (X(t))_{t\in[0,1]}$ is in general not known explicitly, since X is only given implicitly via the SDE. As a consequence X can not be simulated directly and approximation methods need to be applied. The two most popular approximation methods for strong, i.e., pathwise approximation of X, are the Euler-Maruyama and the Milstein approximation scheme, which we will consider both.

A standard procedure for the approximation of $\mathrm{E}(f(X))$, based on weak approximations of X, is the method of direct simulation, which is also known as the classical Monte Carlo method, and works as follows. Let X_m be the Euler-Maruyama approximate solution of X, see Section II.2.1, based on m equidistant time steps. Furthermore, for $N \in \mathbb{N}$, let $X_{m,1}, \dots, X_{m,N}$ denote N independent copies of X_m. Then

$$A_{m,N}(f) = \frac{1}{N} \cdot \sum_{i=1}^{N} f(X_{m,i}) \tag{2}$$

serves as a suitable approximation of $\mathrm{E}(f(X))$. We are actually interested in the relation between the approximation error, $\mathrm{error}(A, f)$, of an algorithm A applied to a functional f, e.g., the classical Monte Carlo algorithm from above, and the computational cost, $\mathrm{cost}(A, f)$, of A applied to f. As the error criterion we consider the worst case root-mean-squared error of A on a whole class F of functions, i.e.,

$$\mathrm{error}(A, F) = \sup_{f\in F} \big(\mathrm{E}\,|S(f) - A(f)|^2\big)^{1/2}.$$

To define the cost of A applied to f, recall that as the model of computation we choose the real number model. That is, we assume arithmetic operations, comparisons, and evaluations of elementary functions to be carried out at unit cost. The same is assumed for any point evaluation of either the drift or the diffusion coefficient a and b, respectively. Moreover, the cost to evaluate any functional f applied to a piecewise linear function (e.g. the linear interpolation of an Euler-Maruyama approximation) is given by the number of breakpoints of the latter. Finally, we suppose that each call to a random generator (either random numbers from $[0, 1]$ or random bits from $\{0, 1\}$) has also cost one. The cost of A applied to f is now given by the sum of all operations that are carried out when running A. Since this is in general a random quantity, we again consider the worst case of the expected cost on the whole class F, i.e.,

$$\mathrm{cost}(A, F) = \sup_{f\in F} \mathrm{E}(\mathrm{cost}(A, f)).$$

Different algorithms can now be compared in terms of their error-cost-relation. To answer the question to what extent restricted Monte Carlo algorithms are inferior to arbitrary Monte Carlo algorithms, we have to compare their ε-complexity. For $\varepsilon > 0$, the ε-complexity of Monte Carlo algorithms, defined via

$$\operatorname{comp}(\varepsilon, F) = \inf\{\operatorname{cost}(A, F) \colon \text{A is a Monte Carlo algorithm}, \operatorname{error}(A, F) \leq \varepsilon\},$$

is the least cost for any Monte Carlo algorithm to have an error of at most ε. The ε-complexity $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F)$ of the subclass consisting of all restricted Monte Carlo algorithms is defined analogously. As an example, for $F = F_p$, the classical Monte Carlo algorithm based on the Euler-Maruyama approximation schemes yields the upper bound $\operatorname{comp}(\varepsilon, F) \leq c \cdot \varepsilon^{-4}$ for some positive constant c.

The squared error of a Monte Carlo algorithm can be decomposed into

$$(\operatorname{error}(A_{m,N}, f))^2 = \frac{1}{N} \cdot \operatorname{Var}(f(X_m)) + \operatorname{Bias}(A_{m,N}(f))^2,$$

with $\operatorname{Bias}(A_{m,N}(f)) = \mathrm{E}(A_{m,N}(f)) - S(f)$ denoting the bias of the algorithm. Clearly, the bias of $A_{m,N}(f)$ only depends on the choice of the approximation scheme X_m, whereas the variance of $A_{m,N}(f)$ depends on the replication number N, too. Hence it stands to reason to try to improve the algorithm by means of variance reduction techniques. The particular variance reduction technique we are interested in is the multilevel Monte Carlo method, which can be interpreted as an iterative control variate method (described in Section IV.2.1), and which has become a standard technique during the last few years. In the context of SDEs, the multilevel Monte Carlo method has been introduced in Giles [24], and is similar to the multilevel Monte Carlo method introduced in Heinrich [33, 34] and Heinrich and Sindambiwe [35] in the context of parametric integration. A broad introduction to multilevel Monte Carlo algorithms is, e.g., given in the overview article by Giles [25], which also provides several examples from different fields of stochastics, where variants of the classical multilevel Monte Carlo algorithm are applied. A multilevel Monte Carlo method based on the Milstein scheme instead of the Euler-Maruyama scheme, leading to an improved order of convergence of the variance of the multilevel Monte Carlo method, is presented in Giles et al. [23].

The multilevel Monte Carlo method for the quadrature of SDEs, i.e., for the approximation of $\mathrm{E}(f(X))$ operates as follows. First of all, we need a hierarchy $Y_0, Y_1, Y_2, \ldots$ of approximation schemes, converging to X in a suitable way (e.g. in mean-square), e.g., $Y_m = X_{2^m}$ being the sequence of Euler-Maruyama approximation schemes based on 2^m equidistant time steps. For a given $L \in \mathbb{N}$, the idea of the method is to split the approximation $\mathrm{E}(f(Y_L))$, exploiting the linearity of the expectation, into a telescoping sum, where the schemes $Y_0, \ldots, Y_{L-1}$ serve as iterative control variates, i.e.,

$$\mathrm{E}(f(Y_L)) = \mathrm{E}(f(Y_0)) + \sum_{\ell=1}^{L} \mathrm{E}\big(f(Y_\ell) - f(Y_{\ell-1})\big). \tag{3}$$

Now, each expectation on the right hand side is independently approximated by means of a classical Monte Carlo algorithm, cf. (2). The squared error of this method can, again, be decomposed into its squared bias and variance, where the bias depends on the highest level L only (actually we do have the same bias as for the classical Monte Carlo algorithm using the approximation scheme X_{2^L}) and the variance is due to Bienaymé given as the sum of the variances of each classical Monte Carlo approximation, i.e., the sum of the variances $\operatorname{Var}(f(Y_\ell) - f(Y_{\ell-1}))$ respective $\operatorname{Var}(f(Y_0))$ divided by their replication number each. Coupling the consecutive approximations Y_ℓ and $Y_{\ell-1}$ on each level $\ell = 1, \ldots, L$ in a suitable way,

e.g., as presented in Section IV.3.1, we can assure that $\mathrm{Var}(f(Y_\ell) - f(Y_{\ell-1}))$ is decreasing in ℓ while the computational cost clearly increases in ℓ. Hence the overall cost of the algorithm can be minimized with respect to a given error bound $\varepsilon > 0$ by balancing these two effects, see Section IV.3.2 for details.

We take a closer look at the Euler-Maruyama method. Actually, randomness only comes into play on account of the realization of the increments of the driving Brownian motion. Thinking of the inverse CDF method each realization of a Brownian increment corresponds to drawing one random number. Hence, considering random bits instead of random numbers naturally leads to the question how to approximate random variables uniformly distributed on the unit interval, by means of random bits, in order to obtain suitable (random) bit approximations of the involved Brownian increments, and in particular how many random bits should be used for each approximation. A natural ansatz is to consider the binary expansion of $u \in [0,1]$, which reads as

$$u = \sum_{i=1}^{\infty} b_i \cdot 2^{-i}$$

with $(b_i)_{i\in\mathbb{N}} \in \{0,1\}^{\mathbb{N}}$, and to cut it off after finitely many terms. For technical reasons this will lead us to uniform distributions on the sets

$$D^{(p)} = \left\{ \sum_{i=1}^{p} b_i \cdot 2^{-i} + 2^{-(p+1)} \colon b_i \in \{0,1\} \text{ for } i = 1, \dots, p \right\}$$

of dyadic numbers from $[0,1[$ based on $p \in \mathbb{N}$ bits and shifted by $2^{-(p+1)}$, cf. Section III.2. Moreover, it is reasonable to question whether one should use the same number of random bits for, e.g., the Euler approximation X_{2^1} based on 2 equidistant time steps and the Euler approximation $X_{2^{10}}$ based on 2^{10} equidistant time steps, since the approximation error of X_{2^1} is way larger than the corresponding quantity for $X_{2^{10}}$, anyway. This actually leads to adding a further dimension of discretization to the multilevel Monte Carlo method, i.e., we wish to consider time discretization and the number of bits used on each level simultaneously. An idea similar to multi-index Monte Carlo, as introduced in Haji-Ali et al. [31]. To analyze multilevel Monte Carlo algorithms based on random bits, at first, we need an analysis of the impact of random bits, as the only source of randomness, on the approximation of the increments of the driving Brownian motion W. Here, it is desirable to have approximations of the involved Brownian increments that are

1.) independent,
2.) based on finitely many random bits, and
3.) allow a coupling such that the telescoping sum (3) is fulfilled.

Observe that 1.) is a technical assumption for the analysis of the resulting random bit Euler-Maruyama scheme, while 3.) is a technical assumption in the analysis of the classical multilevel Monte Carlo algorithm. Actually, we present different (random bit) approximations of Brownian increments, each satisfying 1.) – 3.) only partially. An algorithm providing all three properties is given in Belomestny and Nagapetyan [3]. This algorithm follows a different approach concerning the analysis of the variances leading to rather complicated distributions on the coarser levels.

The approach of taking into account random bits, as a second dimension of discretization has, e.g., become practically feasible due to the availability of reconfigurable hardware architectures, like Field Programmable Gate Arrays (FPGAs). Such devices allow a user-specified

precision for each individual operation on a bit level, i.e., for each operation the number of bits used for the representation of each involved quantity can be individually specified by the user. Furthermore, a generator for random bits is provided, as well. Clearly, in this context an error analysis in the real number model of computation is no longer reasonable, since round-off errors have to be considered. A round-off error analysis of the Euler-Maruyama scheme, under the assumption of availability of random bit approximations of Brownian increments satisfying 1.) – 3.) is provided in Omland [56]. For the construction and for extensive tests of a finite precision multilevel algorithm for FPGAs, with applications in computational finance, we refer to Brugger et al. [7] and Omland et al. [57].

Moreover, the consideration of different bit numbers as a second dimension of discretization seems to become more interesting in the context of Graphics Processing Units (GPUs), since state-of-the-art GPUs do allow three different standard precisions, i.e., three choices of bit numbers that are supported by the GPU, namely, the standard double (64 bit) and single (32 bit) precision, and on top of that, actually motivated by machine learning, the half (16 bit) precision. This naturally corresponds to a three level (multilevel) Monte Carlo algorithm.

A further motivation for studying restricted Monte Carlo algorithms for the quadrature of SDEs stems from the finite-dimensional counterpart, i.e., the finite-dimensional quadrature problem, which aims at the approximation of $\int_{[0,1]^d} f(x)\,\mathrm{d}x$, and which is studied in, e.g., Gao et al. [22], Heinrich et al. [36], Novak [50, 51, 53], Traub and Woźniakowski [64], Ye and Hu [68]. See Novak and Pfeiffer [54] for a related approach to integral equations. Actually, in Heinrich et al. [36], random bit quadrature with respect to the uniform distribution on $[0,1]^d$ and Sobolev spaces and Hölder classes of functionals $f\colon [0,1]^d \to \mathbb{R}$ have been considered. It is shown, that in this context, restricted Monte Carlo algorithms are as powerful as unrestricted Monte Carlo algorithms, and a very small number of random bits suffices to achieve asymptotic optimality. The proofs of these results are based on a reduction of the quadrature problem to a summation problem and on Bakhvalov's trick.

The main analytical contribution of this thesis concerning the quadrature of SDEs based on random bits is the construction and analysis of a random bit multilevel Euler(-Maruyama) algorithm $A^{q,\mathrm{Bak}}_{\varepsilon,F}$ that is almost optimal, and which relies on approximations of the involved Brownian increments satisfying 1.) and 2.), while 3.) is violated. This leads to an additional error term in the bias, compared to the analysis of the standard multilevel Monte Carlo algorithm, which itself relies on the telescoping sum property (3). Indeed, commonly, this telescoping sum property is fulfilled and to the best of the author's knowledge the only other analyzed algorithm violating this condition is given in Müller et al. [47]. However, this additional error can be controlled due to its dependence on the number of random bits used for each single random bit approximation of a Brownian increment. First of all, we have the following upper bound: For $F \in \{F_p, F_\infty\}$ there exists a positive constant c such that

$$\mathrm{error}\big(A^{q,\mathrm{Bak}}_{\varepsilon,F}, F\big) \leq c \cdot \varepsilon \tag{4}$$

and

$$\mathrm{cost}\big(A^{q,\mathrm{Bak}}_{\varepsilon,F}, F\big) \leq c \cdot \varepsilon^{-2} \cdot \begin{cases} (\ln(\varepsilon^{-1}))^2, & \text{if } F = F_p, \\ (\ln(\varepsilon^{-1}))^3, & \text{if } F = F_\infty, \end{cases} \tag{5}$$

for every $\varepsilon \in \,]0, 1/2[$, see Theorem 12. Observe that these upper bounds on the error and the cost do imply a weak asymptotic upper bound on the ε-complexity $\mathrm{comp}^{\mathrm{res}}(\varepsilon, F)$ that does coincide with the best known upper bound on the corresponding quantity for (arbitrary)

Monte Carlo algorithms, which is actually achieved by the (classical) multilevel Monte Carlo Euler algorithm, cf. Theorem 9 and Corollary 4.

An important ingredient for the construction of $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ is Bakhvalov's trick: A small number of independent random variables, each uniformly distributed on $\{1,\dots,2^q\}$, yields a significantly larger number of pairwise independent random variables, each with the same uniform distribution. The number of random bits used by the algorithm $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ is of the order $\varepsilon^{-2}\cdot(\ln(\varepsilon^{-1})^{5/2}$, see the proof of Theorem 8 in Giles et al. [28], and it can be reduced further to $\varepsilon^{-2}\cdot(\ln(\varepsilon^{-1}))^2\cdot\ln(\ln(\varepsilon^{-1}))$, as outlined in Remark 9 in Giles et al. [28].

Furthermore, the algorithm $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ is optimal, up to logarithmic factors. Indeed, under a slightly stronger smoothness assumption on the coefficients of the SDE under consideration as well as a non-degeneracy assumption on the SDE, particularly excluding pathological cases yielding a deterministic solution X, the following is known to hold true: For $F\in\{F_p,F_\infty\}$ there exist positive constants c and ε_0 such that

$$c\cdot\varepsilon^{-2}\le\operatorname{cost}\bigl(A,F\bigr)$$

for every restricted Monte Carlo algorithm A and for every $\varepsilon\in\,]0,\varepsilon_0]$ with

$$\operatorname{error}\bigl(A,F\bigr)\le\varepsilon.$$

Actually, there are two variants of this result, both of which hold true for a much broader class of algorithms. In the first variant, the evaluation of f is allowed at arbitrary points of $x\in C([0,1],\mathbb{R}^r)$ at unit cost, while the number of random bits is taken into account in the same way as described above, see Giles et al. [28, Theorem 15]. In the second variant, which is due to Creutzig et al. [12, Theorem 11], roughly speaking, any kind of randomness is allowed for free, but the cost model with evaluations of f only at piecewise linear functionals is kept. We do not know whether random bits are as powerful as random numbers for the quadrature problem under investigation, but the upper and lower bounds imply that random bits are at least almost as powerful as random numbers.

Besides its asymptotic optimality in the sense of establishing the same asymptotic upper bounds for the ε-complexity as the classical multilevel Monte Carlo algorithm, denoted by $A_{\varepsilon,F}$, at least for the function classes F_p and F_∞, the random bit algorithm $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ lacks two things. One drawback is that we do have no knowledge about the constant c in (4) and (5). Hence we do not know whether we are inferior to $A_{\varepsilon,F}$ by a big (constant) factor. Consequently, we would like to compare $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ and $A_{\varepsilon,F}$ numerically. Since $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ has been developed and analyzed quite recently, no numerical results are available, so far. Therefore, we consider a variant $A_{\varepsilon,F}^{q}$ of $A_{\varepsilon,F}^{q,\mathrm{Bak}}$, which has the same error as $A_{\varepsilon,F}^{q,\mathrm{Bak}}$, but does not rely on the Bakhvalov trick, leading to a by one logarithmic order worse upper bound on its computational cost, see Theorem 11 and Corollary 5. It turns out that for the examples under consideration, namely the Brownian motion and geometric Brownian motion with f being the maximum, and the Ornstein-Uhlenbeck and Cox-Ingersoll-Ross SDEs with f evaluating at the final time instance, the positive constants c in the upper bounds on the error and cost of $A_{\varepsilon,F}^{q}$ respective $A_{\varepsilon,F}$, cf. Theorem 11 and Theorem 9, only differ by a small factor, less than 10.

The second drawback of $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ respective $A_{\varepsilon,F}^{q}$ is that there is no natural straightforward extension to a, desirable, adaptive random bit multilevel algorithm, as first introduced in Giles [24] for functionals from the class F_p, cf. also Section IV.3.3 for a description of the adaptive multilevel algorithm, due to the choice of the numbers of random bits used by $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ respective $A_{\varepsilon,F}^{q}$ for the (random) bit approximation of the involved Brownian increments.

Such an adaptive multilevel Monte Carlo algorithm has several advantages, discussed in Section IV.3.3, over the non-adaptive algorithms $A^{q,\mathrm{Bak}}_{\varepsilon,F}$ respective $A^{q}_{\varepsilon,F}$, since the latter are, e.g., based on the upper bound on the convergence of the Euler-Maruyama approximation scheme, while the classical adaptive multilevel Monte Carlo algorithm, in the following denoted by $A^{\mathrm{adp}}_{\varepsilon,F}$, estimates this rate of convergence depending on the particular SDE as well as on the employed functional f. This can yield an essential improvement of the performance of the algorithm. For example, for SDEs with additive noise, $A^{\mathrm{adp}}_{\varepsilon,F}$ will see the Milstein rate of convergence, i.e., twice the rate employed by the non-adaptive algorithms $A^{q,\mathrm{Bak}}_{\varepsilon,F}$ and $A^{q}_{\varepsilon,F}$.

Following this reasoning, we present a second restricted multilevel Monte Carlo algorithm, denoted by $A^{\dagger}_{\varepsilon,F}$, which allows an adaptive extension, in line with $A^{\mathrm{adp}}_{\varepsilon,F}$, denoted by $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$. Furthermore, we present and discuss a natural generalization of the adaptive multilevel Monte Carlo algorithms (both the classical and the random bit algorithm) to the class F_∞, since the standard algorithm is fitted to the class F_p. The algorithm $A^{\dagger}_{\varepsilon,F}$ and hence also $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, relies heavily on the Brownian bridge construction and uses a somehow optimized choice of the numbers of random bits depending on the support of the Schauder functions used in the Brownian bridge construction, see Section II.1. Indeed the algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ can not only exploit the adaptivity, but also turns out to be superior to the non-adaptive random bit algorithm $A^{q}_{\varepsilon,F}$, at least for the above mentioned examples, suggesting its general superiority. We stress that we do only investigate $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ numerically, since we neither have analytical results for $A^{\dagger}_{\varepsilon,F}$ nor for $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, so far. This is a consequence of the random bit approximation of the involved Brownian increments by means of the Brownian bridge construction, leading to approximations that satisfy properties 2.) and 3.), whilst property 1.) is violated. That is, we do not have independent approximations of the Brownian increments and hence no standard technique can be applied to analyze the error of the associated random bit Euler-Maruyama approximation schemes.

Furthermore, comparing $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ to its classical counterpart $A^{\mathrm{adp}}_{\varepsilon,F}$ we observe the same weak asymptotics in the relation between their error and cost in terms of ε, respectively, and almost the same strong asymptotics, too. That is, the adaptive random bit algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ is almost as powerful as its classical counterpart $A^{\mathrm{adp}}_{\varepsilon,F}$, at least for all considered examples.

Recall, that as a key problem in the analysis of random bit quadrature problems we have identified the random bit approximation of the involved Brownian increments. This actually leads to the following variant of the quantization problem for probability measures, namely, the optimal approximation of probability measures μ by uniform distributions ν on 2^p support points. Our notion of random bit approximations of probability measures stems from the fact that p random bits suffice to sample any such ν. Let d denote the Wasserstein distance of order two on the set $\mathfrak{M}(H)$ of all probability measures on a separable Hilbert space H, and let $\mathfrak{U}(H,p) \subseteq \mathfrak{M}(H)$ denote the set of all uniform distributions on H with a support of size 2^p. Given $\mu \in \mathfrak{M}(H)$ we study the distance

$$\mathrm{rbit}(\mu,p) = \inf\bigl\{d(\mu,\nu)\colon \nu \in \mathfrak{U}(H,p)\bigr\}$$

between μ and $\mathfrak{U}(H,P)$. In the one-dimensional case $H = \mathbb{R}$ this approximation problem has recently been introduced and thoroughly studied for Wasserstein distances of any order $p \geq 1$ in Xu and Berger [67], Berger and Xu [4], and some of the results from Xu and Berger [67] have been generalized to the Banach space $\mathbb{R}^d$, equipped with the maximum norm, for any $d \in \mathbb{N}$, in Chevallier [9].

Random bit approximation is closely related to quantization, which has been studied intensively for finite-dimensional and infinite-dimensional Banach spaces H. More precisely, let $\mathfrak{F}(H,p)$ denote the set of all Borel probability measures on H with a support of size at most 2^p. Obviously, the quantization number

$$\operatorname{quant}(\mu,p) = \inf\bigl\{d(\mu,\nu)\colon \nu \in \mathfrak{F}(H,p)\bigr\}$$

serves as a lower bound for $\operatorname{rbit}(\mu,p)$, i.e.,

$$\operatorname{quant}(\mu,p) \le \operatorname{rbit}(\mu,p)$$

for every $\mu \in \mathfrak{M}(H)$ and every $p \in \mathbb{N}$. A partial list of references on quantization of probability measures includes the monograph Graf and Luschgy [29] and the survey Dereich [16] as well as Creutzig et al. [12], Dereich [13, 14, 15], Dereich and Scheutzow [17], Dereich et al. [18], Luschgy and Pagès [41, 42], Luschgy and Pagès [43]. We stress that the strong asymptotics of $\operatorname{quant}(\mu,p)$ is studied most of the time in the literature, while we only consider the weak asymptotics of $\operatorname{rbit}(\mu,p)$ in p. Observe, however, that we do lose control of the asymptotic constants in the analysis of the random bit quadrature problem, anyway.

For the one-dimensional standard normal distribution μ we derive

$$\operatorname{rbit}(\mu,p) \asymp 2^{-p/2}\cdot p^{-1/2},$$

see Theorem 4, while $\operatorname{quant}(\mu,p) \asymp 2^{-p}$ according to a known general result for quantization. We stress that this random bit approximation result is actually the basis for our analysis, and in that way also for the construction, of our random bit quadrature algorithms, i.e., our random bit multilevel Monte Carlo algorithms. Let X be a centered Gaussian random element with values in a separable Hilbert space and distribution μ with infinite-dimensional support. Assuming the variances λ_i of the random coefficients in the Karhunen-Loève expansion of X to satisfy

$$\lim_{i\to\infty} \lambda_i \cdot i^{\beta} \cdot (\ln(i))^{\alpha} \in \,]0,\infty[,$$

where $\beta > 1$ and $\alpha \in \mathbb{R}$, we establish

$$\operatorname{rbit}(\mu,p) \asymp p^{-(\beta-1)/2}\cdot (\ln(p))^{-\alpha/2},$$

see Theorem 7 and Corollary 3. For the distribution μ of the standard one-dimensional Brownian bridge on $L_2([0,1])$ we, separately, show that

$$\operatorname{rbit}(\mu,p) \asymp p^{-1/2},$$

see Theorem 5 and Corollary 1. Moreover, for scalar autonomous SDEs we consider the distribution μ of the solution on $L_2([0,1])$, and under mild assumptions on the drift and diffusion coefficients we establish

$$\operatorname{rbit}(\mu,p) \asymp p^{-1/2},$$

see Theorem 6 and Corollary 2. In the latter three cases we only need to establish the upper bound for $\operatorname{rbit}(\mu,p)$, since the matching lower bounds even hold for $\operatorname{quant}(\mu,p)$, according to known results for quantization.

This thesis is organized as follows. In Chapter II we present basic concepts from probability theory that are fundamental throughout the main part of the thesis. Namely, we recall the standard one-dimensional Brownian motion and its Lévy-Ciesielski construction, also known as the Brownian bridge construction. Furthermore, we briefly give the notion of a strong solution to an SDE as well as a sufficient criterion for existence and uniqueness of a strong

solution. Finally, we present the two most popular schemes for the pathwise approximation of a solution, namely the Euler-Maruyama and Milstein approximation schemes.

In Chapter III, more precisely in Section III.1, we first introduce the quantization problem, providing depictive interpretations of the problem, and we present well-known results for (the four) particular quantization problems stated above. Thereafter, in Section III.2, we formulate and study the random bit approximation problem for probability measures. This comes along with four sections that are devoted to the analysis of random bit approximations with respect to the above mentioned probability measures, starting with the one-dimensional standard normal distribution. Here, we also derive some asymptotic properties of the distribution function of the standard normal distribution and its inverse. The random bit approximation of the one-dimensional standard normal distribution will then serve as our building basis for the construction and analysis of the random bit approximation of the distribution of the standard one-dimensional Brownian bridge, the distribution of the solution of a scalar SDE, and Gaussian measures, following this ordering. This chapter is then closed presenting the link of quantization and hence of random bit approximation of probability measures to Kolmogorov widths in Section III.3.

In Chapter IV we formulate the computational problem of random bit quadrature of SDEs. To this end, in Section IV.1, we give meaning to the terminology of an algorithm and we define our error and our cost criterion. In Section IV.2 we then present the method of direct simulation respective the classical Monte Carlo algorithm, as well as, variance reduction by means of the control variate method, leading to the classical Euler-Maruyama based multilevel Monte Carlo algorithm, laid in Section IV.3. Here, we also discuss how to estimate (and hence improve the upper bound on) the convergence of the Euler-Maruyama scheme (for a mesh-size tending to zero) in dependence on the SDE under consideration. This will take us to the adaptive multilevel Monte Carlo Euler-Maruyama algorithm fitted to functionals f from the class F_p, and as a new thing we will introduce a natural extension to the class F_∞. Section IV.4 is then devoted to the construction of the two random bit Euler-Maruyama schemes to be employed in the random bit quadrature of SDEs later, as well as, the strong error analysis of the scheme leading to the random bit multilevel Monte Carlo algorithm, which we will show, in Section IV.5, to have the same (in this sense optimal) weak asymptotic upper bound on the ε-complexity as its classical random number based counterpart.

Finally, Chapter V is devoted to numerical experiments. Here we proceed as follows. We first present and discuss, in Section V.1, our results for the standard one-dimensional Brownian motion in a very detailed manner. We then proceed with the geometric Brownian motion, the Ornstein-Uhlenbeck process, and finally the Cox-Ingersoll-Ross process, in this ordering throughout Sections V.2 to V.4. For each example we present numerical results for the non-adaptive but analyzed random bit algorithm $A^q_{\varepsilon,F}$ and for $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, comparing each algorithm to its classical (random number based) counterpart, and finally comparing the two algorithms themselves. For each example we also present known analytical results that are used to estimate the empirical errors of the algorithms.

The results in Chapter III and Chapter IV are mainly based on Giles et al. [27] and Giles et al. [28], respectively.

I.2. Notations

As long as not stated differently, we denote by $|\cdot|$ the Euclidean norm on $\mathbb{R}^d$. Furthermore, we use the following non-standard notations: For functions $f, g\colon M \to [0,\infty]$ on any set M,

we frequently use the notation $f(m) \preccurlyeq g(m)$, if there exists a positive constant c such that $f(m) \leq c \cdot g(m)$ for every $m \in M$. Moreover, $f(m) \succcurlyeq g(m)$ means $g(m) \preccurlyeq f(m)$ and $f(m) \asymp g(m)$ means $f(m) \preccurlyeq g(m)$ and $g(m) \preccurlyeq f(m)$. In order to mention the set M explicitly, we sometimes say that the corresponding relation holds uniformly in $m \in M$.

Moreover, we write $f(x) \approx g(x)$ as x tends to infinity for two functions $f, g\colon]a, \infty[\to \mathbb{R} \setminus \{0\}$ if

$$\lim_{x\to\infty} f(x)/g(x) = 1.$$

Analogously, we define $f(x) \approx g(x)$ as x tends to 0 from above and x tending to 1 from below, respectively.

CHAPTER II

Basic Facts

In this chapter we present some well-known elementary results that are fundamental for this thesis. First of all, we give the definition of the standard Brownian motion and its relation to the standard Brownian bridge. Furthermore, we describe the Lévy-Ciesielski construction, also known as Brownian bridge construction, of the Brownian motion respective of the Brownian bridge, which we will use to approximate the standard Brownian bridge respective the standard Brownian motion in Section III.2.2 and Section IV.4.2, respectively.

Moreover, we introduce the notion of strong approximation of the solution of a stochastic differential equation. In this context, we first give a sufficient condition for existence and uniqueness of a solution. Finally, we introduce the well-known Euler-Maruyama and Milstein approximation schemes for the pathwise approximation of the solution of a stochastic differential equation. We also state results on their respective rate of convergence to the solution.

II.1. Brownian Motion and Brownian Bridge Construction

Due to the importance of the Brownian motion throughout this thesis, we recall its definition, and briefly introduce the Lévy-Ciesielski representation of the Brownian motion, which will be used for the approximation of Brownian paths in Section III.2.2 and Section IV.4.2. Let $W = (W(t))_{t\in[0,1]}$ denote a stochastic process on a probability space $(\Omega, \mathfrak{A}, P)$ with state space $(\mathbb{R}^d, \mathfrak{B}_d)$. Here, $\mathfrak{B}_d$ denotes the Borel-σ-algebra on $\mathbb{R}^d$. Then W is called a (standard) d-dimensional *Brownian motion* or *Wiener process* if

(i) W has continuous paths,
(ii) $W(0) = 0$ P-almost surely,
(iii) the increments $W(t) - W(s)$ with $0 \leq s < t \leq 1$ are independent from the filtration $\mathfrak{F}_s^W$ generated by W up to time s, and normally distributed with zero expectation and variance $(t-s) \cdot E_d$ with E_d denoting the d-dimensional identity matrix.

In particular, the d-dimensional Brownian motion is a vector of d independent one-dimensional Brownian motions. For existence of Brownian motion we refer to Karatzas and Shreve [39, Section 2.2].

Remark 1. Later on, when we consider solutions of stochastic differential equations, it is in general necessary to have a Brownian motion with respect to a strictly larger filtration than $(\mathfrak{F}_t^W)_{t\in[0,1]}$. Namely, we need a filtration $(\mathfrak{F}_t)_{t\in[0,1]}$ that satisfies the *usual conditions*, i.e., $(\mathfrak{F}_t)_t$ is right-continuous and $\mathfrak{F}_0$ contains all P-null-sets. The latter means that the set $\{A \subseteq \Omega\colon \text{there exists } B \in \mathfrak{A}\colon A \subseteq B \text{ and } P(B) = 0\}$ is contained in $\mathfrak{F}_0$.

For the construction of such a (augmented) filtration $(\mathfrak{F}_t)_t$ from $(\mathfrak{F}_t^W)_t$ we refer to Karatzas and Shreve [39, Definition II.7.2, Theorem II.7.9].

Remark 2. We present the *Lévy-Ciesielski construction* or *Brownian bridge construction* of the Brownian motion, cf. Karatzas and Shreve [39, Section II.3]. Let us start with a one-dimensional Brownian motion. The idea is to construct a so-called Schauder basis of

the space of continuous functions over the unit interval vanishing at 0, $C_0([0,1]) = \{f \in C([0,1])\colon f(0) = 0\}$, and to construct a Brownian motion as an infinite linear combination of the basis elements.

To this end, we first introduce the so-called Haar wavelets, see Haar [30], which define a Schauder basis in $L_p([0,1])$, $1 \le p < \infty$, even an orthogonal basis for $p = 2$. We define $h_{0,1} = 1$ and

$$h_{i,j} = 2^{(i-1)/2} \cdot \left(1_{I_{i,j}} - 1_{J_{i,j}}\right)$$

for $i \in \mathbb{N}$ and $j = 1, \dots, 2^{i-1}$, where

$$I_{i,j} = \left[(j-1)/2^{i-1}, (j-1/2)/2^{i-1}\right[$$

and

$$J_{i,j} = \left[(j-1/2)/2^{i-1}, j/2^{i-1}\right[.$$

Next, we need to transform the Haar wavelets into continuous functions in order to obtain a Schauder basis of $C_0([0,1])$. This is achieved by the Schauder functions

$$s_{i,j}(t) = \int_0^t h_{i,j}(u)\,\mathrm{d}u, \qquad t \in [0,1], \tag{6}$$

where $i = 0$ and $j = 1$ or $i \in \mathbb{N}$ and $j = 1, \dots, 2^{i-1}$. Note, that $s_{0,1}$ is a linear function on the unit interval and the remaining Schauder functions are hat functions on the respective subintervals $[(j-1)/2^{i-1}, j/2^{i-1}[$ and constantly zero everywhere else. Hence, they can be interpreted as a local refinement on the corresponding subintervals. We point out that this system of Schauder functions can also be interpreted as a rescaled Faber-Schauder system.

Finally, let $(Y_{i,j})_{i,j}$ with $i = 0$ and $j = 1$ or $i \in \mathbb{N}$ and $j = 1, \dots, 2^{i-1}$ be an independent family of standard normally distributed random variables. Then the Lévy-Ciesielski representation states that

$$W_\ell(t) = W_\ell(t,\omega) = \sum_{i=0}^{\ell} \sum_{j=1}^{2^{max(i-1,0)}} s_{i,j}(t) \cdot Y_{i,j}(\omega) \tag{7}$$

converges to a Brownian motion as $\ell \to \infty$ with uniform convergence for almost every $\omega \in \Omega$, see Karatzas and Shreve [39, Lemma II.3.1, Theorem II.3.2]. In particular, we have convergence with respect to the L_2-norm. Since each Brownian motion W can be represented in that way, we will use the notation

$$W(t) = \sum_{i=0}^{\infty} \sum_{j=1}^{2^{max(i-1,0)}} s_{i,j}(t) \cdot Y_{i,j}.$$

To obtain the analogous result for the d-dimensional Brownian motion, it suffices to replace the independent family $Y_{0,1}, Y_{1,1}, \dots$ of standard normally distributed random variables by independent d-dimensional random vectors, each of which is standard normally distributed.

Though the notation of the Lévy-Ciesielski representation as an iterated sum might seem unnecessary complicated, it will turn out to be a natural notation when we talk about approximations of the Brownian motion W, see Section III.2.2 and Section IV.4.2.

REMARK 3. Let us consider the relation of the (standard) one-dimensional Brownian motion W and the (standard) one-dimensional Brownian bridge, denoted by B. The main difference between the two stochastic processes is that the Brownian bridge B has to be zero at the final time instance, here $B(1) = 0$. Formally, the Brownian bridge B can be derived from the Brownian motion W by setting

$$B(t) = W(t) - t \cdot W(1), \qquad t \in [0,1],$$

see, e.g., Kallenberg [38, p.203]. In particular the process B is independent from $W(1)$ and $W(1)$ is standard normally distributed. This relation can be incorporated in the Lévy-Ciesielski representation of W, see (7), to obtain an analogous representation for B. Since the first summand $s_{0,1} \cdot Y_{0,1}$ is the linear interpolation of a standard normally distributed random variable on the interval $[0,1]$, we have

$$B(t) = W(t) - t \cdot W(1) \stackrel{\mathrm{d}}{=} W(t) - s_{0,1}(t) \cdot Y_{0,1} = \sum_{i=1}^{\infty} \sum_{j=1}^{2^{i-1}} s_{i,j}(t) \cdot Y_{i,j} \tag{8}$$

with the same mode of convergence as for $W(t)$, cf. (7). The right hand side is also called *Lévy-Ciesielski representation of the (standard) Brownian bridge*.

REMARK 4. Let us briefly comment on the terminology *Brownian bridge construction*. Let B denote a standard Brownian bridge on the unit interval. The starting knowledge for the construction is $B(0) = B(1)$. This approximation of B can be extended to the unit interval by linear interpolation, yielding the function that is constantly zero on $[0,1]$. In the first step, which corresponds to $i = 1$ in (8) we assure to hit the right distribution at the midpoint $1/2$, which we know to be $N(0, 1/2)$, and we assure to have a continuous approximation of B on $[0,1]$. Both is given by $s_{1,1} \cdot Y_{1,1}$, where $s_{1,1}$ provides a linear interpolation on $[0, 1/2]$ and $[1/2, 1]$, respectively. Therefore, the Schauder functions are sometimes also called *hat functions*, and from a geometrical point of view $s_{1,1} \cdot Y_{1,1}$ can be interpreted as a bridge from 0 to 0 with its peak at $t = 1/2$. In the second step, which corresponds to $i = 2$ in (8), the analogous construction is (separately) applied to the subintervals $[0, 1/2]$ and $[1/2, 1]$ with the difference that the value in the boundary point $1/2$ is no longer zero but $N(0, 1/2)$-distributed. From a geometrical point of view this can again be interpreted as two local bridges. Likewise in the i-th step, we add bridges on the subintervals $[(k-1) \cdot 2^{-i+1}, k \cdot 2^{-i+1}]$ with $k = 1, \ldots, 2^{i+1}$.

Finally, we present a well-known result for the standard Brownian bridge that we will need in the analysis in Section III.2.3.

LEMMA 1. *Let B be a standard Brownian bridge on $[0,1]$. Then we have*

$$\mathrm{E}\, \|B\|^2_{L_2([0,1])} < \infty.$$

PROOF. By Remark 3, B can be expressed in terms of a standard Brownian motion W

$$B(t) = W(t) - t \cdot W(1)$$

and B is independent from $W(1)$. Hence, using Jensen's inequality and Fubini's rule we have

$$1/2 \cdot \mathrm{E}\, \|B\|^2_{L_2([0,1])} \leq \int_0^1 \mathrm{E}(W(t))^2 \, \mathrm{d}t + \mathrm{E}(W(1))^2 \cdot \int_0^1 t^2 \, \mathrm{d}t = 1/2 + 1/3 = 5/6. \qquad \blacksquare$$

II.2. Strong Approximation of SDEs

Let $r, d \in \mathbb{N}$. Let $(\Omega, \mathfrak{A}, P)$ be a complete probability space and let $\mathfrak{F} = (\mathfrak{F}_t)_{t\in[0,1]}$ be a filtration on $(\Omega, \mathfrak{A}, P)$ that satisfies the usual conditions, cf. Remark 1. Furthermore, let W be a d-dimensional Brownian motion with respect to the filtration $\mathfrak{F}$. We consider an autonomous system

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \qquad t \in [0,1] \tag{9}$$

of SDEs with deterministic initial value

$$X(0) = x_0 \in \mathbb{R}^r$$

and driving Brownian motion W, as well as, drift coefficient $a\colon \mathbb{R}^r \to \mathbb{R}^r$ and diffusion coefficient $b\colon \mathbb{R}^r \to \mathbb{R}^{r\times d}$.

Theorem 1. Existence and uniqueness of the solution. *Let a and b be globally Lipschitz functionals, i.e., there exists a constant $c > 0$ such that*

$$|a(x) - a(y)| \le c \cdot |x - y| \qquad \wedge \qquad |b(x) - b(y)| \le c \cdot |x - y| \tag{10}$$

for all $x, y \in \mathbb{R}^r$. Then there exists a (up to indistinguishability) unique t-continuous strong solution $X(t)$,i.e., a solution with continuous sample paths, which is adapted to the filtration $\mathfrak{F}$, and that satisfies $\mathrm{E}(\int_0^1 |X(t)|^2\,\mathrm{d}t) < \infty$.

For a proof, see, e.g., Øksendal [55, Theorem 5.2.1] or Mao [44, Theorem 2.3.1]. Note, that the latter only shows that the solution $X(t)$ is in $L_2([0,1], \mathbb{R}^d)$.

Remark 5. Theorem 1 can easily be generalized in the following ways.

(i) In the case of autonomous SDEs, the global Lipschitz condition on the coefficients a and b of the SDE is often replaced by the corresponding local Lipschitz condition in combination with the linear growth condition, i.e., there exists a constant $c > 0$ such that for all $x \in \mathbb{R}^r$ there exist neighborhoods $U_x, \widetilde{U}_x \subseteq \mathbb{R}^r$ such that

$$|a(x) - a(y)| \le c \cdot |x - y| \qquad \wedge \qquad |b(x) - b(\tilde{y})| \le c \cdot |x - \tilde{y}|$$

for all $y \in U_x$ and for all $\tilde{y} \in \widetilde{U}_x$, respectively. Furthermore,

$$|a(x)|^2 \le c \cdot (1 + |x|^2) \qquad \wedge \qquad |b(x)|^2 \le c \cdot (1 + |x|^2)$$

for all $x \in \mathbb{R}^r$.

Obviously, the linear growth condition is implied by global Lipschitz continuity. For $f = a$ and $f = b$ we have

$$|f(x)| \le 2 \cdot |f(x) - f(0)| + 2 \cdot |f(0)|.$$

(ii) The results for the compact interval $[0,1]$ carry over to general compact intervals $[T_0, T_1]$.

Having conditions for existence and uniqueness of the solution to equation (9), we need to address the question of computing this solution. Unfortunately, in general, no explicit formula of the solution is known. Therefore, the solution is usually only approximated. However, in some cases the solution is known explicitly, e.g. in the linear case, for which we refer to Mao [44, Chapter 3]. Note, that when we talk about approximating the solution of an SDE, we always think of strong approximation, i.e., pathwise approximation.

In the following we will briefly introduce and discuss the well-known Euler-Maruyama scheme for the approximation of the solution of a system of SDEs, as well as, the Milstein

scheme for the approximation of the solution of a scalar SDE, i.e., $r = d = 1$. Let us mention that it is also possible to define the Milstein scheme for a system of SDEs, but that would lead to Lévy areas and their approximation, what is not in the scope of this work.

II.2.1. Euler-Maruyama Scheme. We will only consider the deterministic equidistant Euler-Maruyama scheme, i.e., with à priori determined breakpoints

$$t_k = t_{k,m} = k/m,$$

where $m \in \mathbb{N}$ and $k = 0, \dots, m$. The corresponding approximation scheme is given by

$$\begin{aligned} X_m(t_{0,m}) &= x_0, \\ X_m(t_{k,m}) &= X_m(t_{k-1,m}) + m^{-1} \cdot a\big(X_m(t_{k-1,m})\big) + b\big(X_m(t_{k-1,m})\big) \cdot V_{k,m} \end{aligned}$$

for $k = 1, \dots, m$, with Brownian increments

$$V_k = V_{k,m} = W(t_{k,m}) - W(t_{k-1,m}).$$

In order to approximate the solution X at any point $t \in [0,1]$, we extend $X_m(t_0), \dots, X_m(t_m)$ by linear interpolation onto the subintervals $]t_{k-1}, t_k[$ for $k = 1, \dots, m$.

The following result is well-known, see, e.g., Creutzig et al. [12, p. 419], cf. also Mao [44, Theorem 2.7.3] for a similar result.

THEOREM 2. *Assuming that the global Lipschitz condition (10) from Theorem 1 is satisfied, the Euler-Maruyama approximate solutions X_m of X satisfy*

$$\Big(\mathrm{E}\Big(\sup_{0\le t\le 1} |X_m(t) - X(t)|^2\Big)\Big)^{1/2} \preccurlyeq m^{-1/2} \cdot (\log(m+1))^{1/2},$$

as well as,

$$\Big(\mathrm{E}\,\|X_m - X\|^2_{L_p([0,1])}\Big)^{1/2} \preccurlyeq m^{-1/2}.$$

For more details on the constants, see, e.g., Mao [44, Theorem 2.7.3]. The constants depend on the considered norm, the time interval $[0,1]$, the initial value x_0 of the SDE (9) and the Lipschitz constants of the coefficients a and b.

Regarding the convergence rate in Theorem 2 one may as for ordinary differential equations (ODEs) take a look at higher order approximation schemes.

II.2.2. Milstein Scheme. In this section, we assume $r = d = 1$, i.e., equation (9) is a scalar SDE, as well as, differentiability of the diffusion coefficient b. As for the Euler-Maruyama scheme, we will only consider the deterministic Milstein scheme with equidistant breakpoints

$$t_k = t_{k,m} = k/m, \qquad k = 0, \dots, m,$$

where $m \in \mathbb{N}$. Denoting the associated Brownian increments by

$$V_k = V_{k,m} = W(t_{k,m}) - W(t_{k-1,m}),$$

the corresponding approximation scheme reads as

$$\begin{aligned} \widehat{X}_m(t_{0,m}) &= x_0, \\ \widehat{X}_m(t_{k,m}) &= \widehat{X}_m(t_{k-1,m}) + m^{-1} \cdot a\big(\widehat{X}_m(t_{k-1,m})\big) + b\big(\widehat{X}_m(t_{k-1,m})\big) \cdot V_{k,m} \\ &\qquad + \tfrac{1}{2} \cdot (b \cdot b')\big(\widehat{X}_m(t_{k-1,m})\big) \cdot m^{-1}\big((m^{1/2} \cdot V_{k,m})^2 - 1\big) \end{aligned}$$

for $k = 1, \dots, m$.

Again, we extend $\widehat{X}_m(t_0), \dots, \widehat{X}_m(t_m)$ by linear interpolation onto the subintervals $]t_{k-1}, t_k[$ for $k = 1, \dots, m$.

The following result is well-known, see, e.g., Milstein and Tretyakov [45, Theorem 1.2.4].

THEOREM 3. *Assuming that the global Lipschitz condition* (10) *from Theorem 1 is satisfied as well as the diffusion coefficient b having a bounded Lipschitz continuous derivative, there exists a positive constant c such that the Milstein approximate solutions $\widehat{X}_m$ of X satisfy*

$$\Big(\mathrm{E}\Big(\max_{0 \le k \le m} \big| \widehat{X}_m(t_k) - X(t_k) \big|^2 \Big) \Big)^{1/2} \le c \cdot m^{-1}.$$

REMARK 6. Note, that for additive noise, i.e., the diffusion coefficient b is independent from X, the derivative of b is constantly zero. Therefore, in this case, the Milstein approximate solutions $\widehat{X}_m$ and the Euler-Maruyama approximate solutions X_m do coincide for all $m \in \mathbb{N}$.

Moreover, in Lemma 11, we will observe that the Milstein scheme interpolated with suitable rescaled Brownian bridges has convergence rate 1 not only in the breakpoints t_k but on the whole interval $[0, 1]$. For additive noise, this upper bound on the convergence rate carries over to a correspondingly modified Euler-Maruyama scheme.

CHAPTER III

Approximation of Probability Distributions

Random bit approximation of probability distributions is very closely related to the thoroughly studied *quantization of probability distributions*. Actually, we will see that we exploit some results, especially lower bounds, from the quantization problem for our random bit approximation problem. Therefore, we will first introduce the quantization problem and state some well-established results from quantization theory.

The chapter is outlined as follows. In Section III.1 we introduce and discuss the quantization of probability distributions. We start with the finite dimensional case, namely with the space of real numbers $\mathbb{R}$ and some generalizations to the d-dimensional space $\mathbb{R}^d$. This will allow us to give some vivid interpretation of the quantization problem. In particular we present the relations to the *n-centers* problem and to *Voronoi partitions*. Thereafter, we present results for the standard normal distribution followed by results for the standard Brownian motion and the standard Brownian bridge, as well as for scalar SDEs.

In Section III.2 we introduce our notion of random bit approximations of probability distributions and we discuss the relation to the quantization problem. Actually, we will see that random bit approximations can be interpreted as a restriction of the quantization problem to a subclass of probability measures with a finite support size. Here, we proceed in the following way. At first, in Section III.2.1, we present, analyze and discuss a random bit approximation of the standard normal distribution. Moreover, we relate our results to recent results in Xu and Berger [67]. Thereafter, in Section III.2.2 we analyze a random bit approximation of the distribution of the one-dimensional standard Brownian bridge in $L_2([0,1])$ based on the random bit approximation of the standard normal distribution from Section III.2.1, implicitly covering the random bit approximation of the finite-dimensional Brownian bridge via componentwise application of the described one-dimensional approximation method. Moreover, in Section III.2.3 we analyze a random bit approximation of the distribution of a scalar SDE under a standard global Lipschitz assumption on the drift coefficient as well as a global Lipschitz assumption on the derivative of the diffusion coefficient. Again the approximation will be based on the random bit approximation of the standard normal distribution from Section III.2.1. We restrict ourselves to the scalar case since we will need to employ the Milstein approximation scheme, cf. Section II.2.2, in order to get the same rate of convergence as for the corresponding quantization problem. In the last section, namely Section III.2.4 we discuss the quantization and random bit approximation of Gaussian random elements. The latter will once more be based on the random bit approximation of the standard normal distribution from Section III.2.1. Gaussian random elements are discussed separately since, in contrast to the afore discussed random bit approximations, they rely on the Karhunen-Loève expansion of the Gaussian random elements. Furthermore, they are not in the main scope of this thesis. In particular they will neither be treated numerically nor will we introduce any quadrature method based on the Karhunen-Loève expansion.

Finally, in Section III.3 we give the relation of quantization and random bit approximation of probability distributions to *Kolmogorov widths*.

The results on the random bit approximations discussed in Section III.2 are published in Giles et al. [27].

III.1. Quantization

The terminology *quantization* stems from the theory of signal processing, more precisely, from the context of pulse-code modulation, where analog sound signals are translated into a digital representation. The digitalization of the analog data requires to *quantize* the signal on an appropriate grid of time points (i.e., it can be considered as the division of a quantity into a discrete number of small parts) such that the original signal can either be reconstructed perfectly or at least with a, in some sense, small error.

Likewise, as a mathematical problem, quantization of probability distributions addresses the approximation of a d-dimensional probability distribution μ by a d-dimensional probability distribution $\widehat{\mu}$ with finite support, i.e., a probability distribution based on only finitely many supporting points. In terms of a d-dimensional random element X, which is often only described implicitly by means of a probability distribution, one is looking for a suitable discrete approximation $\widehat{X}$, i.e., taking only finitely many values from the image space of X, that can actually be simulated.

In the following, we will introduce some basic concepts of quantization for distributions on $\mathbb{R}$, the link to the n-centers problem (for n a natural number), which allows a convenient geometrical interpretation, and results for the weak asymptotics of particular probability distributions as n tends to infinity. In particular for the one-dimensional standard normal distribution. Note, that the results carry over to $\mathbb{R}^d$ in a straightforward way. As a main reference we use Graf and Luschgy [29], in particular the first chapter therein. Thereafter we will consider the more general infinite-dimensional Hilbert space $L_2([0,1])$ and the quantization of the distribution of a Brownian bridge on this space. As a reference we will use Luschgy and Pagès [41]. Finally we deal with the quantization of the distribution of the solution of a scalar SDE on $L_2([0,1])$, cf. (9) with $r = d = 1$. Here, Müller-Gronbach and Ritter [48] will serve as a reference. We stress, that the quantization of the standard normal distribution is the most crucial part. That is because it is the only probability distribution needed for the Brownian bridge construction, as described in Section II.1. Moreover, Brownian increments are the only source of randomness in the class of (scalar) SDEs that we are going to consider, cf. (9).

The first thing we need to do is to quantify the quality of an approximation of a probability distribution. As mentioned before, we start with the space of real numbers $\mathbb{R}$, and we equip it with the Euclidean norm. We point out, that due to norm-equivalences this is no severe restriction when considering weak approximations. Furthermore, we denote by $\mathfrak{M}(\mathbb{R})$ the set of all Borel probability measures on $\mathbb{R}$, which have a finite second moment. The distance between two probability measures μ_1 and μ_2 from $\mathfrak{M}(\mathbb{R})$ is measured by the *Wasserstein distance of order two*, i.e.,

$$d(\mu_1,\mu_2) = \inf\Big\{\big(\mathrm{E}\,|X_1 - X_2|^2\big)^{1/2} \colon P_{X_1} = \mu_1, P_{X_2} = \mu_2\Big\}. \tag{11}$$

We stress that X_1 and X_2 need to be defined on a common probability space. Moreover, both take values in $\mathbb{R}$ and P_{X_1} and P_{X_2} denote the distributions of X_1 and X_2, respectively. Sometimes, in particular in Graf and Luschgy [29], d is called the *Wasserstein-Kantorovitch L_2-metric*. (In fact the metric axioms are a consequence of the separability of the underlying space $\mathbb{R}$.) We mention that, more general, one can consider Wasserstein distances of order

r with $1 \leq r < \infty$, assuming the corresponding finite r-th moment of the involved probability measures. For the discussion and result that the Wasserstein distance is appropriate for studying the quantization problem, we refer to Pollard [58], and especially Pollard [58, Theorem III.3].

Now, for $n \in \mathbb{N}$ we can define the n-th (minimal) quantization error of a given probability measure $\mu \in \mathfrak{M}(\mathbb{R})$. Let

$$\mathfrak{F}_n(\mathbb{R}) = \big\{\nu \in \mathfrak{M}(\mathbb{R})\colon \text{ there exist } x_1, \ldots, x_n\colon \nu(\{x_1, \ldots, x_n\}) = 1\big\}.$$

Then $\mathfrak{F}_n(\mathbb{R})$ is the set of all probability measures on $\mathbb{R}$ with a support of size at most n. The *n-th quantization error of μ* is defined by

$$\operatorname{quant}_n(\mu) = \inf\big\{d(\mu, \nu)\colon \nu \in \mathfrak{F}_n(\mathbb{R})\big\}. \tag{12}$$

One can also find the following definition of the n-th quantization error in terms of a random variable X with distribution μ,

$$\operatorname{quant}_n(X) = \inf\Big\{\big(\mathrm{E}\,|X - f(X)|^2\big)^{1/2}\colon f\colon \mathbb{R} \to \mathbb{R} \text{ measurable}, |f(\mathbb{R})| \leq n\Big\}. \tag{13}$$

The mappings f in (13) are often referred to as *n-quantizers*. An n-quantizer for which the infimum in (13) is attained, is called an *n-optimal quantizer*.

The equivalence of (12) and (13) is covered by Graf and Luschgy [29, Lemma 3.4]. Later, in particular in Section III.2, we will stick to the first definition, (12), nevertheless the second definition, (13), is helpful in the upcoming comparison to the *n-centers problem* and *Voronoi partitions of* $\mathbb{R}$.

Let X be a random variable, taking values in $\mathbb{R}$, with probability distribution $\mu \in \mathfrak{M}(\mathbb{R})$. The *$n$-centers problem of order two* is given by

$$\operatorname{centers}_n(X) = \inf\Big\{\Big(\mathrm{E}\Big(\min_{a\in\alpha} |X - a|^2\Big)\Big)^{1/2}\colon \alpha \subseteq \mathbb{R}, |\alpha| \leq n\Big\}. \tag{14}$$

A set $\alpha \subseteq \mathbb{R}$ for which the infimum in (14) is attained, is called an *n-optimal set of centers for μ*.

To understand why this is called n-centers problem, let us take a look to *Voronoi partitions of* $\mathbb{R}$. Let $\alpha \subseteq \mathbb{R}$ be countable and closed. Then $\{V(a|\alpha)\colon a \in \alpha\}$ with

$$V(a|\alpha) = \Big\{x \in \mathbb{R}\colon |x - a| = \min_{b\in\alpha} |x - b|\Big\} \tag{15}$$

is called *Voronoi diagram of α*, where $V(a|\alpha)$ is the set of all points $x \in \mathbb{R}$ having a as a nearest point in α. Next, we consider a Borel measurable partition $\{A_a\colon a \in \alpha\}$ of $\mathbb{R}$. If there exists a Borel probability measure $\mu \in \mathfrak{M}(\mathbb{R})$ such that

$$A_a \subseteq V(a|\alpha) \quad \mu\text{-almost-surely}$$

for every $a \in \alpha$, then $\{A_a\colon a \in \alpha\}$ is called a *Voronoi partition of $\mathbb{R}$ with respect to α*. The existence of Voronoi partitions with respect to $\alpha \subseteq \mathbb{R}$ is a consequence of Graf and Luschgy [29, Proposition 1.1]. One advantage of Voronoi partitions is that they allow the following geometric interpretation. The elements from a Voronoi partition of $\mathbb{R}$ are closed intervals, and for Voronoi partitions of $\mathbb{R}^d$, to which the definitions and results from the one-dimensional case carry over, see Graf and Luschgy [29, Section I.1], the elements of a Voronoi partition with respect to α are star shaped with respect to the respective $a \in \alpha$, see Graf and Luschgy [29, Proposition 1.2].

We take a look at the link between Voronoi partitions of $\mathbb{R}$ and the n-centers problem. Indeed a set of n centers $\alpha \subseteq \mathbb{R}$ generates a Voronoi partition of $\mathbb{R}$ with respect to some

$\mu \in \mathfrak{M}(\mathbb{R})$. The elements from α are the *centers* of this Voronoi partition in the sense of (15). Conversely, a Voronoi partition of $\mathbb{R}$ with respect to μ, consisting of up to n elements, yields a set of n-centers $\alpha \subseteq \mathbb{R}$.

Now, that we have this geometric interpretation of the n-centers problem at hand, we need to establish the connection to the n-quantization problem. In fact, for a random variable X with distribution $\mu \in \mathfrak{M}(\mathbb{R})$, Graf and Luschgy [29, Lemma 3.1] yields

$$\operatorname{quant}_n(\mu) = \operatorname{centers}_n(X) \tag{16}$$

for all $n \in \mathbb{N}$. Here, we actually see the advantage of the reformulation (13) of the n-quantization problem in terms of a random variable. Indeed, comparing (13) and (14), the proof essentially reduces to the following two constructions. For $\alpha \subseteq \mathbb{R}$ with $|\alpha| \leq n$ we choose a Voronoi partition $\{A_a \colon a \in \alpha\}$ and set $f = \sum_{a\in\alpha} a \cdot 1_{A_a}$ in (13). Conversely, for a given function f taking at most n different values in $\mathbb{R}$, we define α as the image of f in (14).

This completes our characterization of the n-th quantization error $\operatorname{quant}_n(\mu)$ of Borel probability measures $\mu \in \mathfrak{M}(\mathbb{R})$, and we turn to the asymptotic behaviour of $\operatorname{quant}_n(\mu)$, as n tends to infinity.

Example 1. As an elementary example we first consider μ being the uniform distribution on the unit interval. For $n \in \mathbb{N}$, an n-optimal set of centers is given by $\alpha_n = \{\frac{2i-1}{2n} \colon i = 1, \dots, n\}$, see Graf and Luschgy [29, Example 4.17]. Let X be a random variable with distribution μ. An elementary computation shows

$$\mathrm{E}\Big(\min_{a\in\alpha_n} |X-a|^2\Big) = \sum_{i=1}^{n} \int_{\left[\frac{2i-2}{2n}, \frac{2i}{2n}\right]} \big(x - \tfrac{2i-1}{2n}\big)^2 = (4\cdot 3)^{-1} \cdot n^{-1}.$$

By the characterization (16) of the n-th quantization error, we obtain

$$\operatorname{quant}_n(\mu) = (2\cdot\sqrt{3})^{-1} \cdot n^{-1}$$

for all $n \in \mathbb{N}$.

The following result, concerning the quantization of the standard normal distribution, is due to Bucklew and Wise [8, Theorem 2]. It is also covered by Graf and Luschgy [29, Theorem 6.2]. For sake of completeness let us mention that also strong asymptotic results are available, see, e.g., Graf and Luschgy [29, p.124] for the quantization coefficient for the standard normal distribution. Furthermore, the result on the weak asymptotics holds in general for d-dimensional probability measures, that are absolutely continuous with respect to the d-dimensional Lebesgue measure, and which have a finite $2+\varepsilon$ moment for some $\varepsilon > 0$.

Proposition 1. *Let μ be the standard normal distribution. Then we have*

$$\operatorname{quant}_n(\mu) \asymp n^{-1}.$$

The construction of a sequence of n-quantizers for the standard normal distribution μ, in terms of a random variable X with $P_X = \mu$, is, e.g., given in Müller-Gronbach and Ritter [48, Section 3.1].

Remark 7. We give yet another depictly interpretation of the n-quantization problem, see (12) and (13), which in particular applies for the standard normal distribution. Let X

be a random variable with distribution $\mu \in \mathfrak{M}(\mathbb{R})$ and cumulative distribution function F. By F^{-1} we denote the inverse distribution function defined by

$$F^{-1}(u) = \inf\{x \in \mathbb{R} \colon F(x) \geq u\}, \qquad u \in [0,1],$$

which may formally be set to $\pm\infty$ in the boundary points. It is well-known that for U uniformly distributed on the unit interval, $F^{-1}(U)$ has the same distribution as X, i.e., $F^{-1}(U)$ is distributed according to μ, see Proposition 6 in the Appendix. Consequently, taking into account (16) and its discussion, the n-quantization problem of μ respective X (taking up to n values from the image of X) corresponds to the determination of up to n points α from $[0,1]$ such that the set $F^{-1}(\alpha)$ solves the n-centers problem, see (14), i.e., $F^{-1}(\alpha)$ is an n-optimal set for μ. In this sense, we are looking for an optimal knot selection for a piecewise constant approximation of the function F^{-1} on the unit interval.

With Proposition 1 at hand, we turn to the quantization in the infinite-dimensional Hilbert space $L_2([0,1])$ equipped with the standard $L_2([0,1])$-norm. At first, we fix the notation. Actually, the definitions (11) and (12) immediately carry over to $L_2([0,1])$, i.e., the Wasserstein distance of order two is defined by

$$d(\mu_1,\mu_2) = \inf\Big\{\big(\mathrm{E}\,\|X_1 - X_2\|^2_{L_2([0,1])}\big)^{1/2} \colon P_{X_1} = \mu_1, P_{X_2} = \mu_2\Big\} \tag{17}$$

for Borel probability measures $\mu_1, \mu_2 \in \mathfrak{M}(L_2([0,1]))$, as well as, the n-th quantization error of $\mu \in \mathfrak{M}(L_2([0,1]))$ is defined by

$$\mathrm{quant}_n(\mu) = \inf\Big\{d(\mu,\nu) \colon \nu \in \mathfrak{F}_n(L_2([0,1]))\Big\}, \tag{18}$$

where

$$\mathfrak{F}_n(L_2([0,1])) = \big\{\nu_f \in \mathfrak{M}(L([0,1])) \colon f \colon \{1,\dots,n\} \to L_2([0,1]) \text{ measurable}\big\}$$

is the set of all probability measures on $L_2([0,1])$ with a support size of at most n.

We first consider the Wiener measure on $L_2([0,1])$. For the following result, see, e.g., Dereich et al. [18, Example 5.1].

PROPOSITION 2. *For the Wiener measure $\mu \in \mathfrak{M}(L_2([0,1]))$ it holds*

$$\mathrm{quant}_n(\mu) \asymp (\log(n))^{-1/2},$$

i.e., the quantization error converges only with a logarithmic rate, in contrast to the polynomial rate we have for the standard normal distribution, cf. Proposition 1.

Actually, Dereich et al. [18, Example 5.1] even provides tight bounds for the quantization coefficient

$$\lim_{n\to\infty} (\log(n))^{1/2} \cdot \mathrm{quant}_n(\mu) \in \big[1/\sqrt{8}, 1\big].$$

This result is also valid for Wasserstein distances of order r with $1 \leq r < \infty$. For the Wasserstein distance of oder two, it has been shown (in an information theoretical setting) in Binia et al. [6, Example 1], that the lower bound can be improved to $\sqrt{2}/\pi$, which is slightly better. For the general setting of the space $L_p([0,1])$ we refer to Dereich et al. [18, Example 5.1].

For sake of completeness, let us mention that there is a similar result if the space $L_2([0,1])$ is replaced by the space of continuous functions $C([0,1])$, equipped with the supremum norm. Here, the Wiener measure μ satisfies

$$\lim_{n\to\infty} (\log(n))^{1/2} \cdot \mathrm{quant}_n(\mu) \in \big[\pi/\sqrt{8}, \pi\big].$$

Note that the definitions of the Wasserstein distance (17) and the quantization error (18) carry over immediately by replacing $L_2([0,1])$ by $C([0,1])$.

The following result, which can, e.g., be found in Luschgy and Pagès [41, Section 5.4, p.527], handles the (standard) Brownian bridge in the setting of Proposition 2.

PROPOSITION 3. *Let μ be the distribution of the (standard) Brownian bridge in $L_2([0,1])$. Then we have*

$$\operatorname{quant}_n(\mu) \asymp (\log(n))^{-1/2},$$

i.e., we have the same (weak) order of convergence as for the Wiener measure on $L_2([0,1])$.

Note, that the result above can also be derived from Proposition 2 due to the close relation of the Brownian motion and the Brownian bridge, as discussed in Section II.1. Hence, also tight bounds on the quantization coefficient can be derived from those for the Wiener measure.

Finally, we consider the distribution of the solution of a scalar autonomous SDE

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \qquad t \in [0,1] \tag{19}$$

with deterministic initial value $x_0 \in \mathbb{R}$ and a scalar driving Brownian motion, cf. (9) with $r = d = 1$. Here, we use some stronger assumptions on the drift and diffusion coefficients, a and b, than in Section II.2. Namely, we suppose that a and b are differentiable with bounded and Lipschitz continuous derivatives. Obviously, this implies the global Lipschitz property of these coefficients. Consequently, the existence and uniqueness theorem for the solution of an autonomous SDE, namely Theorem 1, applies to the current setting. Furthermore, we assume that $b(x_0) \neq 0$ to exclude the case of a deterministic (ordinary differential) equation. This is essential when considering lower bounds for the quantization error.

The following result is proven in Dereich [14, Theorem 1.1].

PROPOSITION 4. *Let μ be the distribution of the scalar SDE (19). Then the quantization error, see (18), satisfies*

$$\operatorname{quant}_n(\mu) \asymp (\log(n))^{1/2}.$$

The same result for the asymptotic behaviour of the quantization error is obtained in Creutzig et al. [12, Proposition 3] and Luschgy and Pagès [42, Section 3.1, Theorem 1]. Here, (slightly) stronger assumptions on the coefficients of the SDE, see (19), have been used. Note that, as for the Wiener measure, the result carries over to the space $C([0,1])$ equipped with the supremum norm.

III.2. Random Bit Approximation

The main aim in this section is to derive analogous results to Proposition 1, Proposition 2, and Proposition 4, as well as similar results for the distribution of centered Gaussian random elements in a separable Hilbert space, in the *random bit approximation setting*, as we will call it. Recall, that a random bit is a Bernoulli random variable on $\{0,1\}$ with $1/2$ probability of success. Indeed, it will turn out that the random bit approximation problem is a severe restriction of the quantization problem in Section III.1, i.e., at least for some probability distributions, the bit approximation problem is harder than the quantization problem, that means it achieves a worse (weak) asymptotic rate of convergence.

Moreover, we will discuss recent result form Xu and Berger [67] for Borel probability measures over $\mathbb{R}$, and relate them to our results.

In difference to our proceeding in Section III.1, we will immediately give the definitions involved in the random bit approximation problem in a separable Banach space $(V, \|\cdot\|_V)$, instead of generalizing them from $(\mathbb{R}, |\cdot|)$, which was convenient for the quantization problem, due to discussing, e.g., geometrical interpretations of the quantization problem in the special case of the space $\mathbb{R}$.

Let $(V, \|\cdot\|)$ be a separable Banach space and $\mathfrak{M}(V)$ the set of all Borel probability measures on V with a finite second moment. As in Section III.1 we equip $\mathfrak{M}(V)$ with the Wasserstein distance d of order two

$$d(\mu_1, \mu_2) = \inf\Big\{ \Big(\mathrm{E}\, \|X_1 - X_2\|_V^2 \Big)^{1/2} \colon P_{X_1} = \mu_1, P_{X_2} = \mu_2 \Big\}$$

for $\mu_1, \mu_2 \in \mathfrak{M}(V)$, cf. (11) and (17). In contrast to Section III.1, as approximations of a given Borel probability measure $\mu \in \mathfrak{M}(V)$ we will only consider discrete probability measures on V with a support size of at most 2^p and with probability weights being integer multiples of 2^{-p}. For $p \in \mathbb{N}$ let $\nu^{(p)}$ denote the uniform distribution on $\{0,1\}^p$, then the corresponding set of discrete Borel probability measures is denoted by

$$\mathfrak{R}(V, p) = \Big\{ \nu_f^{(p)} \in \mathfrak{M}(V) \colon f \colon \{0,1\}^p \to V \text{ measurable} \Big\},$$

where $\nu_f^{(p)}$ denotes the distribution of f with respect to $\nu^{(p)}$. Since p random bits obviously yield a uniform distribution on $\{0,1\}^p$, clearly, p random bits suffice to sample from any $\nu^{(p)} \in \mathfrak{R}(V, p)$.

Definition 1. Given $\mu \in \mathfrak{M}(V)$ we define the *p random bit approximation error of μ* or simply *p-bit approximation error of μ*, as the distance

$$\mathrm{rbit}(\mu, p) = \inf\big\{ d(\mu, \nu) \colon \nu \in \mathfrak{R}(V, p) \big\}$$

between μ and $\mathfrak{R}(V, p)$. Cf. the definition of the n-th quantization error in (12). In this context, we will also use the notion of a *p (random) bit approximation problem of μ* or simply a *(random) bit approximation of μ*. We use the same terminology for a random variable X with distribution μ.

Our aim is to determine the asymptotic behaviour of $\mathrm{rbit}(\mu, p)$ as p tends to infinity, and to construct probability measures $\mu^{(p)} \in \mathfrak{R}(V, p)$ such that $d(\mu_1, \mu_2)$ is 'close' to $\mathrm{rbit}(\mu, p)$. Specifically, we are interested in separable Hilbert spaces $(H, \|\cdot\|_H)$ as in Proposition 1, Proposition 2, and Proposition 4. That is, μ being the one-dimensional standard normal distribution on $H = \mathbb{R}$, the distribution of a Brownian bridge on $H = L_2([0,1])$, and finally the distribution of the solution of the scalar SDE (19) on $H = L_2([0,1])$. Moreover, we consider centered Gaussian random elements in a separable Hilbert space, having an infinite-dimensional support.

Remark 8. Since, clearly, $\mathfrak{R}(V, p) \subseteq \mathfrak{R}(V, p+1)$ for every $p \in \mathbb{N}$, we also have

$$\mathrm{rbit}(\mu, p+1) \le \mathrm{rbit}(\mu, p)$$

for every $\mu \in \mathfrak{M}(V)$ and every $p \in \mathbb{N}$.

Remark 9. Sometimes we will be interested in a particular subset of $\mathfrak{R}(V, p)$. Namely, the set of all discrete probability measures $\mu^{(p)}$ on V with a support of size (exactly) 2^p and uniform probability weights 2^{-p}. We denote this set by

$$\mathfrak{U}(V, p) = \Big\{ \nu_f^{(p)} \in \mathfrak{R}(V, p) \colon f \colon \{0,1\}^p \to V \text{ is injective} \Big\}$$

for all $p \in \mathbb{N}$. Note that this is no hard restriction for the p-bit approximation problem of $\mu \in \mathfrak{M}(V)$, in the sense, that we have

$$\operatorname{rbit}(\mu, p) = \inf\bigl\{d(\mu, \nu)\colon \nu \in \mathfrak{U}(V, p)\bigr\}$$

for all $p \in \mathbb{N}$. This is an immediate consequence of the fact that there exists a dense embedding of $\mathfrak{U}(V, p)$ in $\mathfrak{R}(V, p)$ with respect to the Wasserstein distance d for all $p \in \mathbb{N}$.

Remark 10. We discuss the relation between random bit approximation and quantization of probability measures for finite and infinite-dimensional spaces V, see Section III.1 for quantization and quantization results. The difference between both notions is that n quantization for $n \in \mathbb{N}$ allows all Borel probability measures $\mathfrak{M}(V)$ with a finite support of size at most n, while p-random bit approximations with $p \in N$ only allow uniform distributions on a support of size 2^p, or probability weights that are integer multiples of 2^p on a support with a size of at most 2^p. Consequently, using the notation

$$\operatorname{quant}(\mu, p) = \operatorname{quant}_{2^p}(\mu),$$

see (12), we have

$$\operatorname{rbit}(\mu, p) \geq \operatorname{quant}(\mu, p)$$

for all $\mu \in \mathfrak{M}(V)$ and for all $p \in \mathbb{N}$. In particular, the quantization problem yields lower bounds on the (weak) rate of convergence for the random bit approximation problem, as p tends to infinity.

Before we start with the analysis and construction of random bit approximations of particular probability distributions $\mu \in \mathfrak{M}(V)$, as already mentioned in the introduction to the current section, we discuss some results from Xu and Berger [67] for the one-dimensional case $V = \mathbb{R}$. Here, the authors have thoroughly studied the case $V = \mathbb{R}$ in a more general setting. Especially, given $p \in \mathbb{N}$ and probability weights $a_1, \dots, a_{2^p}$, their objective is to minimize the Wasserstein distance of order r with $1 \leq r < \infty$, between a Borel probability measure μ on $\mathbb{R}$, having a finite moment of order r, and $\nu = \sum_{k=1}^{2^p} a_k \cdot \delta_{x_k}$ with Dirac measures δ_{x_k} at any point x_k. In Xu and Berger [67] this problem is called finite constrained approximation problem with prescribed weights a_k. A generalization of some of the results from Xu and Berger [67] to the d-dimensional Banach space $\mathbb{R}^d$, for any $d \in N$, equipped with the maximum norm, is given in Chevallier [9].

Remark 11. Let Ψ^{-1} denote the inverse of the distribution function of some $\mu \in \mathfrak{M}(V)$. According to Xu and Berger [67, Remark 5.6(ii)] the unique best approximation $\nu \in \mathfrak{U}(\mathbb{R}, p)$ with respect to the Wasserstein distance d is determined by the points

$$x_k^* = 2^p \cdot \int_{(k-1)\cdot 2^{-p}}^{k\cdot 2^{-p}} \Psi^{-1}(t)\,\mathrm{d}t, \qquad k = 1, \dots, 2^p. \tag{20}$$

We briefly discuss the idea of this result. Actually, it is a consequence of Xu and Berger [67, Theorem 5.5]. Here, it is in particular shown that for uniform probability weights 2^{-p} and Dirac measures $\delta_{x_1}, \dots, \delta_{x_{2^p}}$, the approximation $\nu = \sum_{k=1}^{2^p} 2^{-p} \cdot \delta_{x_k}$ is optimal if and only if x_k is chosen from the interval $[\Psi^{-1}((k-1)\cdot 2^{-p}), \Psi^{-1}(k \cdot 2^{-p})]$, formally excluding the boundary points $\Psi^{-1}(0)$ and $\Psi^{-1}(1)$. This result heavily relies on the (strict) monotonicity of Ψ^{-1} and analogously holds for arbitrary probability weights $a_1, \dots, a_{2^p}$, and in the context of Wasserstein distances of order r with $1 \leq r < \infty$. With this result at hand, one can (for our setting) exploit a special, easily verified, property for the L_2-norm. Namely, the local

expectation x_k^* minimizes $\mathrm{E}\,\|\Psi^{-1}(\cdot) - x_k\|_{L_2}$ on the interval $[(k-1)\cdot 2^{-p}, k\cdot 2^{-p}]$ with respect to x_k.

We now assume that the measure corresponding to Ψ^{-1} is absolutely continuous with respect to the Lebesgue measure on the unit interval. Then in Xu and Berger [67, Theorem 5.15] a constant $c \in]0,\infty[\,\cup\,\{\infty\}$ is given explicitly such that

$$\lim_{p\to\infty} 2^p \cdot \mathrm{rbit}(\mu, p) = c. \tag{21}$$

This in particular gives a lower bound on the convergence rate of $\mathrm{rbit}(\mu,p)$, which is sharp if and only if c is finite. As an elementary example we consider μ being the uniform distribution on the unit interval. Then, we have $c = (2\cdot\sqrt{3})^{-1}$, cf. also Example 1.

Next, we assume that all moments of μ are finite. Then, due to Xu and Berger [67, Theorem 5.20], we have

$$\mathrm{rbit}(\mu, p) \preccurlyeq (2^p)^{-1/2+\varepsilon} \tag{22}$$

for all $\varepsilon > 0$. That is, roughly speaking, an upper bound on the weak rate of convergence of $2^{-p/2}$.

Remark 12. For the proofs of upper bounds for $\mathrm{rbit}(\mu,p)$ we will sometimes use the following simple observation. Let $(\widetilde{V}, \|\cdot\|_{\widetilde{V}})$ denote a separable Banach space different to $(V, \|\cdot\|_V)$ and let $\mu^{(p)} \in \mathfrak{R}(\widetilde{V}, p)$, i.e., a discrete Borel probability measure over $\widetilde{V}$ with support of size at most 2^p. For a measurable mapping $f\colon \widetilde{V} \to V$ we have $\mu_f^{(p)} \in \mathfrak{R}(V,p)$, where $\nu_f^{(p)}$ denotes the distribution of f with respect to $\mu^{(p)}$.

III.2.1. Standard Normal Distribution. Here, we consider $V = \mathbb{R}$. At first we fix some notations. For $p \in \mathbb{N}$ let

$$\begin{aligned} D^{(p)} &= \Big\{ \sum_{i=1}^{p} b_i \cdot 2^{-i} + 2^{-(p+1)} \colon b_i \in \{0,1\} \text{ for } i = 1,\dots,p \Big\} \\ &= \big\{ k\cdot 2^{-p} - 2^{-(p+1)} \colon k = 1,\dots,2^p \big\} \end{aligned}$$

denote the set of dyadic numbers from $[0,1[$ based on p bits, shifted by $2^{-(p+1)}$, so that the set $D^{(p)}$ is symmetric with respect to $1/2$. Furthermore, we define the truncation operator $T^{(p)}$ via

$$T^{(p)}\colon [0,1[\to D^{(p)}, \quad x \mapsto \frac{\lfloor 2^p \cdot x \rfloor}{2^p} + 2^{-(p+1)}, \tag{23}$$

i.e., the application of $T^{(p)}$ means rounding to a nearest element from $D^{(p)}$. From a geometrical point of view, that is, the interval $[(k-1)\cdot 2^{-p}, k\cdot 2^{-p}[$ is projected onto its midpoint $k\cdot 2^{-p} - 2^{-(p+1)}$. Indeed, the operator $T^{(p)}$ is a metric projection of $[0,1[$ equipped with the metric induced by the Euclidean norm, onto the closed subset $D^{(p)}$, see Definition 10 in the Appendix for the definition of a metric projection.

In the sequel, let Y denote a standard normally distributed random variable and let Φ denote the corresponding distribution function with inverse distribution function Φ^{-1}. Observe that $U = \Phi(Y)$ is uniformly distributed on $[0,1]$, see Proposition 6 in the Appendix, so that $T^{(p)}$ is uniformly distributed on $D^{(p)}$. Consequently, the distribution of

$$Y^{(p)} = \Phi^{-1} \circ T^{(p)} \circ \Phi(Y) \tag{24}$$

belongs to $\mathfrak{U}(\mathbb{R}, p)$, i.e., the distribution of $Y^{(p)}$ serves as a p-bit approximation of the standard normal distribution.

THEOREM 4. *Let μ denote the standard normal distribution. Then we have*

$$\operatorname{rbit}(\mu, p) \asymp 2^{-p/2} \cdot p^{-1/2}. \tag{25}$$

Furthermore, let $Y^{(p)}$ be the p-bit approximation of Y as in (24). Then

$$\left(\mathrm{E}\left|Y - Y^{(p)}\right|^2\right)^{1/2} \asymp \operatorname{rbit}(\mu, p), \tag{26}$$

i.e., the sequence $(Y^{(p)})_{p\in\mathbb{N}}$ yields the optimal (weak) rate of convergence for the random bit approximation problem of μ. Moreover,

$$E\big(Y^{(p)}\big) = 0, \tag{27}$$

i.e., $Y^{(p)}$ is a centered random variable for all $p \in \mathbb{N}$, and

$$\sup_{p\in N} \mathrm{E}\big(\big|Y^{(p)}\big|^r\big) < \infty \tag{28}$$

for all $r \geq 1$.

Before we turn to the proof of Theorem 4, we gather some preparing technical results regarding some asymptotic properties of the distribution function Φ and the inverse distribution function Φ^{-1} of the standard normal distribution.

Let φ denote the density function of the standard normal distribution. Concerning the (strong) asymptotics of Φ, we make use of

$$1 - \Phi(x) \approx x^{-1} \cdot \varphi(x) \tag{29}$$

as x tends to infinity, which is well known and follows, e.g., immediately from L'Hôpital's Rule.

LEMMA 2. *For the (strong) asymptotics of the the inverse distribution function Φ^{-1} of the standard normal distribution close to 1 we have*

$$\Phi^{-1}\big(1 - 2^{-p}\big) \approx (\ln(4))^{1/2} \cdot p^{1/2}$$

as p tends to infinity.

PROOF. First we rewrite the claimed property of Φ^{-1}. Let $c = (\ln(4))^{1/2}$ and $x = 1-2^{-p}$, i.e., $p = -\log_2(1-x)$. Then the claim is equivalent to

$$\Phi^{-1}(x) \approx c \cdot \big(-\log_2(1-x)\big)^{1/2}$$

as x tends to 1 (from below). Setting $x = \Phi(y)$ this can equivalently be expressed in terms of Φ by

$$y \approx c \cdot \big(-\log_2\big(1 - \Phi(y)\big)\big)^{1/2}$$

as y tends to infinity. Using (29), the (strict) monotonicity of the logarithmic function, and locally the of $x \mapsto x^2$, and L'Hôpital's Rule, we obtain in fact

$$\begin{aligned} -\log_2(1-\Phi(y)) &\approx -\log_2\big(y \cdot \varphi(y)\big) \\ &\approx -\log_2\big(\exp(-y^2/2)\big) = (2\cdot\ln(2))^{-1} \cdot y^2 = c^{-2} \cdot y^2 \end{aligned}$$

as y tends to infinity. ∎

LEMMA 3. *Let the function* $h\colon]0,\infty[\to \mathbb{R}$ *be defined via*

$$h(x) = \int_0^x \exp(y^2/2)\,\mathrm{d}y.$$

Then we have

$$2^{-p} \cdot p \cdot h\big(\Phi^{-1}(1-2^{-p})\big) \approx \big(\sqrt{2\pi}\cdot\ln(4)\big)^{-1}$$

as p *tends to infinity.*

PROOF. Let $x = \Phi^{-1}(1-2^{-p})$. By L'Hôpital's Rule, or integration by parts, one verifies, similar to (29), that

$$h(x) \approx x^{-1}\cdot\exp(x^2/2) \tag{30}$$

as x tends to infinity. Moreover, Lemma 2 yields

$$p \approx x^2\cdot(\ln(4))^{-1} \tag{31}$$

as p tends to infinity, and by definition of x we have

$$2^{-p} = 1-\Phi(x). \tag{32}$$

Finally, (30) – (32) together with (29) combine to

$$\begin{aligned} 2^{-p}\cdot p\cdot h(x) &\approx \big(1-\Phi(x)\big)\cdot x^2\cdot(\ln(4))^{-1}\cdot x^{-1}\cdot\exp(x^2/2)\\ &\overset{(29)}{\approx} x^{-1}\cdot\varphi(x)\cdot\exp(x^2/2)\cdot x\cdot(\ln(4))^{-1}\\ &= \big(\sqrt{2\pi}\cdot\ln(4)\big)^{-1}\end{aligned}$$

as p tends to infinity. ∎

LEMMA 4. *Let the function* $g\colon]0,\infty[\to \mathbb{R}$ *be defined via*

$$g(x) = \int_{[x,\infty[} (y-x)^2\cdot\varphi(y)\,\mathrm{d}y.$$

Then we have

$$g\big(\Phi^{-1}\big(1-2^{-(p+1)}\big)\big) \approx 2^{-p}\cdot p^{-1}\cdot(\ln(4))^{-1}$$

as p *tends to infinity.*

PROOF. Set $x = \Phi^{-1}(1-2^{-(p+1)})$. We are going to show

$$g(x) \approx 2\cdot\big(1-\Phi(x)\big)\cdot x^{-2} \tag{33}$$

as x tends to infinity. Then plugging in the definition of x in $\Phi(x)$ we obtain

$$g(x) \approx 2^{-p}\cdot x^{-2}$$

as x (and therefore also p) tends to infinity. It remains to apply Lemma 2 to x^{-2} and we immediately get the claim

$$g(x) \approx 2^{-p}\cdot p^{-1}\cdot(\ln(4))^{-1}$$

as x (and therefore also p) tends to infinity, finishing the proof. So, let us turn to (33). By (29) the relation (33) is equivalent to

$$g(x) \approx 2\cdot\varphi(x)\cdot x^{-3} \tag{34}$$

as x tends to infinity. To show this, we will iteratively use L'Hôpital's Rule. Therefore, we compute the first three derivatives of g with respect to x

$$\begin{aligned} g'(x) &= -2 \cdot \int_{[x,\infty[} (y-x) \cdot \varphi(y)\,\mathrm{d}y, \\ g''(x) &= 2 \cdot \int_{[x,\infty[} \varphi(y)\,\mathrm{d}y, \\ g'''(x) &= -2 \cdot \varphi(x), \end{aligned}$$

as well as, for the right hand side of (34)

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}x}\Big(\varphi(x) \cdot x^{-3}\Big) &= -\varphi(x) \cdot x^{-2} - 3 \cdot \varphi(x) \cdot x^{-4}, \\ \frac{\mathrm{d}}{\mathrm{d}x}\Big(-2 \cdot \varphi(x) \cdot x^{-2}\Big) &= 2 \cdot \varphi(x) \cdot x^{-1} + 4 \cdot \varphi(x) \cdot x^{-3}, \\ \frac{\mathrm{d}}{\mathrm{d}x}\Big(2 \cdot \varphi(x) \cdot x^{-1}\Big) &= -2 \cdot \varphi(x) - 2 \cdot \varphi(x) \cdot x^{-2}. \end{aligned}$$

Using three times L'Hôpital's Rule yields

$$\begin{aligned} \lim_{x\to\infty} \frac{g(x)}{2 \cdot \varphi(x) \cdot x^{-3}} &= \lim_{x\to\infty} \frac{g'(x)}{-2 \cdot \varphi(x) \cdot x^{-2}} \\ &= \lim_{x\to\infty} \frac{g''(x)}{2 \cdot \varphi(x) \cdot x^{-1}} \\ &= \lim_{x\to\infty} \frac{g'''(x)}{-2 \cdot \varphi(x)} = 1, \end{aligned}$$

i.e., (33) is verified. ■

LEMMA 5. *For $0 < x < 1$ we define the real valued functions*

$$u(x) = \varphi\big(\Phi^{-1}(1-x)\big)$$

and

$$v(x) = x \cdot \big(\ln(x^{-1})\big)^{1/2}.$$

Then we have

$$u(x) \approx \sqrt{2} \cdot v(x)$$

as x tends to zero (from above).

PROOF. At first, we show an auxiliary result for $\widetilde{u}(x) = u(x)^2$ and $\widetilde{v}(x) = v(x)^2$ with derivatives

$$\begin{aligned} \widetilde{u}'(x) &= 2 \cdot \varphi\big(\Phi^{-1}(1-x)\big) \cdot \Phi^{-1}(1-x), \\ \widetilde{u}''(x) &= 2 \cdot \big(\Phi^{-1}(1-x)\big)^2 - 2, \end{aligned}$$

as well as,

$$\begin{aligned} \widetilde{v}'(x) &= -2 \cdot x \cdot \ln(x) - x, \\ \widetilde{v}''(x) &= -2 \cdot \ln(x) - 3. \end{aligned}$$

Consequently, two consecutive applications of L'Hôpital's Rule yield

$$\lim_{x\to 0}\frac{\widetilde{v}(x)}{\widetilde{u}(x)} = \lim_{x\to 0}\frac{-2\cdot\ln(x)-3}{2\cdot\big(\Phi^{-1}(1-x)\big)^2-2} = \lim_{x\to 0}\frac{\ln\big(x^{-1}\big)}{\big(\Phi^{-1}(1-x)\big)^2}. \tag{35}$$

Due to the local monotonicity of $x\mapsto x^2$ we can apply Lemma 2 with $x=2^{-p}$ to the last quotient, in order to obtain

$$\big(\Phi^{-1}(1-x)\big)^2 \approx \ln(4)\cdot\big(-\log_2(x)\big) = 2\cdot\ln\big(x^{-1}\big)$$

as x tends to zero (from above). Together with (35) that is

$$\widetilde{v}(x)\approx\widetilde{u}(x)/2$$

as x tends to zero (from above). The local monotonicity of $x\mapsto x^{1/2}$ yields the claim. ■

Lemma 6. *We have*

$$\Phi^{-1}(x_1)-\Phi^{-1}(x_0)\succcurlyeq(x_1-x_0)\cdot(1-x_0)^{-1}\cdot\big(-\ln(1-x_0)\big)^{-1/2}$$

uniformly in $1/2<x_0<x_1<1$.

Proof. At first, we note that the derivative of Φ^{-1} at some point $x\in]0,1[$ is given by $\big(\varphi\big(\Phi^{-1}(x)\big)\big)^{-1}$ with φ the standard normal density function. Then the mean value theorem yields the existence of $\xi\in]x_0,x_1[$ such that

$$\Phi^{-1}(x_1)-\Phi^{-1}(x_0)=(x_1-x_0)\cdot\big(\Phi^{-1}\big)'(\xi)\geq(x_1-x_0)\cdot\big(\varphi\big(\Phi^{-1}(x_0)\big)\big)^{-1},$$

where the inequality is due to $\big(\varphi\big(\Phi^{-1}(x)\big)\big)^{-1}$ monotonically decreasing for decreasing x in $]1/2,1[$. The statement is now an immediate consequence of Lemma 5 with $1-x=x_0$ and x_0 tending to 1 (from below). ■

Proof. *(of Theorem 4)* By Definition 1 of the p-bit approximation error of μ together with $Y^{(p)}\in\mathfrak{U}(\mathbb{R},p)$, we have

$$\operatorname{rbit}(\mu,p)\leq\Big(\mathrm{E}\big|Y-Y^{(p)}\big|^2\Big)^{1/2}. \tag{36}$$

Hence we show that the sequence $(Y^{(p)})_{p\in\mathbb{N}}$ satisfies

$$\Big(\mathrm{E}\big|Y-Y^{(p)}\big|^2\Big)^{1/2}\preccurlyeq 2^{-p/2}\cdot p^{-1/2}. \tag{37}$$

To this end, let U be uniformly distributed on the unit interval. Due to Proposition 6 in the Appendix we have

$$\mathrm{E}\big|Y-Y^{(p)}\big|^2=\mathrm{E}\big|\Phi^{-1}(U)-\Phi^{-1}\circ T^{(p)}(U)\big|^2. \tag{38}$$

Furthermore, let $z_k=k\cdot 2^{-p}$ for $k=2^{p-1},\dots,2^p$, and let φ denote the density of the standard normal distribution. Using (38) and the point symmetry of Φ^{-1} respective $\Phi^{-1}\circ T^{(p)}$ at the point $1/2$, we obtain

$$\mathrm{E}\big|Y-Y^{(p)}\big|^2=2\cdot\sum_{k=2^{p-1}+1}^{2^p}A_k,$$

where

$$A_k=\int_{[z_{k-1},z_k[}\big|\Phi^{-1}(u)-x_k\big|^2\,\mathrm{d}u$$

with $x_k=\Phi^{-1}\big(z_k-2^{-(p+1)}\big)\in D^{(p)}$.

We consider different cases, starting with $2^{p-1}+1 \leq k \leq 2^p-2$. Observe that the derivative of Φ^{-1} is given by $\varphi \circ \Phi^{-1}$. For the integrand in A_k, the mean value theorem yields for every $u \in [z_{k-1}, z_k[$ existence of $\xi \in [z_{k-1}, z_k[$ such that

$$\left|\Phi^{-1}(u) - x_k\right| = \left|\Phi^{-1}(u) - \Phi^{-1}(z_k - 2^{-(p+1)})\right| \leq \left|u - (z_k - 2^{-(p+1)})\right| \cdot \left(\varphi\big(\Phi^{-1}(\xi)\big)\right)^{-1}.$$

Since $\varphi \circ \Phi^{-1}$ is monotonically decreasing for increasing argument in $[1/2, 1[$, we obtain

$$\left|\Phi^{-1}(u) - x_k\right| \leq \frac{\left|u - (z_k - 2^{-(p+1)})\right|}{\varphi\big(\Phi^{-1}(z_k)\big)}.$$

Plugging in this estimate in the definition of A_k, a simple computation shows

$$A_k \leq \frac{2^{-3p}}{12 \cdot \left(\varphi\big(\Phi^{-1}(z_k)\big)\right)^2}.$$

Consequently, using integration by substitution, we get

$$\begin{aligned} 2^p \cdot p \cdot \sum_{k=2^{p-1}+1}^{2^p-2} A_k &\leq \frac{2^{-p} \cdot p}{12} \cdot 2^{-p} \sum_{k=2^{p-1}+1}^{2^p-2} \left(\varphi\big(\Phi^{-1}(z_k)\big)\right)^{-2} \\ &\leq \frac{2^{-p} \cdot p}{12} \cdot \int_{1/2}^{1-2^{-p}} \left(\varphi\big(\Phi^{-1}(u)\big)\right)^{-2} \mathrm{d}u \\ &= \frac{2^{-p} \cdot p}{12} \cdot \int_{\Phi^{-1}(1/2)}^{\Phi^{-1}(1-2^{-p})} \left(\varphi\big(\Phi^{-1}(x)\big)\right)^{-1} \mathrm{d}x \\ &= \frac{2^{-p} \cdot p \cdot \sqrt{2\pi}}{12} \cdot h\big(\Phi^{-1}\big(1-2^{-p}\big)\big), \end{aligned}$$

where

$$h(x) = \int_0^x \exp(y^2/2)\,\mathrm{d}y$$

for $x > 0$. By Lemma 3, we have

$$h\big(\Phi^{-1}\big(1-2^{-p}\big)\big) \approx 2^p \cdot p^{-1} \cdot \left(\sqrt{2\pi} \cdot \ln(4)\right)^{-1}$$

as p tends to infinity. Together with the above estimate we obtain

$$\limsup_{p\to\infty} 2^p \cdot p \cdot \sum_{k=2^{p-1}+1}^{2^p-2} A_k \leq \frac{1}{12 \cdot \ln(4)}.$$

It remains to estimate A_k for $k = 2^p - 1$ and $k = 2^p$. Due to the convexity of Φ^{-1} on $[1/2, 1[$ we have (using integration by substitution) for every $2^{p-1}+1 \leq k \leq 2^p$

$$A_k \leq 2 \cdot \int_{[z_{k-1}+2^{-(p+1)}, z_k[} \left|\Phi^{-1}(u) - x_k\right|^2 \mathrm{d}u = 2 \cdot \int_{[x_k, \Phi^{-1}(z_k)[} (x - x_k)^2 \cdot \varphi(x)\,\mathrm{d}x. \tag{39}$$

Furthermore, the convexity of Φ^{-1} on $[1/2, 1[$ implies that this upper bound is monotonically increasing in k. Therefore, the case $k = 2^p$ also covers $k = 2^p - 1$. Put

$$g(x) = \int_{[x,\infty[} (x-y)^2 \cdot \varphi(y)\,\mathrm{d}y \tag{40}$$

for $x > 0$. By Lemma 4 we have

$$g\big(\Phi^{-1}\big(1-2^{-(p+1)}\big)\big) \approx 2^{-p} \cdot p^{-1} \cdot (\ln(4))^{-1}$$

as p tends to infinity. Together with (39) we obtain

$$A_{2^p} \preccurlyeq 2^{-p} \cdot p^{-1} \cdot 2 \cdot (\ln(4))^{-1}.$$

Altogether we have

$$\limsup_{p\to\infty} 2^p \cdot p \cdot \mathrm{E}\big|Y - Y^{(p)}\big|^2 \le \frac{1}{12 \cdot \ln(4)} + 2 \cdot \frac{2}{\ln(4)} = \frac{49}{12 \cdot \ln(4)}.$$

This completes the proof of the upper bound (37).

Next, we show the lower bound

$$2^{-(p/2)} \cdot p^{-1/2} \preccurlyeq \mathrm{rbit}(\mu, p). \tag{41}$$

To this end, we consider an (arbitrary) discrete random variable $\widehat{Y}^{(p)}$ with distribution in $\mathfrak{R}(\mathbb{R}, p)$, which is defined on the same probability space as Y. By $\widehat{x}$ we denote the essential supremum of $\widehat{Y}^{(p)}$. We are going to consider two cases, at first we assume

$$\widehat{x} \le \Phi^{-1}\big(1 - 2^{-(p+1)}\big).$$

We observe that for u from $[1 - 2^{-(p+1)}, 1[$ the monotonicity of Φ^{-1} yields

$$\big|\Phi^{-1}(u) - \widehat{Y}^{(p)}\big| = \big|\Phi^{-1}(u) - \widehat{x}\big| \ge \big|\Phi^{-1}(u) - \Phi^{-1}\big(1 - 2^{-(p+1)}\big)\big|.$$

Consequently, for U being uniformly distributed on the unit interval, we have

$$\begin{aligned}
\mathrm{E}\big(Y - \widehat{Y}^{(p)}\big)^2 &\ge \mathrm{E}\Big(\big(\Phi^{-1}(U) - \widehat{Y}^{(p)}\big)^2 \cdot 1_{\{U \ge 1 - 2^{-(p+1)}\}}\Big) \\
&\ge \mathrm{E}\Big(\big(\Phi^{-1}(U) - \Phi^{-1}\big(1 - 2^{-(p+1)}\big)\big)^2 \cdot 1_{\{U \ge 1 - 2^{-(p+1)}\}}\Big) \\
&= g\big(\Phi^{-1}\big(1 - 2^{-(p+1)}\big)\big),
\end{aligned}$$

where g as in (40). Lemma 4 implies

$$g\big(\Phi^{-1}\big(1 - 2^{-(p+1)}\big)\big) \succcurlyeq 2^{-p} \cdot p^{-1},$$

i.e., the claim (41) for the current case. For the second case, i.e.,

$$\widehat{x} > \Phi^{-1}\big(1 - 2^{-(p+1)}\big)$$

we use Lemma 6 with $x = 1 - 3 \cdot 2^{-(p+2)}$ and $y = 1 - 2^{-(p+1)}$ to conclude that

$$\Phi^{-1}(y) - \Phi^{-1}(x) \succcurlyeq p^{-1/2}.$$

Together with the monotonicity of Φ^{-1} this leads to

$$\begin{aligned}
\mathrm{E}\big(Y - \widehat{Y}^{(p)}\big)^2 &\ge \mathrm{E}\Big(\big(\Phi^{-1}(U) - \widehat{x}\big)^2 \cdot 1_{\{\widehat{Y}^{(p)} = \widehat{x}\}} \cdot 1_{\{U \le x\}}\Big) \\
&\ge \mathrm{E}\Big(\big(\Phi^{-1}(x) - \Phi^{-1}(y)\big)^2 \cdot 1_{\{\widehat{Y}^{(p)} = \widehat{x}\}} \cdot 1_{\{U \le x\}}\Big) \\
&\succcurlyeq p^{-1} \cdot P\Big(\big\{\widehat{Y}^{(p)} = \widehat{x}\big\} \cap \{U \le x\}\Big) \\
&\ge p^{-1} \cdot \Big(P\big(\big\{\widehat{Y}^{(p)} = \widehat{x}\big\}\big) - P(\{U > x\})\Big) \\
&\ge p^{-1} \cdot \big(2^{-p} - P(\{U > x\})\big) \\
&= p^{-1} \cdot 2^{-(p+2)},
\end{aligned}$$

where U is uniformly distributed on the unit interval. This completes the proof of the lower bound (41). Combining (36), (37) and (41) we obtain

$$2^{-p/2} \cdot p^{-1/2} \preccurlyeq \operatorname{rbit}(\mu, p) \leq \left(\mathrm{E}\left|Y - Y^{(p)}\right|^2\right)^{1/2} \preccurlyeq 2^{-p/2} \cdot p^{-1/2},$$

i.e., the claimed bit approximation rate (25), as well as, the claimed particular bit approximation rate (26) for $(Y^{(p)})_{p\in\mathbb{N}}$, see (24).

For the proof of (27), i.e., $Y^{(p)}$ is a centered random variable for all $p \in \mathbb{N}$, we observe that $T^{(p)}(U)$ is uniformly distributed on $D^{(p)}$. Since Φ^{-1} is point symmetric with respect to $1/2$, i.e., $\Phi^{-1}(1-x) = \Phi^{-1}(x)$ for all $x \in]0,1[$, we get (27) for symmetry reasons.

Finally, we show (28). Observe that Φ^{-1} is a convex function on $[1/2, 1[$, and for $r \geq 1$, $x \mapsto x^r$ is convex and monotonically increasing on $[0, \infty[$. Therefore, Lemma 18 from the Appendix yields convexity of $x \mapsto \left(\Phi^{-1}(x)\right)^r$ on $[1/2, 1[$. Consequently, for $k = 2^{p-1} + 1, \ldots, 2^p$, we have

$$2^{-p} \cdot \left|\Phi^{-1}\left(k \cdot 2^p - 2^{-(p+1)}\right)\right|^r \leq \int_{(k-1)\cdot 2^p}^{k\cdot 2^p} \left|\Phi^{-1}(u)\right|^r \mathrm{d}u,$$

which implies due to symmetry of Y and $Y^{(p)}$

$$\mathrm{E}\left|Y^{(p)}\right|^r \leq \mathrm{E}\left|Y\right|^r$$

for all $p \in \mathbb{N}$. In that sense, $\mathrm{E}\left|Y^{(p)}\right|^r$ can be regarded as a midpoint rule for the approximation of $\mathrm{E}\,|Y|^r$. For (28) it remains to observe that all absolute moments of the standard normal distribution are bounded, see Winkelbauer [66]. ■

Remark 13. We compare our results from Theorem 4 with those from Xu and Berger [67], as discussed in Remark 11, for the standard normal distribution μ.

We first note that the measure corresponding to Φ^{-1} is absolutely continuous with respect to the Lebesgue measure on the unit interval. Due to (25), that is, $\operatorname{rbit}(\mu, p) \asymp 2^{-p/2} \cdot p^{-1/2}$, we have $c = \infty$ in (21). Moreover, note that μ has finite moments of any order, see, e.g., Winkelbauer [66]. As a consequence (22) in particular applies to μ, i.e., the order of convergence of $\operatorname{rbit}(\mu, p)$ is only slightly better than the upper bound from (22), which holds indeed for every $\mu \in \mathfrak{M}(\mathbb{R})$ that has finite moments of any order.

Next, we discuss the selection of the support points x_k with $k = 1, \ldots, 2^p$ for $Y^{(p)}$, as in (24). Recall that we use the values of Φ^{-1} applied to the midpoints of a uniform partition of $[0,1]$, namely $x_k(p) = \Phi^{-1}(k \cdot 2^{-p} - 2^{-(p+1)})$. Indeed this selection is not optimal, since (20) with $\Psi^{-1} = \Phi^{-1}$ yields as the (unique) optimal selection $x_k^*(p)$ the local average of Φ^{-1} on the subinterval $[(k-1)\cdot 2^p, k \cdot 2^p[$. Nevertheless, the lower bound in (25) together with (26) implies that our bit approximation $Y^{(p)}$ is optimal in the sense that no other construction, in particular the one based on x_k^*, which we will denote by $Y_*^{(p)}$, can yield a better order of convergence for the bit approximation error than $Y^{(p)}$.

Finally, let us discuss whether the optimal bit approximation $Y_*^{(p)}$ would help in Theorem 4 in proofing the lower bound (41). First of all, instead of an arbitrary discrete random variable $\widehat{Y}^{(p)}$ with distribution in $\mathfrak{R}(\mathbb{R}, p)$ we consider the particular choice $Y_*^{(p)}$. The convexity of Φ^{-1} on $[1/2, 1[$ implies that the essential supremum x^* of $Y_*^{(p)}$ is given by $x_{2^p}^*$ and satisfies

$$x^* > \Phi^{-1}\left(1 - 2^{-(p+1)}\right),$$

i.e., we no longer need a case distinction. In the proof of the remaining case, the more involved one, the only difference is, that the set $\{\widehat{Y}^{(p)} = \widehat{x}\} \cap \{U \leq 1 - 3 \cdot 2^{-(p+2)}\}$ is now

explicitly known as the interval $[1-2^{-p}, 1-3\cdot 2^{-(p+2)}[$. We conclude, the optimal selection of support points x_k^* is of minor help for our proof of the lower bound (41).

REMARK 14. We compare Theorem 4 with the corresponding result for the quantization problem of the standard normal distribution μ, as stated in Proposition 1. We recall,

$$\operatorname{quant}(\mu, p) \asymp 2^{-p}.$$

I.e., the quantization error converges to zero much faster than the corresponding random bit approximation error $\operatorname{rbit}(\mu, p)$. We add that this is in general not the case, since, e.g., for the uniform distribution on $[0,1]$ both quantities are of the same order 2^{-p}, see Example 1 for the quantization result and Remark 11 in particular the example for (21) for the random bit approximation result.

III.2.2. Brownian Bridge. Let B denote a standard Brownian bridge on $[0,1]$, which may be considered as a centered Gaussian element that takes values in $H = L_2 = L_2([0,1])$. For convenience of the reader we briefly recall the main facts of the Lévy-Ciesielski construction of a Brownian motion as presented in Remark 3.

Let $(Y_{i,j})_{i,j}$ with $i \in \mathbb{N}$ and $j = 1, \dots 2^{i-1}$ be an independent family of standard normally distributed random variables and $(s_{i,j})_{i,j}$ the family of Schauder functions given by (6). For a one-dimensional standard Brownian bridge B we have

$$B(t) = \sum_{i=1}^{\infty} \sum_{j=1}^{2^{i-1}} s_{i,j}(t) \cdot Y_{i,j} \tag{42}$$

with convergence with respect to the L_2-norm, see (8).

There are two natural dimensions of discretization for B. One is to cut off the infinite sum in (42) after finitely many summands and the second dimension is the approximation of the involved (independent) standard normal distributions $Y_{i,j}$ by means of random bits.

Concerning the first dimension of discretization, we define

$$B^{(\ell)}(t) = \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} s_{i,j}(t) \cdot Y_{i,j}$$

for $\ell \in \mathbb{N}$, i.e., $B^{(\ell)}$ is the piecewise linear interpolation of B at the points $k \cdot 2^{-\ell}$ with $k = 0, \dots, 2^\ell$. Observe that $B^{(\ell)}$ has $2^\ell - 1$ summands, i.e., employs (the first) $2^\ell - 1$ Schauder functions.

REMARK 15. This approximation is natural in the following sense. In the transition from $\ell - 1$ to ℓ exactly those Schauder functions $s_{\ell,1}, \dots, s_{\ell,2^{\ell-1}}$ with a support size $2^{-\ell+1}$ are taken into account. This corresponds to a 'consistent' local refinement of the approximation $B^{(\ell-1)}$ on all subintervals $[(k-1)\cdot 2^{-\ell+1}, k \cdot 2^{-\ell+1}[$ for $k = 1, \dots, 2^{\ell-1}$. Since the local behaviour of a Brownian bridge is independent from the considered time interval, cf. Markov property of Brownian motion, see, e.g., Kallenberg [38, Theorem 11.11], there is (at least in our setting) no reason for a different local refinement of the approximation on particular subintervals. Hence we either consider the whole set $\{s_{\ell,k} \colon k = 1, \dots, 2^{\ell-1}\}$ of Schauder functions or none of them.

The following result for the (weak) order of convergence of the L_2-error of $B^{(\ell)}$ is well known, see, e.g., Ritter [59, Sect II.3] for references and remarks.

LEMMA 7. *We have*

$$\Big(\mathrm{E}\big|\|B - B^{(\ell)}\|_{L_2}^2\Big)^{1/2} \asymp 2^{-\ell/2}.$$

For the second dimension of discretization, which we use on top of the first on, we consider for fixed $\ell \in \mathbb{N}$ a vector

$$\boldsymbol{p} = (p_{i,j}) \in \mathbb{N}^{2^\ell - 1}$$

with $i = 1, \dots, \ell$ and $j = 1, \dots, 2^{i-1}$, of bit numbers. We define

$$B^{(\ell,\boldsymbol{p})}(t) = \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} s_{i,j}(t) \cdot Y_{i,j}^{(p_{i,j})}, \tag{43}$$

where $Y_{i,j}^{(p_{i,j})}$ is the bit approximation of $Y_{i,j}$ according to (24). Note that the distribution of $B^{(\ell,\boldsymbol{p})}$ belongs to $\mathfrak{U}\big(L_2, |\boldsymbol{p}|\big)$ with

$$|\boldsymbol{p}| = \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} p_{i,j}.$$

This approach, which is appropriate for the construction of multilevel Monte Carlo algorithms, as we will see in Section IV.5, has been suggested in Giles [25, p. 320]. At first, we show a preparing lemma.

LEMMA 8. *The family of Schauder functions* $(s_{i,j})_{i,j}$ *with* $i = 0$ *and* $j = 1$ *or* $i \in \mathbb{N}$ *and* $j = 1, \dots, 2^{i-1}$, *as defined in* (6) *satisfies*

$$\|s_{i,j}\|_{L_2([0,1])}^2 = 3^{-1} \cdot 2^{-2\cdot i}.$$

PROOF. For $i \geq 1$ we have, using integration by substitution,

$$\begin{aligned}\|s_{i,j}\|_{L_2([0,1])}^2 &= \int_{I_{i,j} \cup J_{i,j}} (s_{i,j}(t))^2 \,\mathrm{d}t = 2 \cdot \int_{I_{i,j}} (s_{i,j}(t))^2 \,\mathrm{d}t \\ &= 2 \cdot \int_0^{2^{-i}} \big(2^{(i-1)/2} \cdot t\big)^2 \,\mathrm{d}t = 2^i \cdot \int_0^{2^{-i}} t^2 \,\mathrm{d}t = 3^{-1} \cdot 2^{-2\cdot i}.\end{aligned}$$

■

Now, we can analyze the $|\boldsymbol{p}|$-bit approximation $B^{(\ell,\boldsymbol{p})}$ of $B^{(\ell)}$ for large $|\boldsymbol{p}|$.

LEMMA 9. *We have*

$$\Big(\mathrm{E}\big\|B^{(\ell)} - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2\Big)^{1/2} \asymp \left(\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-p_{i,j}} \cdot p_{i,j}^{-1} \cdot 2^{-2\cdot i}\right)^{1/2}$$

uniformly in $\ell \in \mathbb{N}$ *and* $\boldsymbol{p} \in \mathbb{N}^{2^\ell - 1}$.

PROOF. Using Fubini's theorem, Bienaymé's formula, results from Theorem 4 for the particular bit approximations $Y_{i,j}^{(p_{i,j})}$ of $Y_{i,j}$, and $\|s_{i,j}\|_{L_2}^2 \asymp 2^{-2\cdot i}$, which is due to Lemma 8, we obtain

$$\begin{aligned}\mathrm{E}\big\|B^{(\ell)} - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2 &= \int_0^1 \mathrm{E}\left(\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} \Big(Y_{i,j}^{(p_{i,j})} - Y_{i,j}\Big) \cdot s_{i,j}(t)\right)^2 \mathrm{d}t \\ &= \int_0^1 \mathrm{Var}\left(\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} \Big(Y_{i,j}^{(p_{i,j})} - Y_{i,j}\Big) \cdot s_{i,j}(t)\right) \mathrm{d}t\end{aligned}$$

$$= \int_0^1 \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} \mathrm{Var}\Big(\Big(Y_{i,j}^{(p_{i,j})} - Y_{i,j}\Big) \cdot s_{i,j}(t)\Big)\,\mathrm{d}t$$

$$= \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} \mathrm{E}\Big(Y_{i,j}^{(p_{i,j})} - Y_{i,j}\Big)^2 \cdot \int_0^1 (s_{i,j}(t))^2\,\mathrm{d}t$$

$$\asymp \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-p_{i,j}} \cdot p_{i,j}^{-1} \cdot 2^{-2\cdot i}. \qquad \blacksquare$$

With this auxiliary result we can establish a constructive upper bound on the random bit approximation error $\mathrm{rbit}(\mu, p)$ for the distribution μ of the Brownian bridge B on L_2 by means of the construction $B^{(\ell,\boldsymbol{p})}$.

THEOREM 5. *Let μ be the distribution of the standard Brownian bridge B on L_2. Define $\boldsymbol{p}(\ell) \in \mathbb{N}^{2^\ell - 1}$ for $\ell \in \mathbb{N}$ by*

$$p_{i,j}(\ell) = 2 \cdot (\ell - i + 1) \tag{44}$$

for $i = 1, \ldots, \ell$ and $j = 1, \ldots, 2^{i-1}$. Then we have

$$\mathrm{rbit}(\mu, |\boldsymbol{p}(\ell)|) \le \Big(\mathrm{E}\big\|B - B^{(\ell,\boldsymbol{p}(\ell))}\big\|_{L_2}^2\Big)^{1/2} \preccurlyeq |\boldsymbol{p}(\ell)|^{-1/2} \tag{45}$$

and

$$|\boldsymbol{p}(\ell)| = 2^{\ell+2} - 2 \cdot \ell - 4 \asymp 2^\ell.$$

PROOF. We write $\boldsymbol{p}$ and $p_{i,j}$ instead of $\boldsymbol{p}(\ell)$ and $p_{i,j}(\ell)$, respectively, to simplify the notation. Due to $B^{(\ell,\boldsymbol{p})} \in \mathfrak{U}(L_2, |\boldsymbol{p}|)$, by definition

$$\mathrm{rbit}(\mu, |\boldsymbol{p}|) \le \Big(\mathrm{E}\big\|B - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2\Big)^{1/2}.$$

Hence we show that

$$\Big(\mathrm{E}\big\|B - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2\Big)^{1/2} \preccurlyeq |\boldsymbol{p}|^{-1/2}.$$

Since

$$\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-p_{i,j}} \cdot p_{i,j}^{-1} \cdot 2^{-2\cdot i} \le \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-p_{i,j}} \cdot 2^{-2\cdot i} = 2^{-2} \cdot (2^\ell - 1) \cdot 2^{-2\cdot\ell} \le 2^{-\ell} \tag{46}$$

for the specific choice of the bit numbers $p_{i,j}$, Lemmata 7 and 9 immediately yield

$$\Big(\mathrm{E}\big\|B - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2\Big)^{1/2} \preccurlyeq \Big(\mathrm{E}\big\|B - B^{\ell}\big\|_{L_2}^2\Big)^{1/2} + \Big(\mathrm{E}\big\|B^{(\ell)} - B^{(\ell,\boldsymbol{p})}\big\|_{L_2}^2\Big)^{1/2} \preccurlyeq 2^{-\ell/2}.$$

The explicit formula for $|\boldsymbol{p}|$ is easily verified by induction. Obviously, the claim holds true for $\ell = 1$. We show the induction step from ℓ to $\ell + 1$

$$|\boldsymbol{p}(\ell+1)| = \sum_{i=1}^{\ell+1} \sum_{j=1}^{2^{i-1}} 2 \cdot (\ell + 1 - i + 1)$$

$$= 2^\ell \cdot 2 + \sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} \big(p_{i,j}(\ell) + 2\big) = 2^{\ell+3} - 2 \cdot (\ell + 1) - 4.$$

The explicit formula for $|\boldsymbol{p}|$ implies clearly the weak asymptotic estimate $|\boldsymbol{p}| \asymp 2^\ell$. $\blacksquare$

COROLLARY 1. *Let μ be the distribution of the (standard) Brownian bridge on L_2. Then we have*

$$\operatorname{rbit}(\mu, p) \asymp \operatorname{quant}(\mu, p) \asymp p^{-1/2}.$$

In particular this (weak) order of convergence is achieved by $B^{(\ell,\boldsymbol{p}(\ell))}$, as defined in Theorem 5, i.e.,

$$\left(\mathrm{E}\left\|B - B^{(\ell,\boldsymbol{p}(\ell))}\right\|_{L_2}^2\right)^{1/2} \asymp \operatorname{rbit}(\mu, |\boldsymbol{p}(\ell)|).$$

PROOF. By Proposition 3 we have the lower bound

$$p^{-1/2} \preccurlyeq \operatorname{quant}(\mu, p).$$

Moreover, by definition, cf. also Remark 10, we have

$$\operatorname{quant}(\mu, p) \le \operatorname{rbit}(\mu, p).$$

Finally, Theorem 5, in particular (45), yields the upper bound

$$\operatorname{rbit}(\mu, p) \preccurlyeq p^{-1/2}.$$

The statement for $B^{(\ell,\boldsymbol{p}(\ell))}$ is now an immediate consequence of (45). ∎

REMARK 16. We gather some interesting facts on $B^{(\ell,\boldsymbol{p}(\ell))}$.

(i) Since the Schauder functions $s_{i,j}$ have support of size 2^{-i+1}, the bit approximation $B^{(\ell,\boldsymbol{p}(\ell))}$ involves all Schauder functions with a support size between 1 and $2^{-\ell+1}$. Moreover, the number of random bits that is associated to $s_{i,j}$, see (44), only depends on the size of the support, see Remark 15 for the discussion why this is natural. Furthermore, this number varies linearly between $2 \cdot l$ and 2 in ascending index i.

(ii) Observe that in our construction of $B^{(\ell,\boldsymbol{p}(\ell))}$ the total number of bits $|\boldsymbol{p}(\ell)|$ coincides, up to a multiplicative constant, with the number of terms in $B^{(\ell,\boldsymbol{p}(\ell))}$ and $B^{(\ell)}$. Since each term needs to be approximated with at least one bit, $|\boldsymbol{p}(\ell)|$ is optimal up to a multiplicative constant. That means, in particular, that the bit numbers given by (44) approximately minimize $|\boldsymbol{p}|$ subject to the constraint (46).

(iii) Though the partial sum $B^{(\ell)}$ formally corresponds to $p_{i,j} = \infty$ for $i = 1, \dots, \ell$ and $j = 1, \dots, 2^{i-1}$, we still have the same order $2^{-\ell/2}$ for the convergence of the approximation error of $B^{(\ell)}$ and $B^{(\ell,\boldsymbol{p}(\ell))}$, see Lemma 7 and Corollary 1.

REMARK 17. As mentioned in Remark 16 (ii), the bit numbers given by (44) depend on the index i and they approximately minimize $|\boldsymbol{p}|$, subject to the constraint (46). This raises the question whether the dependency on i is really necessary. To answer this question, we consider constant bit numbers

$$p = p_{i,j} \qquad \text{for } i = 1, \dots, \ell \text{ and } j = 1, \dots, 2^{i-1}. \tag{47}$$

Then, for $p \in \mathbb{N}$, we have

$$\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-p} \cdot p^{-1} \cdot 2^{-2\cdot i} = 2^{-p} \cdot p^{-1} \cdot (1 - 2^{-\ell})/2 \le 2^{-p} \cdot p^{-1} \le 2^{-p/2}, \tag{48}$$

where the last inequality is due to $p \le 2^{p/2}$ for all $p \in \mathbb{N}$. Hence for (46) to hold true we need to have $2^{-p/2} \le 2^{-\ell}$, i.e., $p \ge \ell/2$. On the other hand, $p = \ell$ immediately yields (46), cf. (48)

$$\sum_{i=1}^{\ell} \sum_{j=1}^{2^{i-1}} 2^{-\ell} \cdot \ell^{-1} \cdot 2^{-2\cdot i} = 2^{-\ell} \cdot \ell^{-1} \cdot (1 - 2^{-\ell})/2 \le 2^{-\ell}.$$

Therefore, the minimum of $|\boldsymbol{p}| = (2^{\ell-1}) \cdot p \asymp 2^{\ell} \cdot p$, subject to (46) and (47), is only of the order $2^{\ell} \cdot \ell$, and not of the order 2^{ℓ} as for $B^{(\ell,\boldsymbol{p}(\ell))}$, see Theorem 5.

III.2.3. Scalar SDEs. We consider a scalar autonomous SDE, cf. Section II.2, equation (9) with $r = d = 1$, i.e.,

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \qquad t \in [0,1],$$

with deterministic initial value $x_0 \in \mathbb{R}$ and driving (standard) Brownian motion W. We consider the same setting as for the associated quantization problem, see Section III.1, i.e., we assume that the drift coefficient $a\colon \mathbb{R} \to \mathbb{R}$ and the diffusion coefficient $b\colon \mathbb{R} \to \mathbb{R}$ are differentiable with bounded and Lipschitz continuous derivatives. This yields, in particular,

$$\mathrm{E} \sup_{t\in[0,1]} |X(t)|^2 < \infty. \tag{49}$$

In comparison, global Lipschitz continuous coefficients only yield square integrability of the solution process X, see Theorem 1. Furthermore, we assume that $b(x_0) \neq 0$ in order to exclude the case of a deterministic equation.

At first, we consider the random bit approximation of marginal distributions of X, linearly interpolated to random bit approximations of the distribution μ of X. To this end, we consider the deterministic equidistant Milstein scheme, as introduced in Section II.2.2. For convenience of the reader we restate the approximation scheme in a slightly different notation, which will indeed ease up the notation in the proofs of this section. For $m \in \mathbb{N}$ and $t_k = t_{k,m} = k/m$ the scheme reads as

$$\begin{aligned} X_m(t_0) &= x_0, \\ X_m(t_k) &= X_m(t_{k-1}) + a\big(X_m(t_{k-1})\big) \cdot m^{-1} + b\big(X_m(t_{k-1})\big) \cdot m^{-1/2} \cdot Y_k \\ &\qquad + \tfrac{1}{2}(b \cdot b')\big(X_m(t_{k-1})\big) \cdot m^{-1} \cdot \big(Y_k^2 - 1\big) \end{aligned}$$

where $k = 1, \dots, m$, and with normalized Brownian increments

$$Y_k = Y_{k,m} = m^{1/2} \cdot \big(W(t_k) - W(t_{k-1})\big).$$

Observe that the only source of randomness in this scheme (as in X) are the normalized Brownian increments Y_k. For $q \in \mathbb{N}$ we consider the q-bit approximations

$$Y_k^{(q)} = Y_{k,m}^{(q)} = \Phi^{-1} \circ T^{(q)} \cdot \Phi(Y_k),$$

as introduced in (24), which we know to have the optimal order of convergence $2^{-q/2} \cdot q^{-1/2}$, see Theorem 4. This leads to the *random bit Milstein scheme*

$$\begin{aligned} X_m^{(q)}(t_0) &= x_0, \\ X_m^{(q)}(t_k) &= X_m^{(q)}(t_{k-1}) + a\big(X_m^{(q)}(t_{k-1})\big) \cdot m^{-1} + b\big(X_m^{(q)}(t_{k-1})\big) \cdot m^{-1/2} \cdot Y_k^{(q)} \\ &\qquad + \tfrac{1}{2}(b \cdot b')\big(X_m^{(q)}(t_{k-1})\big) \cdot m^{-1} \cdot \big(\big(Y_k^{(q)}\big)^2 - 1\big) \end{aligned}$$

for $k = 1, \dots, m$.

Our procedure of proofing bit approximation results for μ closely follows Müller-Gronbach and Ritter [48], which deals with the corresponding quantization problem for μ. In the latter setting, approximations $\widetilde{Y}_k^{(q)}$ of Y_k with distributions in $\mathfrak{F}_{2^q}(\mathbb{R})$ and quantization error of order 2^{-q} are available, see Proposition 1. In fact, in Müller-Gronbach and Ritter [48] the following assumptions on the approximation $\widetilde{Y}^{(q)}$ of a standard normally distributed random variable Y have to be fulfilled, see Müller-Gronbach and Ritter [48, p. 1008]

(A1) $\mathrm{E}\big(\widetilde{Y}^{(q)}\big) = 0$ for every $q \in \mathbb{N}$,

(A2) $\sup_{q\in\mathbb{N}} \mathrm{E}\big|\widetilde{Y}^{(q)}\big|^r < \infty$ for every $r \geq 1$,

(A3) $\big(\mathrm{E}\big|Y - \widetilde{Y}^{(q)}\big|^2\big)^{1/2} \preccurlyeq 2^{-q}$ for every $q \in \mathbb{N}$.

In our present setting, (A1) and (A2) are fulfilled due to (27) and (28) from Theorem 4, respectively, and (A3) is replaced by

(A3') $\big(\mathrm{E}\big|Y - \widetilde{Y}^{(q)}\big|^2\big)^{1/2} \preccurlyeq 2^{-q/2} \cdot q^{-1/2}$ for every $q \in \mathbb{N}$,

which holds due to Theorem 4. Indeed it turns out that the order of convergence in (A3) has no influence on the proofs in Müller-Gronbach and Ritter [48]. Only in the results this order of convergence has to be replaced by the corresponding order of convergence in (A3').

Following this discussion, particularly the method of proof for Müller-Gronbach and Ritter [48, Lemma 3] is immediately applicable in the present setting of random bit approximations.

Lemma 10 (Cf. Müller-Gronbach and Ritter [48, Lemma 3] with (A3') instead of (A3)). *We have*

$$\Big(\mathrm{E}\Big(\max_{k=1,\dots,m}\big|X_m(t_k) - X_m^{(q)}(t_k)\big|^2\Big)\Big)^{1/2} \preccurlyeq m^{-1} \cdot 2^{-q/2} \cdot q^{-1/2}$$

uniformly in $m, q \in \mathbb{N}$.

Remark 18. We discuss the consequences of Lemma 10 for the present random bit approximation problem. Let $\nu_m^{(q)}$ denote the joint distribution of $X_m^{(q)}(t_1), \dots, X_m^{(q)}(t_m)$ and let ν denote the corresponding marginal distribution of X. Moreover, we consider the supremum norm on $V = \mathbb{R}^m$. Clearly, the joint distribution of $Y_1^{(q)}, \dots, Y_m^{(q)}$ belongs to $\mathfrak{U}(\mathbb{R}^m, m \cdot q)$, and therefore, $\nu_m^{(q)} \in \mathfrak{R}(\mathbb{R}^m, m \cdot q)$. Hence Theorem 3 and Lemma 10 yield

$$\mathrm{rbit}(\nu, m \cdot q) \leq \Big(\mathrm{E}\Big(\max_{k=1,\dots,m}\big|X(t_k) - X_m^{(q)}(t_k)\big|2\Big)\Big)^{1/2} \preccurlyeq m^{-1} + 2^{-q/2} \cdot q^{-1/2}. \tag{50}$$

Now, we turn to the random bit approximation of the distribution of X on the space $L_2 = L_2([0,1])$. The idea is to employ a time continuous version of the random bit Milstein scheme $X_m^{(q)}$, namely a linear interpolation, which is locally refined by random bit approximations of Brownian bridges that are scaled with respect to the local diffusion. At first we will consider the corresponding modifications of the Milstein scheme X_m, i.e., not in the random bit setting. The scheme obtained in that way will serve as an auxiliary scheme in our analysis of an upper bound of the random bit approximation error of μ, in the same way as X_m did for the upper bound of the random bit approximation error for the particular marginal distributions of X in Remark 18.

Observe that the linear interpolation of $X_m(t_{k-1})$ and $X_m(t_k)$ on the subinterval $[t_{k-1}, t_k]$ is given by

$$X_m(t) = (t - t_{k-1}) \cdot m \cdot X_m(t_k) - (t_k - t) \cdot m \cdot X_m(t_{k-1}),$$

with $t \in [t_{k-1}, t_k]$ and for $k = 1, \dots, m$. Moreover, for $k = 1, \dots, m$

$$B_k(t) = B_{k,m}(t) = m^{1/2} \cdot \big(W\big(t_{k-1} + t \cdot m^{-1}\big) - W(t_{k-1})\big) - t \cdot Y_k$$

with $t \in [0,1]$ defines a Brownian bridge from $W(t_{k-1})$ to $W(t_k)$ on the unit interval, cf. Remark 3. Hence we obtain a Brownian bridge on the subinterval $[t_{k-1}, t_k]$ by rescaling the Brownian bridge B_k with respect to the length of the subinterval. This leads to the approximation scheme

$$\begin{aligned}\overline{X}_m(t) = (t - t_{k-1}) \cdot m \cdot X_m(t_k) + (t_k - t) \cdot m \cdot X_m(t_{k-1}) \\ + b\big(X_m(t_{k-1})\big) \cdot m^{-1/2} \cdot B_k((t - t_{k-1}) \cdot m),\end{aligned}$$

where $t \in [t_{k-1}, t_k]$ and $k = 1, \dots, m$.

LEMMA 11 (Müller-Gronbach and Ritter [48, Lemma 4]). *We have*

$$\sup_{t \in [0,1]} \Big(\mathrm{E}\big|X(t) - \overline{X}_m(t)\big|^2\Big)^{1/2} \preccurlyeq m^{-1}.$$

Finally, we define a random bit approximation of $\overline{X}_m$ by means of $X_m^{(q)}$, and random bit approximations of the Brownian bridges B_k, $k = 1, \dots, m$. To this end, we choose $\boldsymbol{p}(\ell) \in \mathbb{N}^{2^\ell - 1}$ according to (44), and we consider the bit approximations $B_k^{(\ell, \boldsymbol{p}(\ell))} = B_{k,m}^{(\ell, \boldsymbol{p}(\ell))}$, see (43). Together with the random bit Milstein scheme $X_m^{(q)}$ this leads to the scheme

$$\begin{aligned}\overline{X}_m^{q,\ell}(t) = (t - t_{k-1}) \cdot m \cdot X_m^{(q)}(t_k) + (t_k - t) \cdot m \cdot X_m^{(q)}(t_{k-1}) \\ + b\big(X_m^{(q)}(t_{k-1})\big) \cdot m^{-1/2} \cdot B_k^{(\ell, \boldsymbol{p}(\ell))}((t - t_{k-1}) \cdot m),\end{aligned}$$

where $t \in [t_{k-1}, t_k]$ and $k = 1, \dots, m$. Observe that we only replaced X_m and B_k, in the definition of $\overline{X}_m$, by their random bit approximations $X_m^{(q)}$ and $B_k^{(\ell, \boldsymbol{p}(\ell))}$, respectively.

The distribution of $B_k^{(\ell, \boldsymbol{p}(\ell))}$ belongs to $\mathfrak{U}(L_2, |\boldsymbol{p}(\ell)|)$ with $|\boldsymbol{p}(\ell)| = 2^{\ell+2} - 2 \cdot \ell - 4$, see Theorem 5. Therefore, the distribution of $X_m^{(q,\ell)}$ belongs to $\mathfrak{R}(L_2, c(m,q,\ell))$, where

$$c(m,q,\ell) = m \cdot \big(q + 2^{\ell+2} - 2 \cdot \ell - 4\big). \tag{51}$$

Next, we need to determine how good the random bit approximation $X_m^{(q,\ell)}$ is in terms of q and ℓ and likewise how q and ℓ are related.

LEMMA 12. *We have*

$$\Big(\mathrm{E}\big\|X - X_m^{(q,\ell)}\big\|_{L_2}^2\Big)^{1/2} \preccurlyeq m^{-1} + 2^{-q/2} \cdot q^{-1/2} + m^{-1/2} \cdot 2^{-\ell/2}$$

uniformly in $m, q, \ell \in \mathbb{N}$.

PROOF. We closely follow the proof of Müller-Gronbach and Ritter [48, Lemma 5], and we write $\boldsymbol{p}$ instead of $\boldsymbol{p}(\ell)$ to simplify the notation. Due to Lemma 11 it suffices to analyze $\overline{X}_m - X_m^{(q,\ell)}$. We will analyze this difference part by part. Actually, we treat the differences between the linear interpolations of $\overline{X}_m$ and $X_m^{(q)}$ on the subintervals $[t_{k-1}, t_k]$ with $k = 1, \dots, m$ separately from the corresponding differences of the diffusion scaled Brownian bridges B_k and $B_k^{(\ell, \boldsymbol{p}(\ell))}$. For the latter we will add zero in a clever way, such that it can, more or less, be split up into the difference of the local diffusions and the differences between the Brownian bridges themselves. In that sense, the difference is rewritten as

$$\overline{X}_m - X_m^{(q,\ell)} = U_1 + U_2 + U_2,$$

where

$$U_1(t) = (t - t_{k-1}) \cdot m \cdot \big(X_m(t_k) - X_m^{(q)}(t_k)\big) + (t_k - t) \cdot m \cdot \big(X_m(t_{k-1}) - X_m^{(q)}(t_{k-1})\big),$$

as well as,

$$U_2(t) = \Big(b\big(X_m(t_{k-1})\big) - b\big(X_m^{(q)}(t_{k-1})\big)\Big) \cdot m^{-1/2} \cdot B_k\big((t - t_{k-1}) \cdot m\big),$$

and

$$U_3(t) = b\big(X_m^{(q)}(t_{k-1})\big) \cdot m^{-1/2} \cdot \Big(B_k\big((t - t_{k-1}) \cdot m\big) - B_k^{(\ell,\boldsymbol{p})}\big((1 - t_{k-1}) \cdot m\big)\Big)$$

for $t \in [t_{k-1}, t_k]$.

We put

$$\Delta = \Delta_m^{(q)} = \max_{k=1,\dots,m} \big|X_m(t_k) - X_m^{(q)}(t_k)\big|,$$

and observe that

$$\mathrm{E}\big(\Delta^2\big) \preccurlyeq m^{-2} + 2^{-1} \cdot q^{-1},$$

see Lemma 10. Since $\big((t - t_{k-1}) + (t_k - t)\big) \cdot m = 1$, we clearly have $|U_1(t)| \le \Delta$, and therefore

$$\mathrm{E}\big\|U_1\big\|_{L_2}^2 \preccurlyeq \mathrm{E}\big(\Delta^2\big).$$

For the second term, U_2, the Lipschitz continuity of b yields

$$\big|U_2(t)\big| \preccurlyeq \Delta \cdot m^{-1/2} \cdot B_k\big((t - t_{k-1}) \cdot m\big)$$

for $t \in [t_{k-1}, t_k]$ and $k = 1, \dots, m$. Moreover,

$$\mathrm{E}\big\|B_k\big((\cdot - t_{k-1}) \cdot m\big)\big\|_{L_2([t_{k-1},t_k])}^2 = \mathrm{E}\big\|m^{-1/2} \cdot B_k\big\|_{L_2([0,1])}^2 \asymp m^{-1},$$

see Lemma 1. We use the independence of Δ and $(B_1, \dots, B_m)$ together with Fubini's rule to conclude that

$$\begin{aligned}
\mathrm{E}\,\|U_2\|_{L_2([0,1])}^2 &= \sum_{k=1}^m \mathrm{E}\,\|U_2\|_{L_2([t_{k-1},t_k])} \\
&\preccurlyeq \sum_{k=1}^m \mathrm{E}\Big(\Delta^2 \cdot m^{-1} \cdot \big\|B_k\big((\cdot - t_{k-1}) \cdot m\big)\big\|_{L_2([t_{k-1},t_k])}^2\Big) \\
&= \mathrm{E}\big(\Delta^2\big) \cdot m^{-1} \cdot \sum_{k=1}^m \mathrm{E}\big\|B_k\big((\cdot - t_{k-1}) \cdot m\big)\big\|_{L_2([t_{k-1},t_k])}^2 \\
&\asymp m^{-1} \cdot \mathrm{E}\big(\Delta^2\big).
\end{aligned}$$

Combining the estimates for U_1 and U_2 we have

$$\mathrm{E}\big\|U_1 + U_2\big\|_{L_2}^2 \preccurlyeq m^{-2} + 2^{-q} \cdot q^{-1}.$$

It remains to estimate the term U_3. Clearly,

$$\begin{aligned}
&\sup_{m\in\mathbb{N},q\in\mathbb{N}} \mathrm{E} \max_{k=1,\dots,m} \big|X_m^{(q)}(t_k)\big|^2 \\
&\qquad \preccurlyeq \sup_{m\in\mathbb{N},q\in\mathbb{N}} \mathrm{E}\Big(\max_{k=1,\dots,m} \big|X_m(t_k) - X_m^{(q)}(t_k)\big|^2\Big) + \mathrm{E}\Big(\max_{k=1,\dots,m} \big|X_m(t_K)\big|^2\Big).
\end{aligned}$$

Estimating the first term with (50) and the second term by means of (49), we obtain

$$\sup_{m\in\mathbb{N},q\in\mathbb{N}} \mathrm{E} \max_{k=1,\dots,m} \big|X_m^{(q)}(t_k)\big|^2 < \infty. \tag{52}$$

Since b is in particular globally Lipschitz continuous, it satisfies a linear growth condition, see Remark 5, (52) carries over to $\big|b\big(X_m^{(q)}(t_k)\big)\big|^2$. Moreover,

$$\begin{aligned}\mathrm{E}\Big\|B_k\big((\cdot - t_{k-1})\cdot m\big) - B_k^{(\ell,\boldsymbol{p})}\big((\cdot - t_{k-1})\cdot m\big)\Big\|^2_{L_2([t_{k-1},t_k])} &= m^{-1}\cdot \mathrm{E}\Big\|B_k - B_k^{(\ell,\boldsymbol{p})}\Big\|^2_{L_2([0,1])}\\ &\asymp m^{-1}\cdot 2^{-\ell},\end{aligned}$$

see Corollary 1. Together with the independence of $\max_{k=1,\dots,m}\big|X_m^{(q)}(t_k)\big|$ and $(B_1,\dots,B_m)$ we obtain

$$\mathrm{E}\big\|U_3\big\|^2_{L_2} \preccurlyeq m^{-1}\cdot 2^{-\ell}. \qquad \blacksquare$$

Remark 19. Suppose that the Euler-Maruyama scheme, instead of the Milstein scheme, would be employed in the definition of $X_m^{(q,\ell)}$. Then the first term in the upper bound from Lemma 12 would change from m^{-1} to $m^{-1/2}$, so that altogether

$$\Big(\mathrm{E}\big\|X - X_m^{(q,\ell)}\big\|^2_{L_2}\Big)^{1/2} \preccurlyeq m^{-1/2} + 2^{-q/2}\cdot q^{-1/2},$$

which does not suffice for our purposes.

With the result from Lemma 12 we can establish a constructive upper bound on the random bit approximation error $\mathrm{rbit}(\mu,p)$ for the distribution y of the solution process X of the scalar SDE on L_2.

Theorem 6. *Let μ denote the distribution of X on L_2. Furthermore, let*

$$m(\ell) = 2^\ell, \qquad q(\ell) = 2\cdot\ell,$$

and $c(\ell) = c\big(m(\ell), q(\ell), \ell\big)$. Then we have

$$\mathrm{rbit}\big(\mu, c(\ell)\big) \preccurlyeq \Big(\mathrm{E}\big\|X - X_{m(\ell)}^{(q(\ell),\ell)}\big\|^2_{L_2}\Big)^{1/2} \preccurlyeq \big(c(\ell)\big)^{-1/2} \tag{53}$$

and

$$c(\ell) = 2^{\ell+2}\cdot\big(2^\ell - 1\big) \asymp 2^{2\cdot\ell}.$$

Proof. We write m and q instead of $m(\ell)$ and $q(\ell)$, respectively, to simplify the notation. By definition,

$$\mathrm{rbit}\big(\mu, c(\ell)\big) \le \Big(\mathrm{E}\big\|X - X_m^{(q,\ell)}\big\|^2_{L_2}\Big)^{1/2}.$$

Hence we show that

$$\Big(\mathrm{E}\big\|X - X_m^{(q,\ell)}\big\|^2_{L_2}\Big)^{1/2} \preccurlyeq \big(c(\ell)\big)^{-1/2}. \tag{54}$$

By Lemma 12 we have

$$\Big(\mathrm{E}\big\|X - X_m^{(q,\ell)}\big\|^2_{L_2}\Big)^{1/2} \preccurlyeq 2^{-\ell} + 2^{-\ell}\cdot(2\cdot\ell)^{-1/2} + 2^{-\ell/2}\cdot 2^{-\ell/2} \preccurlyeq 2^{-\ell}.$$

The explicit formula for $c(\ell)$ obviously holds true, see (51). This completes the proof for the asymptotic upper bound (54). $\blacksquare$

Corollary 2. *Let μ denote the distribution of X on L_2. Then we have*

$$\mathrm{rbit}(\mu,p) \asymp \mathrm{quant}(\mu,p) \asymp p^{-1/2}.$$

In particular this (weak) order of convergence is achieved by $X_{m(\ell)}^{(q(\ell),\ell)}$*, as defined in Theorem 6, i.e.,*

$$\left(\mathrm{E}\|X - X_{m(\ell)}^{(q(\ell),\ell)}\|_{L_2}^2\right)^{1/2} \asymp \mathrm{rbit}\big(\mu, c(\ell)\big).$$

Proof. By Proposition 4 we have the lower bound

$$p^{-1/2} \preccurlyeq \mathrm{quant}(\mu, p).$$

Moreover, by definition, cf. also Remark 10, we have

$$\mathrm{quant}(\mu, p) \le \mathrm{rbit}(\mu, p).$$

Finally, Theorem 6 yields the upper bound

$$\mathrm{rbit}(\mu, p) \preccurlyeq p^{-1/2}.$$

The statement for $X_{m(\ell)}^{(q(\ell),\ell)}$ is now an immediate consequence of (53). ∎

III.2.4. Gaussian Random Elements. In this section, we consider a centered Gaussian random element X that takes values in a separable Hilbert space $(H, \|\cdot\|_H)$ and that has an infinite-dimensional support. Moreover, let $\langle\cdot,\cdot\rangle_H$ denote the scalar product on H.

The quantization and the random bit approximation of X are considered in a separable section, since Gaussian elements are not in the main scope of the thesis. Nevertheless, they are closely related to the Brownian motion and the Brownian bridge. Indeed X being the Brownian motion is a particular Gaussian process.

The random bit approximation result, i.e., Theorem 7, has been established in Giles et al. [27], and the method of proof is the same as for the Brownian bridge in Section III.2.2. At first we rewrite X in terms of its Karhunen-Loève expansion, which is known to be optimal for the quantization of Gaussian measures, see Luschgy and Pagès [41, i.p. Theorem 3.1 and Theorem 3.2], i.e.,

$$X = \sum_{i=1}^{\infty} \lambda_i^{1/2} \cdot Y_i \cdot e_i$$

with convergence, e.g., in mean square with respect to the norm on H. Here, $(e_i)_{i\in\mathbb{N}}$ is an orthonormal system in H and $Y_i = \langle X, e_i\rangle/\lambda_i^{1/2}$, $i \ge 1$, is a sequence of independent standard normally distributed random variables. Furthermore, the variances λ_i are ordered such that they form a non-increasing sequence of strictly positive numbers.

In the following we assume that

$$\lim_{i\to\infty} \lambda_i \cdot i^{\beta} \cdot (\ln(i))^{\alpha} \in \,]0, \infty[, \tag{55}$$

where $\beta < 1$ and $\alpha \in \mathbb{R}$. There are cases in which the asymptotic behaviour of λ_i is known. We consider, e.g., X a fractional Brownian motion on $[0,1]$ with values in $L_2([0,1])$ and with Hurst exponent $h \in \,]0,1[$, i.e., X has the covariance function

$$\tfrac{1}{2} \cdot \left(s^{2\cdot h} + t^{2\cdot h} - |s-t|^{2\cdot h}\right),$$

for all $s, t \in [0,1]$. Then $\beta = 2\cdot h + 1$ and $\alpha = -2\cdot\beta$, see Luschgy and Pagès [43, Proposition 3.2]. The particular case $\beta = 2$ and $\alpha = 0$ corresponds to a Brownian motion as well as to a Brownian bridge, see also Luschgy and Pagès [41, Example 2.3 (ii)-(iii)] for an explicit formula of the λ_i's. For a d-dimensional analogon, see, e.g., Luschgy and Pagès [43, Proposition 3.4].

PROPOSITION 5. *Let μ denote the distribution of X on H and assume that (55) is satisfied. Then*

$$\operatorname{quant}(\mu, p) \asymp p^{-(\beta-1)/2} \cdot (\ln(p+1))^{-\alpha/2}.$$

As for the random bit approximation of the distribution of the Brownian bridge in Section III.2.2 we consider a two stage discretization. For $m \in \mathbb{N}$ we cut off the Karhunen-Loève expansion of X after m terms, i.e.

$$X^{(m)} = \sum_{i=1}^{m} \lambda_i^{1/2} \cdot Y_i \cdot e_i. \tag{56}$$

Due to the ordering $\lambda_1 \geq \lambda_2 \geq \dots$ these m terms are also called the first m principle components of X. This approximation has the following asymptotic property.

LEMMA 13. *Let X be a centered Gaussian random element in a separable Hilbert space H with Karhunen-Loève expansion $X = \sum_{i=1}^{\infty} \lambda_i^{1/2} \cdot Y_i \cdot e_i$. Furthermore, let the variances λ_i satisfy (55) and let $X^{(m)} = \sum_{i=1}^{m} \lambda_i^{1/2} \cdot Y_i \cdot e_i$ as in (56) denote the sum of the principle components of X. Then*

$$\mathrm{E}\big\|X - X^{(m)}\big\|_H^2 \preccurlyeq m^{-(\beta-1)} \cdot \big(\ln(m+1)\big)^{-\alpha}.$$

PROOF. Observe that

$$\mathrm{E}\big\|X - X^{(m)}\big\|_H^2 = \sum_{i=m+1}^{\infty} \lambda_i \cdot \mathrm{E}(Y_i)^2 = \sum_{i=m+1}^{\infty} \lambda_i.$$

Since $\lambda_i \preccurlyeq i^{-\beta} \cdot (\ln(i))^{-\alpha}$ uniformly in $i \in \mathbb{N}$, we have in particular

$$\sum_{i=m+1}^{\infty} \lambda_i \preccurlyeq \sum_{i=m+1}^{\infty} i^{-\beta} \cdot \big(\ln(i)\big)^{-\alpha} \leq \big(\ln(m+1)\big)^{-\alpha} \cdot \sum_{i=m+1}^{\infty} i^{-\beta}.$$

Using an integral estimate for $\sum_{i=m+1}^{\infty} i^{-\beta}$, we get

$$\mathrm{E}\big\|X - X^{(m)}\big\|_H^2 \preccurlyeq \big(\ln(m+1)\big)^{-\alpha} \cdot \int_m^{\infty} x^{-(\beta-1)}\,\mathrm{d}x = \big(\ln(m+1)\big)^{-\alpha} \cdot m^{-(\beta-1)},$$

as claimed. ∎

On top of the approximation (56) we consider a vector

$$\boldsymbol{p} = (p_1, \dots, p_m) \in \mathbb{N}^m$$

of bit numbers, and we define

$$X^{(m,\boldsymbol{p})} = \sum_{i=1}^{m} \lambda_i^{1/2} \cdot Y_i^{(p_i)} \cdot e_i,$$

where $Y_i^{(p_i)}$ is the bit approximation of Y_i according to (24). Obviously, the distribution of $X^{(m,\boldsymbol{p})}$ belongs to $\mathfrak{U}(H, |\boldsymbol{p}|)$ with

$$|\boldsymbol{p}| = \sum_{i=1}^{m} p_i.$$

THEOREM 7. *Let μ denote the distribution of the Gaussian random element X on H, and assume that (55) is satisfied. We define $\boldsymbol{p}(m) \in \mathbb{N}^m$ for $m \in \mathbb{N}$ by*

$$p_i(m) = \lceil \max(\widetilde{p}_i, 1) \rceil$$

for $i = 1, \ldots, m$, with

$$\widetilde{p}_i = \beta \cdot \log_2(m/i) + \max(\alpha, 0) \cdot \log_2\bigl(\log_2(m+1)/\log_2(i+1)\bigr).$$

Then we have

$$\operatorname{rbit}(\mu, |\boldsymbol{p}(m)|) \le \Bigl(\mathrm{E}\bigl\|X - X^{(m,\boldsymbol{p}(m))}\bigr\|_H^2\Bigr)^{1/2} \preccurlyeq |\boldsymbol{p}(m)|^{-(\beta-1)/2} \cdot (\ln(|\boldsymbol{p}(m)|+1)^{-\alpha/2} \tag{57}$$

and

$$|\boldsymbol{p}(m)| \asymp m.$$

PROOF. We write $\boldsymbol{p}, p_i$, and $\widetilde{p}_i$ instead of $\boldsymbol{p}(m), p_i(m)$, and $\widetilde{p}_i(m)$, respectively, to simplify the notation. We first show $|\boldsymbol{p}(m)| \asymp m$. Due to $p_i \ge 1$ for $i = 1, \ldots, m$, the lower bound obviously holds true. For the upper bound we show

$$p_i \preccurlyeq 1 + \ln(m/i). \tag{58}$$

By definition, $p_i \le 1 + \widetilde{p}_i$, so we show $\widetilde{p}_i \preccurlyeq \ln(m/i)$. Clearly, the statement is true for $\alpha \le 0$. It remains to show

$$\log_2\bigl(\log_2(m+1)/\log_2(i+1)\bigr) \le \log_2(m/i).$$

Since the statement is true for $i = 1$, this is equivalent to $x \cdot \bigl(\log_2(x+1)\bigr)^{-1}$ being monotonically increasing, i.e., having a non-negative derivative, on $]2, \infty[$. Indeed, we have

$$\frac{\mathrm{d}}{\mathrm{d}x}\bigl(x \cdot \bigl(\log_2(x+1)\bigr)\bigr)^{-1} = \frac{\ln(2)}{\bigl(\ln(x+1)\bigr)^2} \cdot \bigl(\ln(x+1) - x/x + 1\bigr)$$

such that $\ln(x+1) > 1$ for $x > 2$ yields the monotonicity claim for $x \cdot \bigl(\log_2(x+1)\bigr)^{-1}$.

Due to $\frac{\mathrm{d}}{\mathrm{d}x}\bigl(x \cdot \ln(x)\bigr) = \ln(x) + 1$, and using an integral estimate for $\sum_{i=1}^m \ln(x)$ we obtain from (58) that

$$|\boldsymbol{p}| \preccurlyeq m + m \cdot \ln(m) - \sum_{i=1}^m \ln(i) \preccurlyeq m + m \cdot \ln(m) - \int_1^m \bigl(x \cdot \ln(x) - 1\bigr)\,\mathrm{d}x = 2 \cdot m - 1,$$

and therefore $|\boldsymbol{p}| \asymp m$.

By definition,

$$\operatorname{rbit}(\mu, |\boldsymbol{p}|) \le \Bigl(\mathrm{E}\bigl\|X - X^{(m,\boldsymbol{p})}\bigr\|_H^2\Bigr)^{1/2}.$$

Hence we show that

$$\Bigl(\mathrm{E}\bigl\|X - X^{(m,\boldsymbol{p})}\bigr\|_H^2\Bigr)^{1/2} \preccurlyeq m^{-(\beta-1)/2} \cdot \bigl(\ln(m+1)\bigr)^{-\alpha/2}.$$

First of all, we note that

$$\mathrm{E}\bigl\|X - X^{(m)}\bigr\|_H^2 \preccurlyeq m^{-(\beta-1)} \cdot \bigl(\ln(m+1)\bigr)^{-\alpha},$$

see Lemma 13. Furthermore, Theorem 4 yields together with (55)

$$\mathrm{E}\bigl\|X^{(m)} - X^{m,\boldsymbol{p}(m))}\bigr\|_H^2 = \sum_{i=1} \mathrm{E}\bigl(Y_i - Y_i^{(p_i)}\bigr)^2 \cdot \lambda_i \asymp \sum_{i=1}^m 2^{-p_i} \cdot p_i^{-1} \cdot i^{-\beta} \cdot \bigl(\ln(i+1)\bigr)^{-\alpha}$$

uniformly in $m \in \mathbb{N}$ and $\boldsymbol{p} \in \mathbb{N}^m$. Since $p_i \geq 1$, we have

$$2^{-p_i} \cdot p_i^{-1} \leq 2^{-p_i} \leq (m/i)^{-\beta} \cdot \big(\ln(m+1)/\ln(i+1)\big)^{-\alpha}$$

uniformly in $m \in \mathbb{N}$ and $i = 1, \dots, m$, and therefore

$$\mathrm{E}\big\|X^{(m)} - X^{(m,\boldsymbol{p})}\big\|_H^2 \preccurlyeq m^{-(\beta-1)} \cdot \big(\ln(m+1)\big)^{-\alpha}. \qquad \blacksquare$$

Corollary 3. *Let μ denote the distribution of the Gaussian random element X on H, as in Theorem 7. Then we have*

$$\mathrm{rbit}(\mu, p) \asymp \mathrm{quant}(\mu, p) \asymp p^{-(\beta-1)/2} \cdot (\ln(p+1))^{-\alpha/2}.$$

In particular, this (weak) order of convergence is achieved by $X^{(m,\boldsymbol{p}(m))}$, as defined in Theorem 7, i.e.

$$\Big(\mathrm{E}\big\|X - X^{(m,\boldsymbol{p}(m))}\big\|_H^2\Big)^{1/2} \asymp \mathrm{rbit}(\mu, |\boldsymbol{p}(m)|).$$

Proof. By Proposition 5 we have the lower bound

$$p^{-(\beta-1)/2} \cdot \big(\ln(p+1)\big)^{-\alpha/2} \preccurlyeq \mathrm{quant}(\mu, p).$$

Moreover, by definition, cf. also Remark 10, we have

$$\mathrm{quant}(\mu, p) \leq \mathrm{rbit}(\mu, p).$$

Finally, Theorem 7 yields the upper bound

$$\mathrm{rbit}(\mu, p) \preccurlyeq p^{-(\beta-1)/2} \cdot \big(\ln(p+1)\big)^{-\alpha/2}.$$

The statement for $X^{(m,\boldsymbol{p}(m))}$ is now an immediate consequence of (57). $\blacksquare$

III.3. Relation to Kolmogorov Widths

In this section we are going to compare the quantization problem, and therefore implicitly the random bit approximation problem, for particular choices of V and $\mu \in \mathfrak{M}(V)$, to a closely related approximation problem, the average Kolmogorov widths. For a discussion of average Kolmogorov widths we refer to Creutzig et al. [12]. While the quantization and the random bit approximation problem are based on probability measures with a finite support, here we consider probability measures supported on finite dimensional subspaces.

We motivate the average Kolmogorov widths as follows. We consider the space $L_2 = L_2([0,1])$ and the quantization problem for the Wiener measure μ on L_2. An n-quantization $\nu \in \mathfrak{F}_n(L_2)$ corresponds to (up to) n fixed L_2 functionals, which we are allowed to use for the approximation of each Brownian path. Then it is somehow natural to ask whether the approximation error $d(\mu, \nu)$ can be reduced if one takes into account all linear combinations of those (up to) n functionals.

Definition 2. As in Section III.1 let $(V, \|\cdot\|_V)$ be a separable Banach space, let $\mathfrak{M}(V)$ the set of all Borel probability measures on V, and let d denote the Wasserstein distance of order 2, as defined in (11). Recall for $\mu_1, \mu_2 \in \mathfrak{M}(V)$

$$d(\mu_1, \mu_2) = \inf\big\{\big(\mathrm{E}\,|X_1 - X_2|^2\big)^{1/2} \colon P_{X_1} = \mu_1, P_{X_2} = \mu_2\big\}.$$

Then for $\mu \in \mathfrak{M}(V)$ the *k-th average Kolmogorov width of order 2* is given by

$$\mathrm{kol}_k(\mu) = \inf\big\{d(\mu, \nu) \colon \dim(\mathrm{span}(\mathrm{supp}(\nu))) \leq k\big\}.$$

REMARK 20. For particular approximation problems, we compare the asymptotic behaviour of average Kolmogorov widths to the corresponding results for quantization. We consider $V = L_2 = L_2([0,1])$.

(i) Let μ be the Wiener measure on L_2. Then by Creutzig et al. [12, Proposition 2]

$$\mathrm{kol}_k(\mu) \asymp k^{-1/2},$$

while for the quantization problem, Proposition 2 yields

$$\mathrm{quant}(n,\mu) \asymp (\log(n))^{-1/2}.$$

(ii) Let μ be the distribution of a scalar SDE with coefficients as in Section III.1. Then by Creutzig et al. [12, Proposition 3]

$$\mathrm{kol}_k(\mu) \asymp k^{-1/2},$$

while for the quantization problem, Proposition 4 yields

$$\mathrm{quant}(n,\mu) \asymp (\log(n))^{-1/2}.$$

Let us finally mention that the results for the average Kolmogorov widths of order 2 carry over to average Kolmogorov widths of order r, i.e., in the definition of $\mathrm{kol}_k(\mu)$, the Wasserstein distance of order 2 is replaced by the Wasserstein distance of order r.

CHAPTER IV

Quadrature of SDEs

In this section we study the quadrature problem for $\mathrm{E}(f(X))$, where X is the solution of an SDE and f a measurable mapping from the path space of X into the real numbers. Namely, let $d, r \in \mathbb{N}$, we consider, cf. Section II.2, SDEs of the form

$$\mathrm{d}X(t) = a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \qquad t \in [0,1], \tag{59}$$

with deterministic initial value

$$X(0) = x_0 \in \mathbb{R}^r,$$

and with a d-dimensional driving Brownian motion W. Furthermore, the drift coefficient $a\colon \mathbb{R}^r \to \mathbb{R}^r$ and the diffusion coefficient $b\colon \mathbb{R}^r \to \mathbb{R}^{r\times d}$ are assumed to be globally Lipschitz functionals. Observe that by Theorem 1 there exists a (up to indistinguishability) unique pathwise time-continuous and square-integrable solution $X = (X(t))_{t\in[0,1]}$. Moreover,

$$f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}$$

is assumed to be Lipschitz continuous with respect to either the supremum-norm or the (classical) L_p-norm on $C([0,1],\mathbb{R}^r)$ with Lipschitz constant of at most 1. Indeed we consider the corresponding classes

$$F_\infty = \bigl\{f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}\colon |f(x) - f(y)| \le \|x-y\|_{\sup}\bigr\}$$

and

$$F_p = \bigl\{f\colon C([0,1],\mathbb{R}^r) \to \mathbb{R}\colon |f(x) - f(y)| \le \|x-y\|_{L_p}\bigr\}, \qquad 1 \le p < \infty,$$

of such Lipschitzian functionals. Actually, it would suffice to assume that the Lipschitz constant is bounded. But for convenience of the reader, and to ease up the notation, we assume this bound to be 1. For example one can think of X being a geometric Brownian motion modeling a stock price and f a payoff functional like, e.g., an Asian option.

In this setting we will construct two particular random bit algorithms for the quadrature of $\mathrm{E}(f(X))$. One algorithm that we can analyze analytically and one algorithm that is only investigated numerically. Nevertheless, the latter algorithm is (at least in our opinion) the intuitively 'better' algorithm, since it has a structure that immediately allows a desirable adaptive multilevel approach, which will be introduced and discussed in Section IV.3.3, and in contrast to the other algorithm it relies on the asymptotically optimal random bit approximation of the Brownian bridge as derived in Section III.2.2.

Of course, we have to give meaning to the notion of a 'better' algorithm, i.e., we need a framework in which we can, in particular, compare different algorithms. This framework is set up in Section IV.1. Let A be a given algorithm. At first, we define, similar as for the quantization and the random bit approximation problem, the stochastic-L_2-approximation error, $\mathrm{error}(A,f)$, of the algorithm A applied to f. Moreover, we define the cost, $\mathrm{cost}(A,f)$, of the algorithm A in terms of the sum of all operations that have to be carried out. One quantity of interest for A will then be the relation between $\mathrm{error}(A,f)$ and $\mathrm{cost}(A,f)$, more

precisely, we want to determine $\mathrm{cost}(A, f)$ in dependence on $\mathrm{error}(A, f)$ Clearly, the cost may not only depend on the solution X of the considered SDE but also on the functional $f \in F$. Indeed we will use a worst case analysis on the classes F_∞ and F_p, respectively.

Section IV.2 is devoted to the method of direct simulation leading to the classical Monte Carlo Euler-Maruyama algorithm. Moreover, we introduce the control variate method, a variance reduction technique, that will lead to multilevel Monte Carlo algorithms, which will be discussed in Section IV.3. Here, more precisely in Section IV.3.1, we present the (standard) multilevel Monte Carlo Euler-Maruyama method, as first introduced in Giles [24]. The approach is similar to the one used in Heinrich [33] in the context of parametric integration. In Section IV.3.2 we restrict ourselves to functionals f from the class F_p and we address an efficiency problem of multilevel Monte Carlo algorithms in dependence on $f(x)$, i.e., in dependence on the solution X of the considered SDE. This will lead to an adaptive multilevel Monte Carlo algorithm, as introduced in Giles [24, Section 5], which we will discuss in Section IV.3.3. We will also discuss modifications of this algorithm, which can, e.g., be found in Giles [25, p. 283], and which are supposed to lead to more (numerical) stability of the algorithm. It will turn out that the adaptive algorithm is fitted to the class of functionals F_p. Therefore, we will briefly discuss an intuitive adaption to the class F_∞, which may unfortunately lead to 'instability' of the algorithm, but helps to get a better understanding of the problematic in the transition from F_p to F_∞. For a general overview on multilevel Monte Carlo algorithms we refer to Giles [25].

Having this whole machinery at hand, we turn to the construction of two particular random bit multilevel Monte Carlo algorithms. To this end, at first we introduce the two random bit Euler-Maruyama schemes these algorithms are based on. This we do in Section IV.4, where we also analyze one of the two schemes, later on referred to as Scheme 1. In Section IV.5 we finally define the random bit multilevel Monte Carlo algorithms themselves. We also analyze the algorithm based on Scheme 1, denoted by $A^q_{\varepsilon,F}$. Furthermore, in Section IV.5.1, we present a variant of Bakhvalov's trick to reduce the number of random bits used by $A^q_{\varepsilon,F}$, and we exploit this trick to improve the upper bound on the cost of $A^q_{\varepsilon,F}$ in terms of ε, which is derived earlier in Section IV.5, at least in terms of its weak asymptotics for small ε. For the classical Bakhvalov trick see, e.g., Bakhvalov [2] and also Heinrich et al. [36] and the references therein.

The results presented in Section IV.4 and Section IV.5 are published in Giles et al. [28].

IV.1. Algorithms, Error, and Cost

In this section we define what kind of algorithms are admissible for the approximation of

$$S(f) = \mathrm{E}(f(X))$$

with X being the solution of an SDE of the form (59), and $f \in F$, with either $F = F_\infty$ or $F = F_p$, being a Lipschitz continuous functional (with Lipschitz constant 1). Moreover, we define the error criterion for the admissible algorithms as well as how we calculate the cost of such an algorithm.

To implement an algorithm we are allowed to use the following operations:

- Input: finitely many (input) functionals $f_1, \dots, f_k$, $k \in \mathbb{N}$ from F as well as the drift and diffusion coefficients a and b, and finitely many real numbers $x_1, \dots, x_{\widetilde{k}}$, $\widetilde{k} \in \mathbb{N}$.
- Arithmetic operations: addition, subtraction, multiplication, and division.
- Jumps depending on comparisons of real numbers, as e.g. $x \leq y$ for $x, y \in \mathbb{R}$.

- Elementary functions like exponential, logarithm, sine, cosine, or the floor and ceiling function.
- Oracles that evaluate the coefficients a and b at each point $x \in \mathbb{R}^r$.
- An oracle that provides evaluations of the input functionals $f_1, \dots, f_k$ on a subset G of $C([0,1], \mathbb{R}^r)$, i.e., we assume that the oracle can calculate $f_i(g)$ for $i = 1, \dots, k$ and for all $g \in G$. Thinking of Euler-Maruyama approximation schemes, G could be the set of all piecewise linear functions in $C([0,1], \mathbb{R}^r)$.
- Output: A finite sequence of real numbers $x_1^*, \dots, x_k^*$, $k \in \mathbb{N}$.

Furthermore, we are allowed to address memory indirectly on a countable number of registers. Moreover, we have access to random numbers, here we distinguish between two cases:

1.) The algorithm has access to a random number generator that provides random numbers according to the uniform distribution on $[0,1]$.
2.) The algorithm has access to a random bit generator that provides random bits according to the uniform distribution on $\{0,1\}$.

Definition 3. A *Monte Carlo algorithm* or *randomized algorithm* is a program that uses a finite number of operations from the above list and finitely many calls to a random number generator according to 1.). If the random number generator is replaced by a random bit generator according to 2.), then we also use the notion of a *random bit* or *restricted Monte Carlo algorithm* or simply *random bit* or *restricted algorithm*. The corresponding model of computation is called *real number model (with oracles)*, see, e.g., Novak [52].

To define the error of a (restricted) Monte Carlo algorithm A, we need to introduce some more notations. By $\mathfrak{I}$ we denote the set of all tuples of functions and real numbers that can serve as an input for algorithm A. Furthermore, we denote by $\mathfrak{S}$ the space of sequences over $[0,1]$ respective $\{0,1\}$. For technical reasons we assume that we have a sequence U of independent random variables $U_i, i \in \mathbb{N}$ on a common probability space $(\widetilde{\Omega}, \widetilde{A}, \widetilde{P})$ each of which is uniformly distributed on $[0,1]$ respective $\{0,1\}$, i.e., one realization of one U_i corresponds to one call to the random number respective random bit generator. The fact that A can only finitely often call to such a generator (otherwise it would not terminate) corresponds to A using only a finite segment from the sequence U. But there is in general no bound on the length of this segment. Finally, we denote by

$$\mathfrak{T} \subseteq \mathfrak{I} \times \mathfrak{S}$$

the set of all inputs for which the algorithm terminates, also called *termination set of* A, and $\widehat{A}\colon \mathfrak{T} \to \mathbb{R}$ is the mapping that maps the input (I, s) on the output $A(I) = A(I, s)$. Then the algorithm A formally corresponds to the mapping

$$A\colon \mathfrak{I} \times \widetilde{\Omega} \to \mathbb{R}, \quad A(I, \widetilde{\omega}) = \begin{cases} \widehat{A}(I, U(\widetilde{\omega})), & \text{if } (I, U(\widetilde{\omega})) \in \mathfrak{T}, \\ 0, & \text{else.} \end{cases}$$

In the following, we assume that A terminates with probability one, i.e., there exists a set $\widetilde{A} \in \widetilde{\mathfrak{A}}$ with $\widetilde{P}(\widetilde{A}) = 1$ and $(I, U(\widetilde{\omega})) \in \mathfrak{T}$ for all $\widetilde{\omega} \in \widetilde{A}$. Moreover, the mapping $A(I, \cdot)\colon \widetilde{\Omega} \to \mathbb{R}$ is supposed to be a random variable, i.e., A is a measurable mapping. For notational convenience we will in general write $A(I)$ instead of $A(I, \cdot)$. This leads to the following definition.

DEFINITION 4. Let $f \in F$ and $A(f)$ square-integrable. Then the *error* of the (restricted) Monte Carlo algorithm A applied to input f for the approximation of $S(f)$ is defined by

$$\operatorname{error}(A, f) = \big(\mathrm{E}(S(f) - A(f))^2\big)^{1/2}.$$

This error is due to its structure also known as the *root-mean-squared error* of A applied to f. Accordingly, the *worst case error* of A on the class F is defined by

$$\operatorname{error}(A, F) = \sup_{f \in F} \operatorname{error}(A, f).$$

Sometimes we may also write $\operatorname{error}(A, F, X)$ to stress that A is an algorithm for the approximation of $\mathrm{E}(f(X))$.

REMARK 21. Observe that the squared error of A applied to f can be decomposed into the squared *bias* and the variance of A

$$(\operatorname{error}(A, f))^2 = \big(\operatorname{Bias}(A(f))\big)^2 + \operatorname{Var}(A(f)) = \big(S(f) - \mathrm{E}(A(f))\big)^2 + \operatorname{Var}(A(f)).$$

Here, $\operatorname{Bias}(A(f)) = \mathrm{E}\big(S(f) - A(f)\big)$ is typically an error that is induced by the underlying approximation scheme for the approximation of the solution X of the SDE under consideration, i.e., $\operatorname{Bias}(A(f))$ can be interpreted as the least error we have using a particular approximation scheme. The variance $\operatorname{Var}(A(f))$ of the algorithm can be interpreted as an additional statistical error due to $A(f)$ being a random quantity (for the approximation of an expectation).

Analogously to $\operatorname{error}(A, f)$ we use the notations

$$\operatorname{Bias}(A(F)) = \sup_{f \in F} \operatorname{Bias}(A(f)) \qquad \text{and} \qquad \operatorname{Var}(A(F)) = \sup_{f \in F} \operatorname{Var}(A(f)).$$

If $\operatorname{Bias}(A, F) \neq 0$, we call A a *biased algorithm*, and we call A *unbiased* if $\operatorname{Bias}(A, F) = 0$.

It remains to define the (computational) cost of a (restricted) Monte Carlo algorithm A. Actually, the cost for a given input I can be measured counting all the operations that are carried out by A. In this way, up to a multiplicative constant, the (computational) cost represents the expected runtime of the algorithm on a computer. In this procedure we ignore the cost associated to the input and output operations, as well as, the cost for simple copy commands and for memory operations, like reading from and writing to memory. Note that the later heavily depends on the employed hardware, anyway.

We assume that all non-algorithm specific and non-input depending operations, i.e., arithmetic operations, comparisons and jumps, and evaluations of elementary functions can be carried out at unit cost, i.e., cost 1. Moreover, we suppose that each call to a random number or random bit generator has unit cost, too. Furthermore, we assume that we have oracles that can evaluate the coefficients a and b at a single point $x \in \mathbb{R}^r$ at unit cost, respectively, as well as that for each functional $f \in F$ there is an oracle such that f applied to a continuous functional from $C([0,1], \mathbb{R}^r)$, implicitly given by $k \in \mathbb{N}$ time points from $[0,1]$, can be evaluated at cost k, by this oracle.

Observe that in the literature, see, e.g., Müller-Gronbach et al. [49, p. 19], the cost for the evaluation of the coefficients a and b is often regarded as a constant $c \gg r \geq 1$. This makes sense because the evaluation needs the assignment of r coordinates of a sample point from $\mathbb{R}^r$ and may require complicated computations or even physical measurements. Nevertheless, for our purposes (where constants do not matter in the end) it is convenient to consider $c = 1$. The same holds true for the evaluation of each functional f from F.

Finally, we observe that in general the (computational) cost of A does not only depend on the input f but also on the random numbers respective random bits that are used by the algorithm. Hence the (computational) cost is a random quantity. Analogously to defining the error of A applied to f, we assume that the (computational) cost of computing A for input f is a random variable

$$\operatorname{cost}(A, f, \cdot)\colon \widetilde{\Omega} \to \mathbb{N}.$$

Definition 5. Let $f \in F$ and let $\operatorname{cost}(A, f, \cdot) \in L_1(\widetilde{\Omega})$. The *(computational) cost* of applying the (restricted) Monte Carlo algorithm A to input f is defined by

$$\operatorname{cost}(A, f) = \mathrm{E}\big(\operatorname{cost}(A, f, \cdot)\big).$$

The *worst case (computational) cost* of A on the class F is defined by

$$\operatorname{cost}(A, F) = \sup_{f \in F} \mathrm{E}\big(\operatorname{cost}(A, f)\big).$$

Having the definitions of the error and the cost of an (Monte Carlo) algorithm at hand, we can relate them in a way that allows us to compare classes of algorithms.

Definition 6. Let $\varepsilon > 0$. Then we call

$$\operatorname{comp}(\varepsilon, F, X) = \inf\big\{\operatorname{cost}(A, F)\colon A \text{ is a Monte Carlo algorithm}, \operatorname{error}(A, F, X) \le \varepsilon\big\}$$

the *ε-complexity* for the approximation of S (on the class F) by means of Monte Carlo algorithms. Analogously, if we only consider restricted Monte Carlo algorithms, we use the notation $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F, X)$ to denote the *ε-complexity* for the approximation of S (on the class F) by means of restricted Monte Carlo algorithms. If X is fixed, i.e., can not be mistaken, then we may also write $\operatorname{comp}(\varepsilon, F)$ and $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F)$, respectively.

Observe that we have the relation

$$\operatorname{comp}(\varepsilon, F, X) \le \operatorname{comp}^{\mathrm{res}}(\varepsilon, F, X).$$

To illustrate that random bits are in general indeed a severe restriction, we present a computational problem being trivial when it comes to random numbers, but being unsolvable if only random bits are available, cf. Giles et al. [28, Example 14].

Example 2. Let F denote the class of all functions $f\colon [0,1] \to \mathbb{R}$ that are constant except for either finitely many or countably many points from the unit interval, or more general except for a subset of $[0,1]$ having Lebesgue measure zero. The aim is to compute

$$S(f) = \int_{[0,1]} f(\omega)\, \lambda(\mathrm{d}\omega)$$

for $f \in F$. We consider the ε-complexity of this problem. Let $\varepsilon > 0$. Clearly, we have

$$\operatorname{comp}^{\mathrm{res}}(\varepsilon, F) = \infty,$$

since finitely many random bits do allow only algorithms with a fixed finite support. More precisely, an algorithm using no more than p random bits can access at most 2^p fixed points from $[0,1]$ in order to solve the integration problem. Hence, for each restricted algorithm A we can choose a function f that is equal to 0 in all its accessible support points, and strictly larger than ε everywhere else. Clearly, we have $A(f) = 0$ while $S(f) > \varepsilon$. Therefore, each restricted algorithm A violates the condition $\operatorname{error}(A, F) \le \varepsilon$ in the definition of $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F)$. Hence, $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F)$ is unbounded, as claimed.

On the other hand, we have $S(f) = f(\omega)$ with probability one for $\omega \in [0,1]$, and therefore

$$\operatorname{comp}(\varepsilon, F) = 1.$$

IV.2. Direct Simulation and Classical Monte Carlo

Before we turn to the concept of multilevel Monte Carlo algorithms for the approximation of $S(f) = \mathrm{E}(f(X))$, for a better understanding, we first consider the corresponding classical Monte Carlo method, which is, in the context of multilevel algorithms, also known as the concept of singlelevel Monte Carlo algorithms. Classical Monte Carlo algorithms rely on the method of direct simulation, which is explained in the following.

Definition 7. Given $c \in \mathbb{R}$, we consider a real-valued random variable Z with expectation c. Here, Z is called a *basic experiment* for the approximation of c. Furthermore, let $(Z_k)_{k\in\mathbb{N}}$ be an independent sequence of random variables defined on a common probability space, and each sharing the same distribution as Z. The Z_k's are called *(independent) copies* or *(independent) replications* of Z. Then

$$A_n = \frac{1}{n} \cdot \sum_{k=1}^{n} Z_i$$

serves as an approximation of c and A_n is called a *direct simulation* for the approximation of c.

Observe that in the afore definition A_n indeed converges to c, as n tends to infinity, due to the strong law of large numbers.

Now, we introduce the classical Monte Carlo method. We suppose to have a sequence $(X_k)_{k\in\mathbb{N}}$ of approximations with the following three properties:

- Realizations of X_k can be computed with a randomized algorithm, see Definition 3,
- X_k is square-integrable for every $k \in \mathbb{N}$, and
- X_k converges to X, as k tends to infinity, in a suitable way, which we specify later.

Of each approximation X_k we take $n_k \in \mathbb{N}$ independent copies $X_{k,1}, \dots, X_{k,n_k}$, i.e., an i.i.d. sequence with $X_{k,1} \sim X_k$.

Think, e.g., of Euler-Maruyama approximate solutions X_k based on k equidistant steps, see Section II.2.1, in this context, we call X_k an approximation scheme and $X_{k,1}, \dots, X_{k,n_k}$ are called (independent) realizations of X_k.

The direct simulation of $f(X_k)$ for the approximation of $S(f)$ is given by

$$A_{k,n_k}(f) = \frac{1}{n_k} \cdot \sum_{i=1}^{n_k} f(X_{k,i}),$$

which corresponds to the arithmetical mean of the (independent) copies $X_{k,1}, \dots, X_{k,n_k}$. The algorithm A_{k,n_k} is called a classical Monte Carlo algorithm. Observe that $\mathrm{E}(A_{k,n_k}(f)) = \mathrm{E}(f(X_k))$. Assuming square-integrability of each X_k we have

$$\begin{aligned}(\operatorname{error}(A_{k,n_k}, f))^2 &= \big(\operatorname{Bias}(A_{k,n_k}(f))\big)^2 + \operatorname{Var}(A_{k,n_k}) \\ &= \mathrm{E}\big(f(X) - f(X_k)\big)^2 + \frac{1}{n_k} \operatorname{Var}(f(X_k)),\end{aligned}$$

see Remark 21. We make two main observations:

- $\operatorname{Bias}(A_{k,n_k}(f))$ is independent from the replication number n_k for each $k \in \mathbb{N}$, i.e., relies only on the approximation X_k itself, and

- $\mathrm{Var}(A_{k,n_k}(f))$ converges to zero as n_k tends to infinity, and $\mathrm{Var}(f(X_k))$ can be interpreted as a scaling factor.

Clearly, there are two parameters for adjusting the classical Monte Carlo algorithm $A_{k,n_k}(f)$. One is the chosen approximation X_k and the other one is the associated replication number n_k. As the bias of $A_{k,n_k}(f)$ is independent from n_k and X_k only influences the variance of $A_{k,n_k}(f)$ by a constant, the procedure is usually as follows. At first we choose an approximation X_k from the sequence $(X_k)_{k\in\mathbb{N}}$ such that the bias is sufficiently small (for our purposes) and then we choose the replication number n_k such that the statistical error is also sufficiently small (for our purposes).

Observe that working in a worst case setting one is aiming for approximations X_k that lead to a small bias for all $f \in F$.

Definition 8. Given a sequence of approximation schemes $(X_k)_{k\in\mathbb{N}}$. For $f \in F$ we say that $(X_k)_{k\in\mathbb{N}}$ converges to X (with respect to f) with *weak order* or *weak rate* $\alpha > 0$ if

$$\big|\mathrm{E}\big(f(X) - f(X_k)\big)\big| \preccurlyeq \big(\mathrm{E}\big(\mathrm{cost}(X_k, f)\big)\big)^{-\alpha}$$

uniformly in k. The *weak order* or *weak rate of convergence* with respect to the function class F is again defined in the worst case setting, i.e., $(X_k)_{k\in\mathbb{N}}$ converges to X (with respect to F) with *weak order* or *weak rate* $\alpha > 0$ if

$$\sup_{f\in F}\big|\mathrm{E}\big(f(X) - f(X_k)\big)\big| \preccurlyeq \sup_{f\in F}\big(\mathrm{E}\big(\mathrm{cost}(X_k, f)\big)\big)^{-\alpha}$$

uniformly in k.

Observe that $\mathrm{cost}(X_k, f)$ is defined by treating X_k as an algorithm, see Section IV.1.

Example 3. We compute the weak asymptotic relation between the error and the cost of the algorithm A_{k,n_k} on the class F_p for $(X_k)_{k\in\mathbb{N}}$ being the sequence of Euler-Maruyama approximate solutions X_k based on k equidistant time steps, see Section II.2.1. Observe that the cost for one realization $f(X_{k,i})$ is, up to a multiplicative constant, given by k, and therefore

$$\mathrm{cost}(A_{k,n_k}, F_p) \preccurlyeq k \cdot n_k. \tag{60}$$

For convenience and regarding Remark 6, concerning the improved Euler-Maruyama rate for additive noise, we will denote the weak rate of convergence of X_k with respect to the L_p-norm by α. Observe that due to F_p being the class of Lipschitz functionals with bounded Lipschitz constant, the convergence rate carries over to the sequence $(f(X_k))_{k\in\mathbb{N}}$ for all $f \in F_p$, i.e.,

$$\big(\mathrm{Bias}(A_{k,n_k}(F_p))\big)^2 \preccurlyeq \left(\frac{\mathrm{cost}(A_{k,n_k}(F_p))}{n_k}\right)^{-2\cdot\alpha}$$

uniformly in k. Consequently, we have

$$\big(\mathrm{error}(A_{k,n_k}, F_p)\big)^2 \preccurlyeq k^{-2\cdot\alpha} + n_k^{-1}. \tag{61}$$

To determine the parameters k and n_k such that $k^{-2\cdot\alpha}$ and n_k^{-1} obey the same weak asymptotic law, we assume to have some weak asymptotic bound $\varepsilon_k > 0$ on the error of A_{k,n_k}. That means, we claim bias and variance both to be weak asymptotically equivalent to ε_k. We obtain

$$n_k^{-1} \asymp \varepsilon_k^2 \Leftrightarrow n_k \asymp \varepsilon_k^{-2},$$

as well as,

$$k^{-2\cdot\alpha} \asymp \varepsilon_k^2 \Leftrightarrow k \asymp \varepsilon_k^{-1/\alpha}.$$

Note that we ignore that k and n_k need to be natural numbers. This can, e.g., be achieved using the ceiling function. Finally, we obtain for the cost of A_{k,n_k}, see (60),

$$\operatorname{cost}(A_{k,n_k}, F_p) \preccurlyeq k \cdot n_k \asymp \varepsilon_k^{-1/\alpha} \cdot \varepsilon_k^{-2} \asymp \varepsilon_k^{-(2+1/\alpha)}. \tag{62}$$

By Theorem 2, we obtain the weak rate of convergence $\alpha = 1/2$ for $(f(X_k))_{k\in\mathbb{N}}$ for all $f \in F_p$ and therefore

$$\operatorname{cost}(A_{k,n_k}, F_p) \preccurlyeq \big(\operatorname{error}(A_{k,n_k}, F_p)\big)^{-4}. \tag{63}$$

For additive noise, an underlying one dimensional SDE with a one dimensional driving Brownian motion, and a functional f that depends only on the breakpoints of the Euler-Maruyama method, e.g. evaluation in the final time point, Remark 6 yields together with Theorem 3 that we have $\alpha = 1$ and therefore

$$\operatorname{cost}(A_{k,n_k}, f) \preccurlyeq \big(\operatorname{error}(A_{k,n_k}, f)\big)^{-3}.$$

REMARK 22. Note that the dependence between error and cost of the classical Monte Carlo algorithm in Example 3 gets even worse (by a logarithmic term) if we replace F_p by F_∞, as the Euler-Maruyama scheme already has a worse weak rate of convergence on this class of functions, see Theorem 2.

REMARK 23. The estimate (62) does only depend on the weak rate of convergence α of the sequence of estimators $(f(X_k))_{k\in\mathbb{N}}$. As we have seen, there are functionals f for which the rate α increases if we employ a higher order approximation scheme. But we pay a price, namely we need stronger assumptions on the class of functionals F and on the diffusion coefficient b of the SDE under consideration, e.g., the Milstein scheme requires that b is differentiable, and not only globally Lipschitz. For more higher order approximation schemes the condition would get even more restrictive. Let us also mention that there arise (as already for the Milstein scheme) further problems approximating the solution of a system of SDEs driven by a multi-dimensional Brownian motion, in form of iterated integrals with respect to (different) Brownian motions, keyword Lévy areas, which we will not discuss any further. However, we will always have the exponent $2 + 1/\alpha$ in (62), which is essentially bigger than 2, as long as we consider biased Monte Carlo algorithms, like A_{k,n_k}.

Remark 23 raises the question, whether there is another way to improve the relation between the error and the cost of a Monte Carlo algorithm A_{k,n_k}. A classical approach is to change the employed estimator $f(X_k)$ in order to reduce the variance. For an overview on common variance reduction techniques we refer to Müller-Gronbach et al. [49, Chapter 5]. It will turn out later, that multilevel Monte Carlo algorithms can be interpreted as an iterated application of a particular variance reduction technique, the control variate method, which we will briefly describe in the following section. As a motivation, let us already mention that there exists a multilevel Monte Carlo algorithm that either hits the exponent 2 in (62) or misses it by some logarithmic factor, only, depending on some conditions that are specified later.

IV.2.1. Control Variate Method. We consider two square-integrable random variables Z and $\widetilde{Z}$ defined on a common probability space. Furthermore, we assume that the expectation of $\widetilde{Z}$ is either known or can be computed easily, and our aim is the approximation of $\mathrm{E}(Z)$. Since the expectation is linear, for every $c \in \mathbb{R}$, clearly,

$$\widetilde{Z}_c = Z - c \cdot \big(\widetilde{Z} - E\big(\widetilde{Z}\big)\big) \tag{64}$$

serves as a basic experiment for the approximation of $\mathrm{E}(Z)$. The corresponding direct simulation, cf. Definition 7 is given by

$$\widetilde{A}_{n,c} = \frac{1}{n} \cdot \sum_{i=1}^{n} \big(Z_i - c \cdot \big(\widetilde{Z}_i - E\big(\widetilde{Z}\big)\big)\big) = c \cdot \mathrm{E}\big(\widetilde{Z}\big) + \frac{1}{n} \cdot \sum_{i=1}^{n} \big(Z_i - c \cdot \widetilde{Z}_i\big) \tag{65}$$

with $\big(Z_1, \widetilde{Z}_1\big), \ldots, \big(Z_n, \widetilde{Z}_n\big)$ being independent copies of $\big(Z, \widetilde{Z}\big)$. The task within this setting is to choose c and $\widetilde{Z}$ such that the variance of $(Z - c \cdot \widetilde{Z})$ is (significantly) smaller than the variance of Z. In that context, $c \cdot \widetilde{Z}$ is called a *control variate* for Z.

For an optimal choice of c given Z and $\widetilde{Z}$, as well as, for a further discussion of the control variate method, see, e.g., Müller-Gronbach et al. [49, Section 5.2].

Remark 24. There are two straightforward generalizations of the control variate method that we want to discuss. Actually, these generalizations will turn out to be the basic concept for the multilevel Monte Carlo method to be introduced in Section IV.3.

(i) So far we assumed that $\mathrm{E}\big(\widetilde{Z}\big)$ is either known or can be computed easily. Now, we assume it to be more complicated (but $\widetilde{Z}$ still a reasonable control variate for Z). In that case $\mathrm{E}\big(\widetilde{Z}\big)$ may also be approximated via a direct simulation (with basic experiment $\widetilde{Z}$) which is independent from the direct simulation of the difference $(Z - c \cdot \widetilde{Z})$. The corresponding enhanced direct simulation is given by

$$\widetilde{A}_{\widetilde{n},n,c} = \frac{c}{\widetilde{n}} \cdot \sum_{k=1}^{\widetilde{n}} \widetilde{Z}^{(k)} + \frac{1}{n} \cdot \sum_{i=1}^{n} \big(Z_k - c \cdot \widetilde{Z}_k\big)$$

with $\big(Z_1, \widetilde{Z}_1\big), \ldots, \big(Z_n, \widetilde{Z}_n\big)$ as in (65), $\widetilde{Z}^{(1)}, \ldots, \widetilde{Z}^{(\widetilde{n})}$ independent copies of $\widetilde{Z}$, and $\big(\widetilde{Z}^{(1)}, \ldots, \widetilde{Z}^{(\widetilde{n})}\big)$ independent from $\big(\big(Z_1, \widetilde{Z}_1\big), \ldots, \big(Z_n, \widetilde{Z}_n\big)\big)$. Observe that the expectation can be written in terms of

$$\mathrm{E}\big(\widetilde{A}_{\widetilde{n},n,c}\big) = c \cdot \mathrm{E}\big(\widetilde{Z}\big) + \mathrm{E}\big(Z - c \cdot \widetilde{Z}\big). \tag{66}$$

As an example, consider $Z = f(X)$, where X is the solution of an SDE as in (59) and $f \in F$. Moreover, for $k \in \mathbb{N}$, let X_{2^k} denote the Euler-Maruyama approximate solution of X based on 2^k equidistant time steps, and let $\widetilde{Z} = f(X_{2^k})$ be the control variate for Z. In this case, $\mathrm{E}\big(\widetilde{Z}\big)$ is in general unknown, but can (easily) be approximated by means of a direct simulation, cf. Example 3.

(ii) The second generalization is the iterative application of the control variate method. Let $\widetilde{Z}$ be a control variate for Z and $c \in \mathbb{R}$ as in (64). As in (i), we assume that $\mathrm{E}\big(\widetilde{Z}\big)$ in not explicitly known and furthermore, we assume that a 'good' approximation of $\mathrm{E}\big(\widetilde{Z}\big)$ would require some complicated respective time demanding computation. Nevertheless, we suppose that $\widetilde{Z}$ is still a reasonable control variate for Z, in terms of the variance reduction. It stands to reason to use a further control variate $\widehat{Z}$ for $\widetilde{Z}$ such that $\mathrm{E}\big(\widehat{Z}\big)$ is at least easier to approximate then $\mathrm{E}\big(\widetilde{Z}\big)$. Of course, one might ask why we do not use $\widehat{Z}$ as a control variate for Z from the beginning. Indeed the

hope of this iterative method is a further reduction of the computational cost of the approximation of $\mathrm{E}(Z)$. The direct simulation for $\mathrm{E}(\widetilde{Z})$ is given by

$$\widehat{A}_{\widehat{n},\widehat{c}} = \widehat{c} \cdot \mathrm{E}(\widehat{Z}) + \frac{1}{\widehat{n}} \cdot \sum_{k=1}^{\widehat{n}} (\widetilde{Z}_k - \widehat{c} \cdot \widehat{Z}_k).$$

For the approximation of $\mathrm{E}(Z)$ we obtain

$$\widehat{A}_{n,\widehat{n},c,\widehat{c}} = c \cdot \widehat{c} \cdot \mathrm{E}(\widehat{Z}) + \frac{c}{\widehat{n}} \cdot \sum_{k=1}^{\widehat{n}} (\widetilde{Z}_k - \widehat{c} \cdot \widehat{Z}_k) + \frac{1}{n} \cdot \sum_{i=1}^{n} (Z_i - c \cdot \widetilde{Z}_i)$$

with $(\widetilde{Z}_1, \widehat{Z}_1), \ldots, (\widetilde{Z}_{\widehat{n}}, \widehat{Z}_{\widehat{n}})$ independent copies of $(\widetilde{Z}, \widehat{Z})$ and $(Z_1, \widetilde{Z}_1), \ldots, (Z_n, \widetilde{Z}_n)$ independent copies of $(Z, \widetilde{Z})$, as well as, independence of $((\widetilde{Z}_1, \widehat{Z}_1), \ldots, (\widetilde{Z}_{\widehat{n}}, \widehat{Z}_{\widehat{n}}))$ and $((Z_1, \widetilde{Z}_1), \ldots, (Z_n, \widetilde{Z}_n))$, i.e., we employ two independent direct simulations. Observe that the expectation can be written in terms of

$$\mathrm{E}(\widehat{A}_{n,\widehat{n},c,\widehat{c}}) = c \cdot \widehat{c} \cdot \mathrm{E}(\widehat{Z}) + c \cdot \mathrm{E}(\widetilde{Z} - \widehat{c} \cdot \widehat{Z}) + \mathrm{E}(Z - c \cdot \widetilde{Z}) \tag{67}$$

As an example, similar as in (i), we consider Euler-Maruyama approximate solutions X_{2^k} and X_{2^ℓ} based on 2^k and 2^ℓ equidistant time steps, respectively. We set $Z = f(X)$, $\widetilde{Z} = f(X_{2^k})$, and $\widehat{Z} = f(X_{2^\ell})$ with $\ell < k$.

Finally, we observe that both generalizations can be combined, immediately.

Before we turn to multilevel Monte Carlo algorithms, we first introduce asymptotic confidence intervals, which can be used to judge the reliability of empirical estimations, i.e., in particular of the result of a direct simulation.

IV.2.2. Asymptotic Confidence Intervals. The advantage of asymptotic confidence intervals is that they do not need any information on the basic experiment except for the empirical expectation and variance. Also no boundedness assumptions on the basic experiment are needed, as, e.g., in the Hoeffding inequality. For a discussion on confidence intervals and for more details on the topic, see, e.g., Müller-Gronbach et al. [49, Section 3.5].

We introduce asymptotic confidence intervals in the general terminology of the direct simulation, cf. Definition 7. Let Z be a square-integrable random variable with expectation $\mathrm{E}(Z)$ to be approximated, and $(Z_k)_{k\in\mathbb{N}}$ a sequence of independent copies of Z. The empirical expectation, which corresponds to the output of the direct simulation, is denoted by

$$E_n = \frac{1}{n} \cdot \sum_{k=1}^{n} Z_k$$

for $n \in \mathbb{N}$, and for $n \geq 2$ the (unbiased) empirical variance V_n of E_n is given by

$$V_n = \frac{1}{n-1} \cdot \sum_{k=1}^{n} (Z_k - E_n)^2. \tag{68}$$

To obtain a symmetric confidence interval $I_n = [E_n - \varepsilon, E_n + \varepsilon]$ with $\varepsilon > 0$ based on E_n and V_n only, we can employ the central limit theorem stating that for sufficiently large n, the distribution of E_n is approximately Gaussian, see, e.g., Billingsley [5, Section 27, Theorem 27.1].

The following result is well-known, see, e.g., Müller-Gronbach et al. [49, Theorem 3.18].

THEOREM 8. *Let Z be a square-integrable non-trivial random variable, i.e.,* $\mathrm{Var}(Z) > 0$, *and $\delta \in]0,1[$. Moreover, let Φ^{-1} denote the inverse distribution function of the standard normal distribution. Then $L_n = \Phi^{-1}(1-\delta/2) \cdot (V_n/n)^{1/2}$ defines an asymptotic confidence interval*

$$I_n = [E_n - L_n, E_n + L_n]$$

to the confidence level $1-\delta$ for $E(Z)$, i.e.,

$$\lim_{n\to\infty} P\big(\{E(Z) \in I_n\}\big) = 1 - \delta.$$

REMARK 25. The value of $\Phi^{-1}(1-\delta/2)$ can either be taken from a look-up table or it can be computed by means of a numerical routine. The particular asymptotic confidence interval with confidence level 0.95, i.e., $\delta = 0.05$, as we will use it throughout Chapter V is given by

$$I_n = \big[E_n - 1.96 \cdot (V_n/n)^{1/2}, E_n + 1.96 \cdot (V_n/n)^{1/2}\big]$$

for $n \geq 2$.

IV.3. Multilevel Monte Carlo (MLMC) Methods

For the multilevel Monte Carlo method we take into account a whole hierarchy of approximation schemes $(X_k)_{k\in\mathbb{N}_0}$ with weak convergence for all $f \in F$, i.e., there exists a positive constant c such that

$$\sup_{f\in F}\big|\mathrm{E}(f(X) - f(X_k))\big| \leq c \cdot \sup_{f\in F}\big|\mathrm{E}(f(X) - f(X_\ell))\big|$$

for $k \geq \ell$, and with monotonically increasing cost for the computation of a single copy of X_k, in k, and consequently for $f(X_k)$. For $L \in \mathbb{N}_0$, in terms of the expectation, Remark 24 specifically (66) and (67), clearly yield the particular approach

$$\mathrm{E}\big(f(X_L)\big) = \mathrm{E}\big(f(X_0)\big) + \sum_{\ell=1}^{L} \mathrm{E}\big(f(X_\ell) - f(X_{\ell-1})\big) \tag{69}$$

where each expectation on the right hand side is independently approximated by means of a direct simulation A_{ℓ,n_ℓ} using $n_\ell \in \mathbb{N}$ independent copies $(B_{\ell,i})_{i=1,\dots,n_\ell}$ of the basic experiments

$$B_\ell = \begin{cases} f(X_0), & \text{if } \ell = 0, \\ f(X_\ell) - f(X_{\ell-1}), & \text{if } \ell \geq 1. \end{cases}$$

Note that for each control variate the constant $c \in \mathbb{R}$ in (64) is à priori chosen to be one. Property (69) is also called *telescoping sum property (of the expectation)*. The corresponding multilevel algorithm reads as

$$A_{L,n}(f) = \sum_{\ell=0}^{L} A_{\ell,n_\ell} = \sum_{\ell=0}^{L} \frac{1}{n_\ell} \cdot \sum_{i=1}^{n_\ell} B_{\ell,i} \tag{70}$$

with $n = (n_0, \dots, n_L)$. Concerning the error of $A_{L,n}(f)$, observe that due to the telescoping sum property (69) we have

$$\mathrm{Bias}\big(A_{L,n}(f)\big) = \mathrm{E}\big(f(X) - A_{L,n}(f)\big) = \mathrm{E}\big(f(X) - f(X_L)\big), \tag{71}$$

i.e., the bias of $A_{L,n}$ only depends on the most accurate approximation scheme under consideration, and Bienaymé's formula yields

$$\operatorname{Var}(A_{L,n}(f)) = \sum_{\ell=0}^{L} \frac{1}{n_\ell} \cdot \operatorname{Var}(B_\ell). \tag{72}$$

So, the crucial point for this algorithm is that in each direct simulation A_{ℓ,n_ℓ} with $\ell \geq 1$ the approximation schemes X_ℓ and $X_{\ell-1}$ need to be coupled in a way that the variance of $\big(f(X_\ell) - f(X_{\ell-1})\big)$ is sufficiently small, indeed it has to decrease uniformly in $\ell \in \mathbb{N}$. How such a coupling, based on the underlying sample paths of the driving Brownian motion W, can be achieved in the classical multilevel Monte Carlo setting is described in detail in Section IV.3.1, and for the random bit approximation setting two coupling methods are presented in Section IV.4.1 and Section IV.4.2.

For a better understanding of the necessity of an appropriate coupling, we suppose that in each direct simulation A_{ℓ,n_ℓ} with $\ell \geq 1$, the realizations of $f(X_\ell)$ and $f(X_{\ell-1})$ are independent. Then the variances of the basic experiments B_ℓ satisfy

$$\operatorname{Var}(B_\ell) = \operatorname{Var}(f(X_\ell)) + \operatorname{Var}(f(X_{\ell-1})),$$

and therefore the variance of the multilevel algorithm $A_{L,n}$ is given by

$$\operatorname{Var}(A_{L,n}) = \frac{1}{n_0} \cdot \sum_{i=1}^{n_0} \operatorname{Var}(f(X_0)) + \sum_{\ell=1}^{L} \frac{1}{n_\ell} \cdot \operatorname{Var}(f(X_\ell)) + \operatorname{Var}(f(X_{\ell-1})).$$

As a consequence $A_{L,n}$ is inferior to the classical Monte Carlo algorithm

$$A_{L,\widetilde{n}_L} = \frac{1}{\widetilde{n}_L} \cdot \sum_{i=1}^{\widetilde{n}_L} f(X_{L,i})$$

with $X_{L,1}, \dots, X_{L,\widetilde{n}_L}$ independent copies of X_L. Indeed, let $\widetilde{n}_L = n_L$. Since $\mathrm{E}(A_{L,n_L}) = \mathrm{E}(A_{L,n})$ we have

$$\operatorname{Bias}(A_{L,n_L}(F)) = \operatorname{Bias}(A_{L,n}(F)),$$

while

$$\operatorname{Var}(A_{L,n_L}(F)) \leq \operatorname{Var}(A_{L,n}(F)).$$

Altogether, this means

$$\operatorname{error}(A_{L,n_L}, F) \leq \operatorname{error}(A_{L,n}, F) \qquad \text{with} \qquad \operatorname{cost}(A_{L,n_L}, F) \leq \operatorname{cost}(A_{L,n}, F).$$

IV.3.1. MLMC Euler-Maruyama. From now on, we will investigate a particular multilevel Monte Carlo method, namely the method based on a, later to be specified, particular subsequence of the Euler-Maruyama approximate solutions $(X_m)_{m\in\mathbb{N}}$ with X_m using m equidistant time steps, see Section II.2.1. This method is also called *multilevel Monte Carlo Euler-Maruyama method*, and we use the shorthands *MLMC Euler method* as well as *multilevel Euler method*. Observe that, by Theorem 2, we have the claimed weak convergence of $(X_m)_{m\in\mathbb{N}_0}$ to X as m tends to infinity, for all f from the Lipschitz classes $F = F_p$ and $F = F_\infty$, respectively.

As already mentioned the coupling of the different employed approximation schemes, here of Euler-Maruyama approximate solutions based on a different number of equidistant time steps, is essential for the reduction of the variance of the estimator. Hence, before we turn to the construction of this particular multilevel method, we describe the standard coupling

of different Euler-Maruyama approximate solutions, that will lead us to the subsequence $(X_{2^k})_{k\in\mathbb{N}_0}$ of $(X_m)_{m\in\mathbb{N}}$. To this end, we recall the definition of the Euler-Maruyama scheme with breakpoints $t_k = t_{k,m} = k/m$ for $k = 0,\dots,m$ from Section II.2.1, namely

$$\begin{aligned} X_m(t_{0,m}) &= x_0,\\ X_m(t_{k,m}) &= X_m(t_{k-1,m}) + m^{-1}\cdot a\big(X_m(t_{k-1,m})\big) + b\big(X_m(t_{k-1,m})\big)\cdot V_{k,m} \end{aligned}$$

for $k = 1,\dots,m$, with Brownian increments

$$V_k = V_{k,m} = W(t_k) - W(t_{k-1}).$$

Observe that this is a pathwise approximation where each realization relies on the increments of a single sample path of the driving Brownian motion W. Since the randomness only enters X_m via W, as the SDE itself, the coupling clearly has to rely on the sample paths of W. For an even number $m \in \mathbb{N}$ of time steps we define the coupled Euler-Maruyama approximate solution $\widetilde{X}_{m/2}$ via

$$\begin{aligned} \widetilde{X}_{m/2}(t_{0,m/2}) &= x_0,\\ \widetilde{X}_{m/2}(t_{k,m/2}) &= \widetilde{X}_{m/2}(t_{k-1,m/2}) + (m/2)^{-1}\cdot a\big(\widetilde{X}_{m/2}(t_{k-1,m/2})\big)\\ &\qquad + b\big(\widetilde{X}_{m/2}(t_{k-1,m/2})\big)\cdot \widetilde{V}_{k,m/2} \end{aligned}$$

for $k = 1,\dots,m/2$ and with Brownian increments

$$\widetilde{V}_k = \widetilde{V}_{k,m/2} = V_{2\cdot k,m} + V_{2\cdot k-1,m} = W(t_{k,m/2}) - W(t_{k-1,m/2}). \tag{73}$$

That is, for the coupling we always assume that the finer approximation uses exactly twice as much time steps as the coupled coarser approximation. We stress that $\widetilde{V}_{k,m/2} = V_{k,m/2}$, implying that $\widetilde{X}_{m/2}$ and $X_{m/2}$ share the same distribution, which is convenient in the analysis of multilevel algorithms, cf. the telescoping sum property (69). Recall that continuous extensions of the two schemes X_m and $\widetilde{X}_{m/2}$ to the unit interval are obtained via piecewise linear interpolation of $X_m(t_{0,m}),\dots,X_m(t_{m,m})$ and $\widetilde{X}_{m/2}(t_{0,m/2}),\dots,\widetilde{X}_{m/2}(t_{m/2,m/2})$ on the subintervals $]t_{k-1,m}, t_{k,m}[$ and $]t_{k-1,m/2}, t_{k,m/2}[$, respectively, see Section II.2.1.

With the coupling at hand, we can give a general description of the multilevel Euler algorithm, according to (70), that will also be applicable for the random bit multilevel Euler algorithms in Section IV.5. Let $L \in \mathbb{N}$ be the maximum level that is used by the multilevel Euler method, i.e., the algorithm involves at least two levels. On every level $\ell = 1,\dots,L$ the algorithm involves a fine approximation X_{2^ℓ} and a coarse approximation $\widetilde{X}_{2^{\ell-1}}$, as defined above. On level $\ell = 0$ we only consider the fine approximation X_{2^0}. Finally, let $N = (N_0,\dots,N_L) \in \mathbb{N}^{L+1}$ be the vector of replication numbers on the levels $\ell = 0,\dots,L$. For notational convenience we set $\widetilde{X}_{2^{-1}} = 0$.

To define the multilevel method we consider an independent family of random elements $\big(X_{2^\ell,i}, \widetilde{X}_{2^{\ell-1},i}\big)$ with $\ell = 0,\dots,L$ and $i = 1,\dots,N_\ell$ such that

$$\big(X_{2^\ell,i}, \widetilde{X}_{2^{\ell-1},i}\big) \overset{\mathrm{d}}{=} \big(X_{2^\ell}, \widetilde{X}_{2^{\ell-1}}\big)$$

for all $\ell = 0,\dots,L$ and $i = 1,\dots,N_\ell$. For $f \in F$, the multilevel method finally reads as

$$A_{L,N}(f) = \frac{1}{N_0}\cdot\sum_{i=1}^{N_0} f\big(X_{2^0,i}\big) + \sum_{\ell=1}^{L}\frac{1}{N_\ell}\cdot\sum_{i=1}^{N_\ell}\big(f\big(X_{2^\ell,i}\big) - f\big(\widetilde{X}_{2^{\ell-1},i}\big)\big). \tag{74}$$

Observe that in terms of (70), $\big(f\big(X_{2^\ell,i}\big)-f\big(\widetilde{X}_{2^{\ell-1},i}\big)\big)_{i=1,\dots,N_\ell}$ corresponds to N_ℓ independent realizations of the basic experiment $f\big(X_{2^\ell}\big)-f\big(\widetilde{X}_{2^{\ell-1}}\big)$.

Before we turn to the analysis of $A_{L,N}$, we estimate the cost of $A_{L,N}$. For that purpose we consider the cost of one realization of $f\big(X_{2^\ell}\big)-f\big(\widetilde{X}_{2^{\ell-1}}\big)$. At first we observe that the number of arithmetic operations and evaluations of the coefficients (each having unit cost) in the computation of one realization of X_{2^ℓ} is, up to a multiplicative constant, bounded by 2^ℓ and yields a natural bound for $\widetilde{X}_{2^{\ell-1}}$, as well. Furthermore, we recall that the cost for the evaluation of f is linear in the number of breakpoints of X_{2^ℓ} and $\widetilde{X}_{2^{\ell-1}}$, namely, $2^\ell+1$ and $2^{\ell-1}+1$, respectively, i.e., we again have the cost bound 2^ℓ, up to a multiplicative constant, which is independent from the particular choice of $f\in F$. Finally, one realization of X_{2^ℓ} and hence also the coupled realization of $\widetilde{X}_{2^{\ell-1}}$ involves 2^ℓ Brownian increments. For a d-dimensional driving Brownian motion this requires $d\cdot 2^\ell$ calls to the random number generator. Altogether, we conclude that the cost of one realization of $f\big(X_{2^\ell}\big)-f\big(\widetilde{X}_{2^{\ell-1}}\big)$ is, up to a multiplicative constant, dominated by the number of calls to the random number generator. For the whole multilevel algorithm $A_{L,N}$ this yields

$$\operatorname{cost}\big(A_{L,N},f\big)\preccurlyeq d\cdot\sum_{\ell=0}^{L}N_\ell\cdot 2^\ell$$

and since the constant hidden in $\preccurlyeq$ is independent from the choice of $f\in F$, we obtain

$$\operatorname{cost}\big(A_{L,N},F\big)\preccurlyeq d\cdot\sum_{\ell=0}^{L}N_\ell\cdot 2^\ell. \tag{75}$$

For $\varepsilon\in\,]0,1/2[$ we consider the algorithm

$$A_{\varepsilon,F}=A_{L(\varepsilon,F),N(\varepsilon,F)}$$

with maximal level

$$L=L(\varepsilon,F)=\begin{cases}\big\lceil\log_2(\varepsilon^{-2})\big\rceil, & \text{if } F=F_p,\\ \big\lceil\log_2(\varepsilon^{-2})+\log_2\big(\log_2(\varepsilon^{-2})\big)\big\rceil, & \text{if } F=F_\infty\end{cases} \tag{76}$$

and with replication numbers

$$N_\ell=N_\ell(\varepsilon,F)=\begin{cases}\big\lceil(L+1)\cdot 2^{-\ell}\cdot\varepsilon^{-2}\big\rceil, & \text{if } F=F_p,\\ \big\lceil(L+1)\cdot 2^{-\ell}\cdot\max(\ell,1)\cdot\varepsilon^{-2}\big\rceil, & \text{if } F=F_\infty\end{cases} \tag{77}$$

for $\ell=0,\dots,L$.

The following result is known, see, e.g., Creutzig et al. [12, Remark 8]. For convenience of the reader and due to its importance in the random bit approximation to the multilevel Euler method in Section IV.5 we present the proof.

THEOREM 9. *Let $F=F_\infty$ or $F=F_p$ with $1\le p<\infty$. Then there exists a positive constant c such that the multilevel Euler method $A_{\varepsilon,F}$ satisfies*

$$\operatorname{error}\big(A_{\varepsilon,F},F\big)\le c\cdot\varepsilon$$

and

$$\operatorname{cost}\big(A_{\varepsilon,F},F\big)\le c\cdot\varepsilon^{-2}\cdot\begin{cases}\big(\ln(\varepsilon^{-1})\big)^2, & \text{if } F=F_p,\\ \big(\ln(\varepsilon^{-1})\big)^3, & \text{if } F=F_\infty\end{cases}$$

for every $\varepsilon\in\,]0,1/2[$.

Here, we do not use the notation $\preccurlyeq$ since it is more convenient to use constants in the proof of Theorem 9, which is presented below.

COROLLARY 4. *In terms of the ε-complexity* $\operatorname{comp}(\varepsilon, F)$*, see Definition 6, Theorem 9 implies an upper bound that is of the same order as* $\operatorname{cost}(A_{\varepsilon,F}, F)$*, i.e., there exists a positive constant c such that*

$$\operatorname{comp}\big(\varepsilon, F\big) \leq c \cdot \varepsilon^{-2} \cdot \begin{cases} \big(\ln(\varepsilon^{-1})\big)^2, & \text{if } F = F_p, \\ \big(\ln(\varepsilon^{-1})\big)^3, & \text{if } F = F_\infty. \end{cases}$$

REMARK 26. Before we turn to the proof of Theorem 9 we first discuss the results therein, and in Corollary 4. Of course, one is interested in classifying the upper bound on the ε-complexity in Corollary 4, i.e., one is interested in a lower bound on $\operatorname{comp}(\varepsilon, F)$, too. Hence, in the following, we present lower bounds derived in Creutzig et al. [12] under a slightly stronger smoothness assumption as well as a non-degeneracy assumption on the diffusion coefficient b of the SDE, see (59), which in particular excludes pathological cases yielding a deterministic solution. In Creutzig et al. [12] the authors establish lower bounds for $L_p([0,1])$ as well as for $C([0,1])$, i.e., for our function classes $F = F_p$ and $F = F_\infty$, that are only based on the number of evaluations of the functional f from F and the cost associated to each evaluation. Hence, these lower bounds, in particular, serve as lower bounds for the ε-complexity in our setting.

For sake of completeness and convenience of the reader, let us mention that in Creutzig et al. [12] the authors do not consider the ε-complexity directly, but the n-th minimal error, which is defined by

$$\operatorname{error}_n(F, X) = \inf\big\{\operatorname{error}(A, F, X) \colon A \text{ is a Monte Carlo algorithm}, \operatorname{cost}(A, F) \leq n\big\}.$$

The relation to the ε-complexity is as follows, let α and β be positive constants, then

$$\operatorname{error}_n(F, X) \preccurlyeq n^{-\alpha} \cdot \ln(n)^\beta$$

corresponds to

$$\operatorname{comp}(\varepsilon, F, X) \preccurlyeq \varepsilon^{-1/\alpha} \cdot \big(\ln(\varepsilon^{-1})\big)^{\beta/\alpha}.$$

The analogous relations hold for $\succcurlyeq$ and $\asymp$, respectively.

In Creutzig et al. [12] the authors actually consider three different cost models, two of which are relevant to us. Namely, variable subspace sampling and fixed subspace sampling. For the latter, f may be evaluated on a finite-dimensional subspace of $C([0,1])$ or $L_p([0,1])$, respectively. This corresponds to the classical Monte Carlo Euler approach, where we consider one fixed approximation scheme X_m based on m equidistant time steps, i.e., we consider the finite-dimensional subspace consisting of piecewise linear functions with breakpoints $t_k = k/m$. Likewise, variable subspace sampling allows an increasing sequence of finite-dimensional subspaces. Following the reasoning for fixed subspace sampling, this setting corresponds to the multilevel Euler setting using the hierarchy of 2^ℓ-dimensional subspaces consisting of piecewise linear functions with breakpoints $t_k = k/2^\ell$, for levels $\ell = 0, \dots, L$, respectively.

Finally, Theorem 11 and Theorem 12 from Creutzig et al. [12] yield the following lower bounds. For fixed subspace sampling we have

$$\varepsilon^{-4} \cdot \big(\ln(\varepsilon^{-1})\big)^3 \preccurlyeq \operatorname{comp}(\varepsilon, F_\infty, X)$$

and

$$\lim_{\varepsilon\to 0} \operatorname{comp}(\varepsilon, F_p, X)\cdot \varepsilon^4 \cdot \big(\ln(\varepsilon^{-1})\big)^{-3} > 0.$$

As a consequence the upper bound on the ε-complexity of the classical Monte Carlo Euler method implied by (63) is sharp, up to logarithmic factors.

For variable subspace sampling we have

$$\varepsilon^{-2} \preccurlyeq \operatorname{comp}(\varepsilon, F, X)$$

for $F = F_p$ and $F = F_\infty$. That is the upper bounds on the ε-complexity in Corollary 4 are sharp, up to logarithmic factors.

Proof of Theorem 9. We present the proof for $F = F_\infty$. Regarding Remark 21, we show that both, squared bias and variance of $A_{\varepsilon,F}$ are bounded by ε^2, up to a multiplicative constant.

By Theorem 2 there exists a constant $c_1 > 0$ such that

$$\mathrm{E}\big\|X - X_{2^\ell}\big\|_{\sup}^2 \le c_1 \cdot 2^{-\ell}\cdot \max(\ell, 1).$$

for every $\ell \ge 1$ and $\mathrm{E}\,\|X_{2^0}\|_{\sup}^2 \le c_1$. Hence we get using (71) and Jensen's inequality

$$\begin{aligned}\big(\operatorname{Bias}\big(A_{L,N}(f)\big)\big)^2 &= \big(\mathrm{E}\big(f(X) - f(X_{2^L})\big)\big)^2\\ &\le \big(\mathrm{E}\,\|X - X_{2^L}\|_{\sup}\big)^2 \le \mathrm{E}\,\|X - X_{2^L}\|_{\sup}^2 \le c_1\cdot 2^{-L}\cdot L\end{aligned}$$

for all L and N, and for every $f \in F_\infty$. For the particular choice $L = L(\varepsilon, F)$ we immediately get

$$\big(\operatorname{Bias}\big(A_{\varepsilon,F}(F)\big)\big)^2 \le 2\cdot c_1\cdot \varepsilon^2,$$

i.e., the claimed upper bound of order ε^2. For the variance we get, using (72)

$$\begin{aligned}\operatorname{Var}\big(A_{L,N}(f)\big) &\le \sum_{\ell=0}^{L} \frac{1}{N_\ell}\,\mathrm{E}\big\|f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})\big\|_{\sup}^2 \le \sum_{\ell=0}^{L}\frac{1}{N_\ell}\,\mathrm{E}\big\|X_{2^\ell} - \widetilde{X}_{2^{\ell-1}}\big\|_{\sup}^2\\ &\le \frac{c_1}{N_0} + 2\cdot\sum_{\ell=1}^{L}\Big(\mathrm{E}\big\|X_{2^\ell} - X\big\|_{\sup}^2 + \mathrm{E}\big\|\widetilde{X}_{2^{\ell-1}} - X\big\|_{\sup}^2\Big)\\ &\le 6\cdot c_1\cdot\sum_{\ell=0}^{L} 2^{-\ell}\cdot\max(\ell,1)\cdot N_\ell^{-1}\end{aligned}$$

for all L and N and for every $f \in F_\infty$. For the particular choice $N = N(\varepsilon, F)$ we immediately get

$$\operatorname{Var}\big(A_{L,N}(F)\big) \le 6\cdot c_1\cdot\varepsilon^2,$$

i.e., the claimed upper bound of order ε^2, too.

To derive the cost bound for $A_{\varepsilon,F}$ it remains to observe that there exists a constant $c_2 > 0$ such that

$$\begin{aligned}\sum_{\ell=0}^{L} N_\ell(\varepsilon, F)\cdot 2^\ell &= \sum_{\ell=0}^{L}(L+1)\cdot\max(\ell,1)\cdot\varepsilon^{-2}\\ &\le c_2\cdot\varepsilon^{-2}\cdot L^3 \le 2\cdot c_2\cdot\varepsilon^{-2}\cdot\big(\log_2(\varepsilon^{-1})\big)^3\end{aligned} \tag{78}$$

for all ε.

The proof for $F = F_p$ follows step by step the proof for $F = F_\infty$, using the corresponding rate of convergence, with respect to the L_p-norm, from Theorem 2. ■

REMARK 27. Let us point to two weaknesses of the algorithm $A_{\varepsilon,F}$. One weakness is that the à priori choice of $L(\varepsilon, F)$ and $N(\varepsilon, F)$ is based on the upper bound on the order of convergence of the Euler-Maruyama approximate solutions $(X_k)_{k\in\mathbb{N}}$ with respect to the norm associated to F. That choice is indeed suboptimal if the order of convergence of $(X_k)_k$ is higher. Unfortunately, the order of convergence of $(X_k)_k$ is usually not known.

A further weakness of $A_{\varepsilon,F}$ is that in a practical application we would like to choose L and N such that the error of $A_{\varepsilon,F}$ is smaller than a given error bound $\varepsilon > 0$. Indeed, we want to minimize $\mathrm{cost}(A_{\varepsilon,F}, f)$ subject to the constraint $\mathrm{error}(A_{\varepsilon,F}, f) \leq \varepsilon$. Observe that this minimization problem implicitly requires knowledge on the convergence of $(X_k)_k$ as well.

IV.3.2. Bias and Variance Estimation. In order to solve the minimization problem from Remark 27 we recall that we have the decomposition of the squared stochastic L_2-error into the squared bias and the variance of $A_{\varepsilon,F}(f)$

$$(\mathrm{error}(A_{\varepsilon,F}, f))^2 = \mathrm{Bias}(A_{\varepsilon,F}(f))^2 + \mathrm{Var}(A_{\varepsilon,F}(f)),$$

see Remark 21, i.e., $\mathrm{error}(A_{\varepsilon,F}, f)$ is regulated by $L = L(\varepsilon, F)$ and $N = N(\varepsilon, F)$. Moreover, $\mathrm{cost}(A_{\varepsilon,F}, f)$ depends only on L and N, too, see (75). Hence, to solve the minimization problem, i.e. an optimal choice of L and N, we do need to know the strong asymptotics of the expectation and the variance of $(f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}}))_{\ell\in N}$. Unfortunately the (strong) asymptotics are in general not known and hence need to be estimated. Indeed we will use bias and variance approximations on a small number of levels L' based on which we estimate the strong asymptotics, respectively. Here, we suppose a specific structure of the weak asymptotics, which is based on the upper bound from Theorem 2 and fortified by the numerical experiments in Chapter V. In that sense, our estimates of the strong asymptotics are indeed estimates of an upper bound of the same, respectively. Note that, relying on upper bounds we will rather overestimate than underestimate the bias and the variance on each level, resulting in a, hopefully only, slightly higher computational cost on the one hand and a smaller error of the multilevel Monte Carlo algorithm. Moreover, the adaptive randomized algorithm which we are going to consider, see Section IV.3.3, is supposed to have an error smaller than some given error bound. Hence overestimations of bias and variance are even leading to a more reliable algorithm. Furthermore, the numerical experiments in Chapter V, for which we will compare the input error bound to the error of the algorithm's output, indicate that there is no substantial overestimate of the bias and variance on the different levels.

However, we have to refine the choices of L and N to the particular functional f, i.e., $L = L(\varepsilon, f)$ and $N = N(\varepsilon, f)$. The corresponding multilevel Euler algorithm will then, for the moment, be denoted by A_ε, to indicate that it does no longer depend on the upper bounds for the convergence of the Euler-Maruyama scheme from Theorem 2, as does $A_{\varepsilon,F}$. Consequently, in the following we show how to estimate

$$\big(\mathrm{Bias}_\ell(f)\big)_{\ell\in\mathbb{N}} = \big(\mathrm{E}\big(f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})\big)\big)_{\ell\in\mathbb{N}} = \big(\mathrm{E}\big(f(X_{2^\ell}) - f(X_{2^{\ell-1}})\big)\big)_{\ell\in\mathbb{N}}$$

and

$$\big(\mathrm{Var}_\ell(f)\big)_{\ell\in\mathbb{N}} = \big(\mathrm{Var}\big(f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})\big)\big)_{\ell\in\mathbb{N}} = \big(\mathrm{Var}\big(f(X_{2^\ell}) - f(X_{2^{\ell-1}})\big)\big)_{\ell\in\mathbb{N}}$$

based on approximations of the respective quantities for the first $L' \in \mathbb{N}$ members of the sequences, respectively.

In order to have *good* upper bounds on the strong asymptotics of $(\mathrm{Bias}_\ell(f))$ and $(\mathrm{Var}_\ell(f))$ we, clearly, need *good* approximations of $(\mathrm{Bias}_\ell(f))_{\ell=1,\dots,L'}$ and $(\mathrm{Var}_\ell(f))_{\ell=1,\dots,L'}$, respectively. These approximations will be based on direct simulations of the basic experiments $(f(X_{2^\ell}) - f(X_{2^{\ell-1}}))_{\ell=1,\dots,L'}$, i.e., $\mathrm{Bias}_\ell(f)$ is approximated by the empirical expectation and $\mathrm{Var}_\ell(f)$ by the empirical variance of the corresponding basic experiment. One way to determine the quality and therefore the reliability of such approximations is to consider asymptotic confidence intervals to some confidence level $1-\delta$, with δ close to 0, as introduced in Section IV.2.2.

For the moment we only consider the minimization problem for $f \in F_p$. We follow the approach that is, e.g., used in Giles [25] in terms of an adaptive multilevel Euler algorithm, see Section IV.3.3. We proceed in two stages. To this end, we decompose the upper bound ε on the root-mean-squared error into $\varepsilon^2 = \varepsilon_B^2 + \varepsilon_V^2$, where ε_B denotes the maximal allowed error for the bias, i.e.,

$$\big|\mathrm{Bias}\big(A_\varepsilon(f)\big)\big|^2 = \big(\mathrm{E}\big(f(X) - f(X_{2^L})\big)\big)^2 \le \varepsilon_B^2, \tag{79}$$

and ε_V^2 denotes the maximal allowed error for the variance, i.e.,

$$\mathrm{Var}\big(A_\varepsilon(f)\big) \le \varepsilon_V^2.$$

For the bias we observe that in practical applications $f(X)$ is usually not known. Otherwise, we could immediately sample from $f(X)$ instead of approximating X first. Hence we need to estimate $\mathrm{Bias}(A_\varepsilon(f))$. For notational convenience, we set $\mathrm{Bias}_0(f) = \mathrm{E}(f(X_{2^0}))$. Due to the upper bound on the strong convergence of $(X_{2^\ell})_{\ell\in\mathbb{N}}$ from Theorem 2, and the Lipschitz continuity of f we obtain weak convergence of $(X_{2^\ell})_{\ell\in\mathbb{N}}$ with respect to f and hence

$$\mathrm{E}(f(X)) = \lim_{m\to\infty} \mathrm{E}(f(X_{2^m})) = \lim_{m\to\infty} \sum_{\ell=0}^{m} \mathrm{Bias}_\ell(f) = \sum_{\ell=0}^{\infty} \mathrm{Bias}_\ell(f),$$

and therefore by the telescoping sum property (69)

$$\mathrm{Bias}\big(A_\varepsilon(f)\big) = \sum_{\ell=0}^{\infty} \big(\mathrm{Bias}_\ell(f)\big) - \mathrm{E}\big(f(X_{2^L})\big) = \sum_{\ell=L+1}^{\infty} \mathrm{Bias}_\ell(f).$$

Hypothesis 1. Motivated by the upper bound from Theorem 2, and suitable in the numerical experiments in Chapter V, we suppose $(\mathrm{Bias}_\ell(f))_{\ell\in\mathbb{N}}$ to have the law

$$\big|\mathrm{Bias}_\ell(f)\big| \le c_B \cdot \big(2^{-\ell}\big)^\alpha = c_B \cdot \big(2^{-\alpha}\big)^\ell$$

for some $\alpha > 0$, i.e., the sequence $(|\mathrm{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ is bounded from above by a geometrical sequence with initial value c_B and quotient $2^{-\alpha}$.

We add that the constant $c_B > 0$ may actually depend on $f \in F_p$, and recall that, here, we are interested in an upper bound on the strong asymptotic, i.e., constants do matter. Moreover, note that, for the SDE under consideration, see (59), Theorem 2 ensures $\alpha \ge 1/2$.

In order to estimate $\mathrm{Bias}(A_\varepsilon(f))$ we observe that we have $2^{-\alpha} \in]0,1[$. Hence the well-known formula for the remainder of a geometric series is applicable and we get

$$\mathrm{Bias}\big(A_\varepsilon(f)\big) \le \sum_{\ell=L+1}^{\infty} \big|\mathrm{Bias}_\ell(f)\big| \le c_B \cdot \sum_{L+1}^{\infty} \big(2^{-\alpha}\big)^\ell = c_B \cdot \frac{(2^{-\alpha})^{L+1}}{1-2^{-\alpha}}.$$

Consequently, condition (79), for the choice of the highest level L, is implied by

$$\frac{c_B \cdot \left(2^{-\alpha}\right)^{L+1}}{1-2^{-\alpha}} \leq \varepsilon_B. \tag{80}$$

Solving the inequality for L yields

$$L \geq \log_{2^{-\alpha}}\left(\varepsilon_B \cdot (1-2^{-\alpha})\right) - \log_{2^{-\alpha}}(c_B) - 1.$$

As L needs to be an integer it is natural to simply apply the ceiling function, i.e., we choose

$$L = \left\lceil \frac{\ln\left(\varepsilon_B \cdot (1-2^{-\alpha})\right) - \ln(c_B)}{\ln(2^{-\alpha})} - 1 \right\rceil. \tag{81}$$

REMARK 28. It remains to estimate the exponential decay α of $(|\operatorname{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ in terms of the level ℓ as well as the constant c_B. This can be done by means of a classical log-log-linear-regression. To this end, for a given $L' \in \mathbb{N}$, we plot the logarithm of the (via direct simulation) approximated values of $|\operatorname{Bias}_\ell(f)|$, $\ell = 1, \ldots, L'$, against the logarithm of the cost associated to one realization of the corresponding basic experiment $f(X_{2^\ell}) - f(X_{2^{\ell-1}})$. Recall that for Euler-Maruyama approximate solutions $(X_{2^\ell})_{\ell\in\mathbb{N}}$ the cost of $f(X_{2^\ell})$ is given by 2^ℓ, up to a multiplicative constant. Hence it is convenient to choose the logarithm with basis 2 and to assume the cost of $(f(X_{2^\ell}) - f(X_{2^{\ell-1}}))$ to be exactly 2^ℓ, i.e., we are aiming for a log-linear regression of the approximations of the absolute values of $\operatorname{Bias}_\ell(f)$ plotted against the corresponding levels $\ell = 1, \ldots, L'$. Hypothesis 1 suggests that for sufficiently good approximations of $(|\operatorname{Bias}_\ell(f)|)_{\ell=1,\ldots,L'}$ their logarithmic values should be bounded by a straight line. The corresponding parameters are estimated by a linear least-square fit, as described in Section A.2 in the Appendix, with $(|\operatorname{Bias}_\ell(f)|)_{\ell=1,\ldots,L'}$ as response variable and with design matrix $D = [D_1, D_2]$ where $D_1 = [1, \ldots, L']^T$ and $D_2 = [1, \ldots, 1]^T$. Observe that the decay rate α corresponds to the slope of this fit and c_B to the exponential of the intercept of the fit and the y-axis.

Having determined the highest level L we can turn to the numbers $N_\ell \in N$ of replications on each level $\ell = 0, \ldots, L$. For notational convenience, analogously as for $\operatorname{Bias}_\ell(f)$, we set $\operatorname{Var}_0(f) = \operatorname{Var}(f(X_{2^0}))$. Moreover, we define

$$C_\ell = C_\ell(f) = \begin{cases} \operatorname{cost}\left(f(X_{2^0})\right), & \text{if } \ell = 0, \\ \operatorname{cost}\left(f(X_{2^\ell}) - f(X_{2^{\ell-1}})\right), & \text{if } \ell \geq 1 \end{cases}$$

to be the cost associated to one realization of the right hand side, respectively. Then the minimization problem from Remark 27 translates to minimizing

$$\sum_{\ell=0}^{L} N_\ell \cdot C_\ell,$$

i.e., the cost of $A_\varepsilon(f)$, neglecting all constants, with respect to N subject to

$$\operatorname{Var}\left(A_\varepsilon(f)\right) = \sum_{\ell=0}^{L} N_\ell^{-1} \cdot \operatorname{Var}_\ell(f) \leq \varepsilon_V^2.$$

Actually, we do not solve this minimization problem as an integer optimization problem, but as a problem in $\mathbb{R}_+^{L+1}$ and we use the ceiling function to obtain integer values again. This makes sense since $\operatorname{Var}(A_\varepsilon(f))$ is decreasing for increasing components of N and it assures that $N_\ell \geq 1$ for all $\ell = 0, \ldots, L$.

The optimization problem can be solved by means of Lagrange multipliers, i.e., we consider the Lagrange function

$$\mathcal{L}(\lambda, N) = \sum_{\ell=0}^{L} (N_\ell \cdot C_\ell) + \lambda^2 \cdot \big(\mathrm{Var}\big(A_\varepsilon(f)\big) - \varepsilon_V^2\big).$$

For $\ell = 0, \dots, L$ considering $\nabla_{N_\ell} \mathcal{L}(\lambda, N) = 0$ yields

$$N_\ell = \lambda \cdot (\mathrm{Var}_\ell(f)/C_\ell)^{1/2} \tag{82}$$

as a necessary condition for N being an extremal point. Inserting the N_ℓ's in $\nabla_\lambda \mathcal{L}(\lambda, N) = 0$ we obtain

$$\lambda = \varepsilon_V^{-2} \cdot \sum_{\ell=0}^{L} (\mathrm{Var}_\ell(f) \cdot C_\ell)^{1/2}.$$

Hence inserting λ in (82) and using the ceiling function we get

$$N_\ell = \left\lceil \varepsilon_V^{-2} \cdot (\mathrm{Var}_\ell(f)/C_\ell)^{1/2} \cdot \sum_{k=0}^{L} (\mathrm{Var}_k(f) \cdot C_k)^{1/2} \right\rceil \tag{83}$$

for $\ell = 0, \dots, L$.

The question whether an extremal point yields a (local) maximum or a (local) minimum of a Lagrangian function $\mathcal{L}$ leads in general to the bordered Hessian matrix associated to $\mathcal{L}$. For such a bordered Hessian matrix one usually has to consider the leading principle minors and cannot argue via positive definiteness to get conditions for maxima or minima, see, e.g., Simon and Blume [62, Section 19.3] for a discussion and results. Here, we give a rather intuitive argument why N according to (83) yields actually a minimum of $\mathcal{L}(\lambda, N)$. Due to the monotonicity of $\sum_{\ell=0}^{L} N_\ell \cdot C_\ell$ in N and the constraint $\mathrm{Var}(A_\varepsilon(f)) \leq \varepsilon_V^2$ obviously being fulfilled for all components of N tending to infinity, simultaneously, the function $\sum_{\ell=0}^{L} N_\ell \cdot C_\ell$ is unbounded in N and hence can not have a local maximum.

REMARK 29. It remains to approximate the values $\mathrm{Var}_\ell(f)$ for $\ell = 0, \dots, L$. This can, once again, be done by means of direct simulations, and using the empirical variance, see (68). Yet for large L it is clear that a direct simulation for $\mathrm{Var}_L(f)$ might come along with a time demanding computation. At least for a number of replications that is sufficiently large to have a good approximation of $\mathrm{Var}_L(f)$ in the sense of a relatively small asymptotic confidence interval, see Section IV.2.2, compared to the absolute value of $\mathrm{Var}_L(f)$. Therefore, we only approximate $(\mathrm{Var}_\ell(f))_{\ell=0,\dots,L'}$ with $L' \ll L$ and we then estimate $\mathrm{Var}_\ell(f)$ for $\ell = L'+1, \dots, L$ based on the approximations of $(\mathrm{Var}_\ell(f))_{\ell=1,\dots,L'}$. To do this, we proceed in the same way as for the bias.

HYPOTHESIS 2. With the same argument as in Hypothesis 1, we suppose that $(\mathrm{Var}_\ell(f))_{\ell \in \mathbb{N}}$ has the law

$$\mathrm{Var}_\ell(f) \leq c_V \cdot \big(2^{-\ell}\big)^\beta = c_V \cdot \big(2^{-\beta}\big)^\ell$$

for some $\beta > 0$, i.e., the sequence $(\mathrm{Var}_\ell(f))$ is bounded from above by a geometric sequence with initial value c_V, that may depend on $f \in F_p$, and quotient $2^{-\beta}$.

Note that, for the SDE under consideration, see (59), Theorem 2 yields $\beta \geq 1$.

Once we have the approximate values for $(\mathrm{Var}_\ell(f))_{\ell=1,\dots,L'}$ we apply a log-linear-regression, analogously to the one for $(|\,\mathrm{Bias}_\ell(f)|)_{\ell=1,\dots,L'}$, as described in detail in Remark 28, to estimate the parameters β and c_V. It remains to compute the bounds for $(\mathrm{Var}_\ell(f))_{\ell=L'+1,\dots,L}$ according to the law in Hypothesis 2.

Let us recapitulate. For a given error bound $\varepsilon > 0$ that is split into a bias and a variance bound via $\varepsilon^2 = \varepsilon_B^2 + \varepsilon_V^2$, and a given functional $f \in F_p$, (81) and (83) yield an optimal choice for the highest level L depending on ε_B, and the replication numbers N_ℓ on level $\ell = 0, \dots, L$ depending on ε_V, in the sense that the computational cost for the error of the corresponding f-depending multilevel Euler algorithm A_ε applied to f, to fall below the error bound ε, is minimized.

REMARK 30. There are two problems. One is that we do not know the highest level L in advance and hence we do not know what is a good choice for the number L' of levels that are involved in the linear-regressions for the estimation of our upper bound on the strong asymptotic behaviour of $(|\,\mathrm{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\mathrm{Var}_\ell(f))_{\ell\in\mathbb{N}}$. The second point is the cost for the estimations mentioned before. Actually, if the replication numbers on the levels up to L' used for the estimations of c_B, α, c_V, and β are less than the corresponding replication numbers N_ℓ according to (83), then the samples from this estimation may be reused and consequently the cost for the estimation is negligible compared to the overall cost of $A_\varepsilon(f)$. But in general this might not be the case, especially for cost intensive levels ℓ close to L', since N_ℓ is decreasing in ℓ as $\mathrm{Var}_\ell(f)/C_\ell$ is decreasing in ℓ, see (83).

These problems actually lead to the adaptive multilevel Euler algorithm as first introduced in Giles [24], which we consider in Section IV.3.3. In this algorithm there will be no L' to be chosen and no additional samples for the bias and variance estimation will be present, i.e., the cost for these estimations will turn out to be negligible.

REMARK 31. We compare the classical Monte Carlo Euler algorithm, introduced in Section IV.2, to the multilevel Euler algorithm A_ε.

As already mentioned in Remark 30, the computational cost associated to the choice of the highest level L used by A_ε is negligible, at least in terms of an adaptive multilevel Euler algorithm. For the classical Monte Carlo Euler algorithm $A_{L,\widetilde{N}_L}$, which uses the same L as A_ε, recall that the bias of A_ε depends only on the highest level L, this is not the case, since $A_{L,\widetilde{N}_L}$ does not take into account any samples on the levels $\ell = 0, \dots, L'$. That is, the whole cost for the estimation of L is additional, whereas on the contrary the cost for the estimation of the variance of $A_{L,\widetilde{N}_L}$ is in general insignificant since the samples of a classical direct simulation for the estimation of the variance of $f(X_{2^L})$ can be utilized by $A_{L,\widetilde{N}_L}$.

Another advantage of the multilevel Euler algorithm lies in the penalty for choosing a too large highest level L. This might indeed happen since one is rather careful in choosing L, especially for the classical algorithm $A_{L,\widetilde{N}_L}$, since the bias contribution to the error of $A_{L,\widetilde{N}_L}$ is a fixed constant, so if L is too small we can not fall below the bias error bound ε_B. Moreover, while the multilevel Euler algorithm allows to add further levels, the classical Monte Carlo Euler algorithm requires a completely new simulation.

In the following, we consider the effect (on the cost) for choosing the highest level $L+1$ instead of L, which then easily generalizes to $L+\ell$ for $\ell \in \mathbb{N}$. At first we observe that (at least for L being large) the variances of $A_{L+1,\widetilde{N}_{L+1}}$ and $A_{L,\widetilde{N}_L}$ are of the same order. This is due to the convergence of the variance of $(f(X_{2^\ell}))_{\ell\in\mathbb{N}}$ to the variance of $f(X)$, which is due to Theorem 2. In terms of the law in Hypothesis 2, we have $\beta \geq 1$ and can hence apply a

geometric sequence estimation of the remainder of the series associated to $(\mathrm{Var}_\ell(f))_{\ell\in\mathbb{N}}$, as we did for $(|\,\mathrm{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$, that gives us

$$\big|\mathrm{Var}(f(X)) - \mathrm{Var}(f(X_{2^L}))\big| \le c_V \cdot \frac{\big(2^{-\beta}\big)^{L+1}}{1-2^{-\beta}}.$$

As a consequence the numbers of realizations $\widetilde{N}_{L+1}$ and $\widetilde{N}_L$ of the basic experiments almost coincide. Since the cost for one realization of $f(X_{2^{L+1}})$ is almost twice as much as for one realization of $f(X_{2^L})$, we conclude that roughly

$$\mathrm{cost}\big(A_{L+1,\widetilde{N}_{L+1}}\big) = 2 \cdot \mathrm{cost}\big(A_{L,\widetilde{N}_L}\big).$$

We turn to the corresponding effect (on the cost) for the multilevel Euler algorithms $A_{L,N^{(L)}}$ and $A_{L+1,N^{(L+1)}}$ using the replication numbers $N_\ell^{(L)}$ given in (83) in terms of the variance bound from Hypothesis 2 with $\beta = 1$, i.e.,

$$N_\ell^{(L)} = \left\lceil \varepsilon_V^{-2} \cdot \big(c_V \cdot 2^{-2\cdot\ell}\big)^{1/2} \cdot \sum_{\ell=0}^{L} \big(c_V \cdot 2^{-\ell} \cdot 2^\ell\big) \right\rceil.$$

Hence we can assume $N_\ell^{(L)}$ to be of the form

$$N_\ell^{(L)} = c \cdot 2^{-\ell} \cdot (L+1)$$

for some positive constant c. That is, we have $N_\ell^{(L)} \cdot C_\ell = L+1$, up to a multiplicative constant, and therefore

$$\frac{\mathrm{cost}\big(A_{L+1,N^{(L+1)}}, f\big)}{\mathrm{cost}\big(A_{L,N^{(L)}}, f\big)} = \frac{\Big(\sum_{\ell=0}^{L} N_\ell^{(L+1)} \cdot C_\ell\Big) + N_{L+1}^{(L+1)} \cdot C_{L+1}}{\sum_{\ell=0}^{L} N_\ell^{(L)} \cdot C_\ell} = \frac{L+2}{L+1} + \frac{L+2}{(L+1)^2},$$

which is approximately $(L+3)/(L+1)$, i.e., the additional cost is (at least for large L) rather small compared to the overall cost.

We summarize, over estimating L by $L+1$ increases the cost for the classical Monte Carlo Euler algorithm by a factor of about 2 and by a much smaller factor of about $(L+3)/(L+1)$ for the multilevel Euler algorithm, for reasonable L. Likewise, choosing $L+\ell$ with $\ell \in \mathbb{N}$ instead of L yields the factors 2^ℓ and $\prod_{k=1}^{\ell}(L+k+2)/(L+k)$, respectively, and hence even enlarges the difference between the two factors.

Before we introduce and discuss the adaptive multilevel Euler algorithm, we give a general multilevel Euler result (in our setting) for known bias and variance decay α and β, respectively, according to Giles [25, Theorem 1], which is a slight generalization of the original result from Giles [24, Theorem 3.1]. For the proof we refer to Cliffe et al. [10, Theorem 1].

THEOREM 10. *Let $f \in F_p$ and assume that there exist positive constants c_1, c_2, c_3, α, and β with $\alpha \ge 1/2$ and*

(i) $|\,\mathrm{Bias}_\ell\,| \le c_1 \cdot \big(2^{-\alpha}\big)^\ell$,
(ii) $\mathrm{Var}_\ell \le c_2 \cdot \big(2^{-\beta}\big)^\ell$,
(iii) $C_\ell \le c_3 \cdot 2^\ell$.

Then there exists a positive constant c_4 such that for any $\varepsilon < \exp(-1)$ there are values of L and N for which the multilevel Euler algorithm $A_{L,N}(f)$ satisfies

$$\mathrm{error}\big(A_{L,N}, f\big) \le \varepsilon$$

with computational complexity

$$\operatorname{cost}\big(A_{L,N}, f\big) \leq c_4 \cdot \begin{cases} \varepsilon^{-2}, & \text{if } \beta > 1, \\ \varepsilon^{-2} \cdot (\log(\varepsilon)), & \text{if } \beta = 1, \\ \varepsilon^{-2-(1-\beta)/\alpha}, & \text{if } 0 < \beta < 1, \end{cases}$$

which, clearly, yields an upper bound for the ε-complexity of F_p, see Definition 6, i.e.,

$$\operatorname{comp}(\varepsilon, F_p) \leq \operatorname{cost}(A_{L,N}, F_p).$$

REMARK 32. Condition (iii) in Theorem 10 corresponds to X_{2^ℓ} being the Euler-Maruyama approximate solution based on 2^ℓ steps. Actually, this condition can be generalized to different approximation schemes via $C_\ell \leq c_3 \cdot (2^\gamma)^\ell$ and $\alpha \geq 1/2 \cdot \min(\beta, \gamma)$ for a positive constant γ. The only consequence is that the bound on the computational complexity changes to

$$c_4 \cdot \begin{cases} \varepsilon^{-2}, & \text{if } \beta > \gamma, \\ \varepsilon^{-2} \cdot (\log(\varepsilon)), & \text{if } \beta = \gamma, \\ \varepsilon^{-2-(\gamma-\beta)/\alpha}, & \text{if } 0 < \beta < \gamma. \end{cases}$$

Observe that by our definition of the cost of an algorithm, in terms of a random variable, Theorem 10 also covers the case, where adaptive approximation schemes for X are employed.

REMARK 33. Theorem 10 only claims that there exists a choice for L and N. For an explicit formula we refer to the proof of Theorem 3.1 in Giles [24]. Actually, this choice depends on which of the three cases for β we are in. However, it will turn out that the particular choice in (81) and (83) is suitable for the upcoming adaptive multilevel Euler algorithm in Section IV.3.3. This algorithm does not know the decay β in advance and we actually have to estimate it several times during one run of the algorithm. Hence β might, e.g., oscillate around 1, i.e., switch cases for L and N in terms of Theorem 10.

IV.3.3. Adaptive MLMC Euler-Maruyama Algorithm. As mentioned before, we turn to the adaptive multilevel Euler algorithm, as first introduced in Giles [24, Section 5], which in particular solves the problems discussed in Remark 30. We proceed as follows. We first describe the basic algorithm and then we discuss some slight modifications, as, e.g., described in Giles [25, Section 3.4] that are supposed to lead to a more stable algorithm.

DEFINITION 9. Let $f \in F_p$. Then the *basic adaptive multilevel Euler algorithm* reads as follows.

Input: Beside the SDE (drift coefficient, diffusion coefficient, and initial value) and the functional $f \in F_p$, the algorithm needs

- error bound $\varepsilon > 0$ (accuracy) with squared-bias-variance-split $\varepsilon^2 = \varepsilon_B^2 + \varepsilon_V^2$,
- least highest level $L \geq 2$, maximum highest level $L_{\max} \geq L$,
- least replication numbers $\widetilde{N} = (\widetilde{N}_0, \dots, \widetilde{N}_L)$ such that reliable approximations of Var_ℓ for $\ell = 0, \dots, L$ are possible.

Step 1: For $\ell = 0, \dots, L$ evaluate the $\widetilde{N}_\ell$ realizations of $f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})$ and approximate $\operatorname{Var}_\ell(f)$ by means of the empirical variance based on these realizations.

Step 2: Compute optimal numbers of realizations $N = (N_0, \dots, N_L)$ according to (83) based on the approximations of $\operatorname{Var}_0(f), \dots, \operatorname{Var}_L(f)$.

Step 3: Compute number of additional realizations $\mathrm{d}N = N - \widetilde{N}$ and set $\widetilde{N} = N$.
IF($|\mathrm{d}N| > 0$)
There are additional realizations needed to establish the variance bound ε_V^2
→ GoTo *Step 4*
ELSE
No additional realizations are needed to establish the variance bound ε_V^2
→ GoTo *Step 5*.

Step 4: For $\ell = 0, \dots, L$ evaluate extra number of realizations $\mathrm{d}N_\ell$ and approximate $\mathrm{Var}_\ell(f)$ by means of the empirical variance based on the total number N of realizations.
→ GoTo *Step 2*.

Step 5: For $\ell = 0, \dots, L$ compute approximations of $\mathrm{Bias}_\ell(f)$ by means of direct simulations based on the total N realizations, and estimate the decay α as well as the constant c_B for the upper bound of $|\mathrm{Bias}_\ell(f)|$ in terms of Hypothesis 1 by means of a log-linear regression, see Remark 28.
Check for weak convergence:
IF ((80) is fulfilled)
Weak convergence bound ε_B is achieved
→ GoTo *Output*.
ELSE IF($L < L_{\max}$)
Bias error still to large, hence add another level, i.e., set $L = L + 1$.
Estimate $\mathrm{Var}_L(f)$ based on the approximations of $\mathrm{Var}_\ell(f)$ for $\ell = 0, \dots, L-1$, according to Hypothesis 2, by means of a log-linear-regression, see Remark 29
→ GoTo *Step 2*.
ELSE
Give a *Warning* that the weak error bound ε_B could not be established
→ GoTo *Output*.

Output: Compute approximation of $\mathrm{E}(f(X))$ based on the total number N of realizations, cf. (74).

In the following we will denote this algorithm by $A_{\varepsilon,F}^{\mathrm{adp}}$.

REMARK 34. Observe that the input-condition $L \geq 2$ is necessary for the log-linear-regressions of $|\mathrm{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$, respectively. In each regression we want to fit two parameters by means of a linear least-square fit. Hence we need at least two values and the first value we can employ for each fit is the respective one from level $\ell = 1$. The approximations of $|\mathrm{Bias}_0(f)|$ and $\mathrm{Var}_0(f)$ can not be used since they involve no coupled approximations and may therefore distort the corresponding fits, respectively.

The upper bound $L_{\max}$ on the highest level to be used by $A_{\varepsilon,F}^{\mathrm{adp}}$ is necessary to limit the runtime of the algorithm.

Before we turn to the modifications of $A_{\varepsilon,F}^{\mathrm{adp}}$ let us point to the fact that we only have approximations of $(|\mathrm{Bias}_\ell(f)|)_\ell$ and $(\mathrm{Var}_\ell(f))_\ell$, and hence there is no guarantee that the algorithm really satisfies

$$\mathrm{error}\big(A_{\varepsilon,F}^{\mathrm{adp}}, f\big) \leq \varepsilon,$$

as does the algorithm in Theorem 10.

The following two modifications can, e.g., be found in Giles [25, Section 3.4, p. 283].

REMARK 35. We consider a modification concerning the approximations of $\mathrm{Var}_\ell(f)$ and $|\operatorname{Bias}_\ell(f)|$ for $\ell = 2, \dots, L$. First of all, we notice that N_ℓ is typically decreasing for increasing ℓ, cf. Theorem 2. Hence the empirical approximations of $|\operatorname{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ are in general getting less and less accurate for increasing ℓ. Therefore, it is a reasonable idea to use the more accurate approximations from the lower levels for the higher levels. This can be done by iteratively reworking $|\operatorname{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ via

$$|\operatorname{Bias}_\ell(f)| = \max\bigl(|\operatorname{Bias}_\ell(f)|, 1/2 \cdot |\operatorname{Bias}_{\ell-1}(f)| \cdot 2^{-\alpha}\bigr)$$

and

$$\mathrm{Var}_\ell(f) = \max\bigl(\mathrm{Var}_\ell(f), 1/2 \cdot \mathrm{Var}_{\ell-1}(f) \cdot 2^{-\beta}\bigr)$$

for $\ell = 2, \dots, L$. Here α and β denote the estimates of the exponential decays of $(|\operatorname{Bias}_\ell(f)|)_\ell$ and $(\mathrm{Var}_\ell(f))_\ell$ by means of least-square fits, respectively. This post-processing has the effect that $(|\operatorname{Bias}_\ell(f)|)_\ell$ and $(\mathrm{Var}_\ell(f))_\ell$ are not allowed to decrease with more than a factor of $1/2$ relative to the anticipated value. The factor $1/2$ respects the fact that $|\operatorname{Bias}_{\ell-1}(f)|$ and $\mathrm{Var}_{\ell-1}(f)$ may be over-estimated, respectively. This procedure is applied each time $(\mathrm{Var}_\ell(f))_{\ell=0,\dots,L}$ is approximated, i.e., in *Step 1* and *Step 4*, and each time $(|\operatorname{Bias}_\ell(f)|)_{\ell=0,\dots,L}$ is approximated, i.e., in *Step 5*.

REMARK 36. We address a second modification concerning the estimation of the exponential decays α and β of $(|\operatorname{Bias}_\ell(f)|)_{\ell\geq 1}$ and $(\mathrm{Var}_\ell(f))_{\ell\geq 1}$, respectively. By Theorem 2, we know that α is at least $1/2$ and β should at least be 1. However, we are more careful with β, since it regulates the numbers of replications and therefore the accuracy of all estimations of $(|\operatorname{Bias}_\ell(f)|)_\ell$ and $(\mathrm{Var}_\ell(f))_\ell$. Hence we set

$$\alpha = \max(\alpha, 1/2) \qquad \text{and} \qquad \beta = \max(\beta, 1/2)$$

respectively, in *Step 5*.

Since throughout the rest of the thesis we will always include the modifications from Remarks 35 and 36 in the adaptive multilevel Euler algorithm as introduced in Definition 9, we will stick to the notation $A^{\mathrm{adp}}_{\varepsilon,F}$.

So far, whenever a multilevel Euler algorithm involved knowledge on the f-dependent exponential decay of the bias and the variance of the differences $f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})$ we have assumed that f is from F_p. We now consider the case $f \in F_\infty$, more precisely, the intuitive adjustment concerning the upper bounds in Hypothesis 1 and 2, and the numerical problems evolving from these adjustments.

Recall that due to the Lipschitz continuity of $f \in F_p$ the upper bounds

$$\bigl|\operatorname{Bias}_\ell(f)\bigr| \leq c_B \cdot \bigl(2^{-\alpha}\bigr)^\ell \qquad \text{and} \qquad \mathrm{Var}_\ell(f) \leq c_V \cdot \bigl(2^{-\beta}\bigr)^\ell$$

are motivated by the upper bound of the Euler-Maruyama approximate solutions

$$\Bigl(\mathrm{E}\bigl\|X_{2^\ell} - X\bigr\|^2_{L_p([0,1])}\Bigr)^{1/2} \preccurlyeq 2^{-\ell/2}$$

from Theorem 2. Furthermore, Theorem 2 states that exchanging the L_p-norm by the supremum norm worsens this upper bound by a logarithmic factor, namely

$$\Bigl(\mathrm{E}\sup_{0\leq t\leq 1}\bigl|X_{2^\ell}(t) - X(t)\bigr|^2\Bigr)^{1/2} \preccurlyeq 2^{-\ell/2} \cdot \ell^{\,1/2}.$$

Consequently, for $f \in F_\infty$ it stands to reason to change the bias and variance bound to

$$\left|\mathrm{Bias}_\ell(f)\right| \leq c_B \cdot \left(2^{-\alpha_1}\right)^\ell \cdot \ell^{\alpha_2} \tag{84}$$

and

$$\mathrm{Var}_\ell(f) \leq c_V \cdot \left(2^{-\beta_1}\right)^\ell \cdot \ell^{\beta_2} \tag{85}$$

for positive constants c_B, α_1, α_2 and c_V, β_1, β_2, respectively. Note that naturally it also stands to reason to incorporate this change in the modification described in Remark 36, i.e., α_2 and β_2 are bounded from below by the constant $1/2$, too.

Observe that in the respective linear least-square fits involved in the adaptive multilevel algorithm $A^{\mathrm{adp}}_{\varepsilon,F}$ we now have to fit three parameters, i.e., we also need at least three values for each fit, and hence we have to choose an input highest level $L \geq 3$ for $A^{\mathrm{adp}}_{\varepsilon,F}$, cf. Remark 34. Moreover, note that though the effect of the new upper bounds vanishes for large ℓ, since for all positive constants γ_1, γ_2 and δ there exists $\ell_\delta = \ell_\delta(\gamma_1, \gamma_2) \in \mathbb{N}$ such that

$$\left(2^{-\gamma_1}\right)^\ell \cdot \ell^{\gamma_2} \leq \left(2^{-\gamma_1+\delta}\right)^\ell \tag{86}$$

for all $\ell \geq \ell_\delta$, the adaptive multilevel Euler algorithm $A^{\mathrm{adp}}_{\varepsilon,F}$ might choose a highest level L such that the logarithmic factor still has a substantial contribution. Furthermore, the algorithm $A^{\mathrm{adp}}_{\varepsilon,F}$ usually starts with $L = 3$ and then iteratively increases L by one. So in the beginning the asymptotics of (86) definitely does not kick in, yet.

From now on we will use the notation $A^{\mathrm{adp}}_{\varepsilon,F}$ for the adaptive multilevel Euler algorithm that supposes the asymptotic behaviours (84) and (85) for $(|\mathrm{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\mathrm{Var}_\ell(f))_{\ell\in\mathbb{N}}$, respectively. Moreover, by $A^{\mathrm{adp}^\star}_{\varepsilon,F}$ we denote the adaptive multilevel Euler algorithm that supposes the corresponding asymptotic behaviours from Hypothesis 1 and Hypothesis 2.

Remark 37. We present the major problem concerning the adjusted bias and variance bounds (84) and (85), namely the linear least-square fits for the approximations of c_B, α_1, α_2 and c_V, β_1, β_2, respectively. To this end, note that the design matrix for the respective least-square fits, see Section A.2 in the Appendix, is given by $D = D(L) = [D_1(L), D_2(L), D_3(L)]$ with $D_1(L) = [1, \ldots, L]^T$, $D_2(L) = [\log_2(1), \ldots, \log_2(L)]^T$, and $D_3(L) = [1, \ldots, 1]^T$ for $L \geq 3$. Furthermore, recall that a main indicator for the reliability of a least-square fit is the condition number $\kappa(D^T D)$ of $D^T D$, see Remark 43. That is, if $\kappa(D^T D)$ is large then even a small perturbation of the values $(|\mathrm{Bias}_\ell(f)|)_{\ell=1,\ldots,L}$ respective $(\mathrm{Var}_\ell(f))_{\ell=1,\ldots,L}$ can lead to a relatively large deviation in the parameter vectors $(c_B, \alpha_1, \alpha_2)$ and (c_V, β_1, β_2), respectively, and these values of $|\mathrm{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ are (only) approximated by the corresponding empirical estimates based on the N_ℓ realizations according to (83).

The rather large condition numbers $\kappa(D^T D)$ for reasonable L, in fact for $L = 3, \ldots, 30$ are pictured in Figure 1. In comparison, in Figure 2, we also picture the considerably smaller corresponding condition numbers for $\overline{D} = \overline{D}(L) = \left[\overline{D}_1(L), \overline{D}_2(L)\right]$ with $\overline{D}_1(L) = [1, \ldots, L]^T$ and $\overline{D}_2(L) = [1, \ldots, 1]^T$, the design matrix for the least-square fits according to Hypothesis 1 and Hypothesis 2.

IV.4. Random Bit Euler-Maruyama Schemes

In this section we study multilevel Euler algorithms for which the underlying Euler-Maruyama approximation schemes may only use random bits instead of random numbers form $[0, 1]$. This in particular excludes the use of Brownian increments, and the coupling of consecutive approximation schemes becomes a non-trivial issue. Actually, we study two

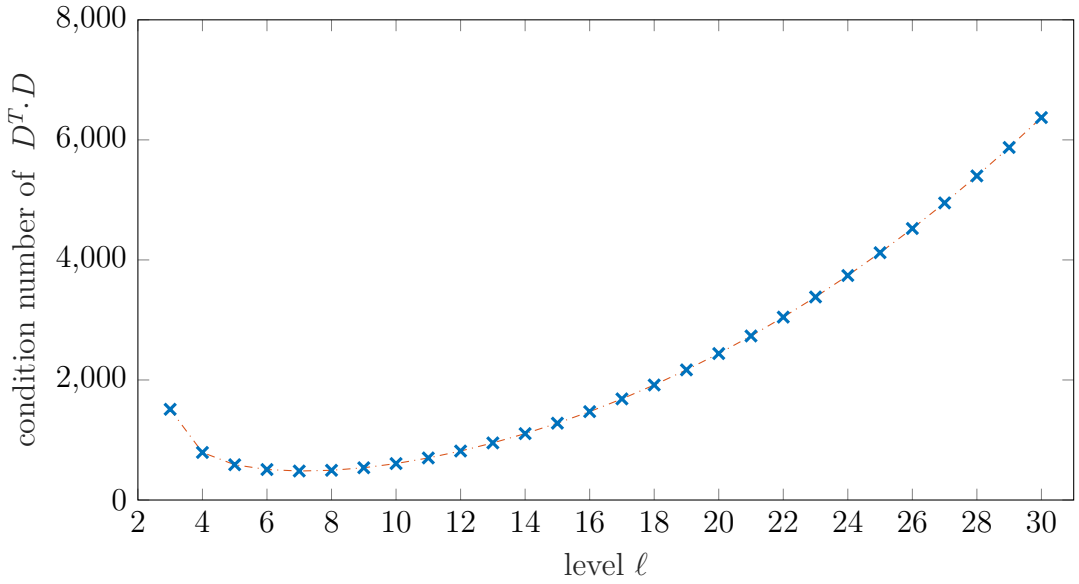

FIGURE 1. Condition numbers $\kappa(D^T D)$, where $D = D(L)$ is the design matrix for the linear least-square fits used to approximate the parameter vectors $(c_B, \alpha_1, \alpha_2)$ and (c_V, β_1, β_2) describing the decay of the upper bound of $(|\operatorname{Bias}_\ell(f)|)_\ell$ and $(\operatorname{Var}_\ell(f))_\ell$ in terms of (84) and (85), respectively.

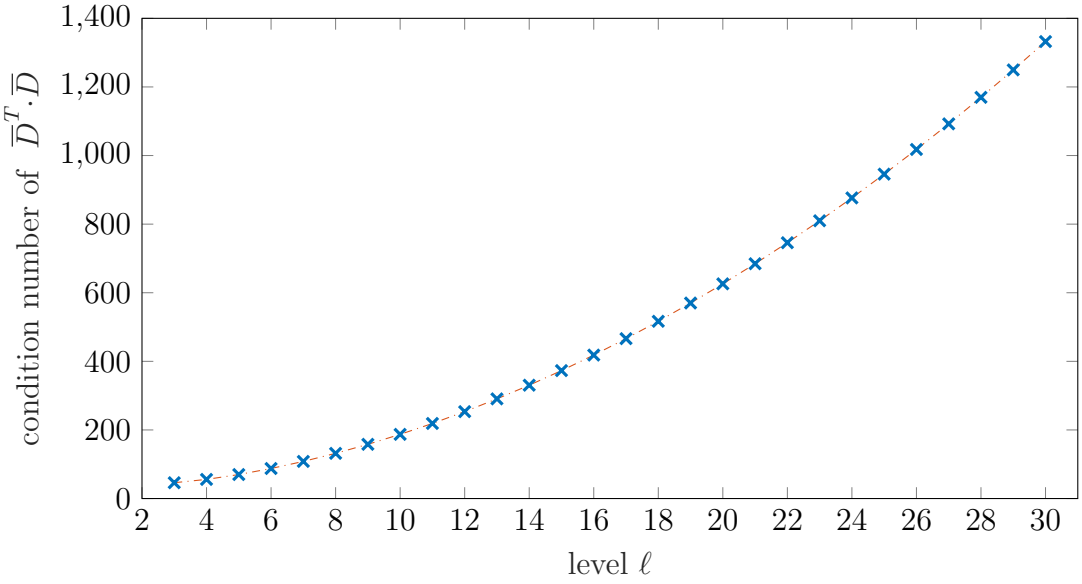

FIGURE 2. Condition numbers $\kappa\big(\overline{D}^T \overline{D}\big)$, where $\overline{D} = \overline{D}(L)$ is the design matrix for the linear least-square fits used to approximate the parameter vectors (c_B, α) and (c_V, β) describing the decay of the upper bound of $(|\operatorname{Bias}_\ell(f)|)_\ell$ and $(\operatorname{Var}_\ell(f))_\ell$ in terms of Hypothesis 1 and 2, respectively.

different kinds of random bit Euler-Maruyama schemes and their associated multilevel couplings. Both approximation schemes rely on the particular (random) bit approximation of the standard normal distribution, as defined and analyzed in Section III.2.1. We briefly recapitulate this bit approximation.

Let Φ denote the distribution function of the standard normal distribution with inverse distribution function Φ^{-1}, and let Y be standard normally distributed. At first we recall the

truncation operator

$$T^{(q)}\colon [0,1[\to D^{(q)}, \quad x \mapsto \frac{\lfloor 2^q \cdot x \rfloor}{2^q} + 2^{-(q+1)},$$

where $q \in \mathbb{N}$ and $D^{(q)} = \big\{\sum_{i=1}^{q} b_i \cdot 2^{-i} + 2^{-(q+1)} \colon b_i \in \{0,1\} \text{ for } i = 1,\dots,q\big\}$, see (23). Then

$$Y^{(q)} = \Phi^{-1} \circ T^{(q)} \circ \Phi(Y) \tag{87}$$

serves as a canonical approximation of Y, see (24). Moreover, recall that $T^{(q)} \circ \Phi(Y)$ is uniformly distributed on $D^{(q)}$ and hence q random bits suffice to simulate the distribution of $Y^{(q)}$. For further properties, see Theorem 4. For a standard normally distributed random vector Y an approximation $Y^{(q)}$ is obtained by applying $\Phi^{-1} \circ T^{(q)} \circ \Phi$ to each of the components of Y separately.

IV.4.1. Independent Increments (Scheme 1). Here, we use the bit approximation (87) in a straightforward way, i.e., we study a random bit Euler-Maruyama scheme, analogously to the random bit Milstein scheme in Section III.2.3, for the Euler-Maruyama scheme see Section II.2.1, with

$$V_k = V_{k,m} = m^{-1/2} \cdot \big(m^{1/2} \cdot \big(W(t_{k,m}) - W(t_{k-1,m})\big)\big)^{(q)} \tag{88}$$

for $k = 1,\dots,m$. A suitable coupling is easily achieved by

$$\widetilde{V}_k = \widetilde{V}_{k,m} = V_{2\cdot k-1,m} + V_{2\cdot k,m} \tag{89}$$

for $k = 1,\dots,m/2$, cf. (73). To indicate the dependence of this coupled Euler-Maruyama scheme on the bit number q we use the notations $X_m^{(q)}$ and $\widetilde{X}_{m/2}^{(q)}$, and likewise $V_{k,m}^{(q)}$ and $\widetilde{V}_{k,m/2}^{(q)}$ for the bit approximations of the Brownian increments. For convenience of the reader we present the random bit Euler-Maruyama scheme

$$\begin{aligned} X_m^{(q)}(t_{0,m}) &= x_0, \\ X_m^{(q)}(t_{k,m}) &= X_m^{(q)}(t_{k-1,m}) + a\big(X_m^{(q)}(t_{k-1},m)\big) \cdot m^{-1} + b\big(X_m^{(q)}(t_{k-1,m})\big) \cdot V_{k,m}^{(q)} \end{aligned}$$

for $k = 1,\dots,m$. This corresponds to the construction and also to the notation of the random bit Milstein scheme in Section III.2.3, however, since the random bit Milstein scheme will not be considered throughout this thesis again, we can allow ourselves this reuse of notation. The coupled random bit Euler-Maruyama scheme $\widetilde{X}_{m/2}^{(q)}$ reads as

$$\begin{aligned} \widetilde{X}_{m/2}^{(q)}(t_{0,m/2}) &= x_0, \\ \widetilde{X}_{m/2}^{(q)}(t_{k,m/2}) &= \widetilde{X}_{m/2}^{(q)}(t_{k-1,m/2}) + (m/2)^{-1} \cdot a\big(\widetilde{X}_{m/2}^{(q)}(t_{k-1,m/2})\big) \\ &\qquad + b\big(\widetilde{X}_{m/2}^{(q)}(t_{k-1,m/2})\big) \cdot \widetilde{V}_{k,m/2}^{(q)} \end{aligned}$$

for $k = 1,\dots,m/2$. In order to approximate X at any point in the unit interval, we extend $X_m^{(q)}(t_0),\dots,X_m^{(q)}(t_m)$ and $\widetilde{X}_{m/2}^{(q)}(t_0),\dots,\widetilde{X}_{m/2}^{(q)}(t_{m/2})$ by piecewise linear interpolation onto the subintervals $]t_{k-1},t_k[$ for $k = 1,\dots,m$, and $k = 1,\dots,m/2$, respectively, as we did for the classical Euler-Maruyama scheme in Section II.2.1. Proper relations of the bit number q and the number m of Euler-Maruyama steps will be presented in Section IV.5.

Observe that the simulation of the joint distribution of $X_m^{(q)}$ and $\widetilde{X}_{m/2}^{(q)}$ requires $d \cdot m \cdot q$ random bits. Clearly, the bit approximations $V_{1,m}^{(q)},\dots,V_{m,m}^{(q)}$ as well as $\widetilde{V}_{1,m/2}^{(q)},\dots,\widetilde{V}_{m/2,m/2}^{(q)}$ of the Brownian increments are independent. We stress, however, that the distributions of

$\widetilde{V}^{(q)}_{k,m/2}$ and $V^{(q)}_{k,m/2}$ do not coincide, which introduces an additional bias term in the multilevel analysis. We also say that the telescoping sum property (69) is violated.

IV.4.2. Matching Distributions (Scheme 2). For the second type of a random bit Euler-Maruyama scheme we combine the bit approximation (87) of the standard normal distribution with the Lévy-Ciesielski representation of the Brownian motion, as described in Section II.1.

Recall that the Lévy-Ciesielski representation of the d-dimensional standard Brownian motion W is given by

$$W(t) = \sum_{i=0}^{\infty} \sum_{j=1}^{2^{\max(i-1,0)}} s_{i,j}(t) \cdot Y_{i,j}$$

with convergence, e.g. in mean square with respect to the L_2-norm, see Remark 2. Here $s_{0,1}, s_{1,1}, \dots$ is the sequence of Schauder functions, see (6), and $Y_{0,1}, Y_{1,1}, \dots$ is an independent sequence of d-dimensional random vectors, each of which has independent standard normally distributed components. In particular, for

$$m = 2^{\ell}, \qquad \ell \in \mathbb{N}_0,$$

and $k = 1, \dots, m$ we have

$$W(t_{k,m}) = \sum_{i=0}^{\ell} \sum_{j=1}^{2^{\max(i-1,0)}} s_{i,j}(t) \cdot Y_{i,j}$$

for $t_{k,m} = k/m$. Analogously to approximating the distribution of the standard Brownian bridge in $L_2([0,1])$ in Section III.2.2, for a discussion of the relation of the Lévy-Ciesielski representation of a Brownian motion and a Brownian bridge see Remark 3, we apply the bit approximation (87) to each of the random vectors $Y_{i,j}$. Moreover, to respect Itô calculus we normalize the variances. For this we put

$$v(q) = \mathrm{Var}\big(Y^{(q)}\big),$$

where Y denotes a standard normally distributed random variable.

Theorem 5 suggests that we spend $2 \cdot (\ell + 1 - i)$ random bits for the approximation of each component of $Y_{i,j}$, i.e., the number of random bits is chosen according to the size of the support of the corresponding Schauder function $s_{i,j}$. In that sense the choice for $Y_{0,1}$ is a straightforward generalization of the choice in Theorem 5. The particular choice leads to

$$V_k = V_{k,m} = \sum_{i=0}^{\ell} \sum_{j=1}^{2^{\max(i-1,0)}} \big(s_{i,j}(t_{k,m}) - s_{i,j}(t_{k-1,m})\big) \cdot v\big(2 \cdot (\ell + 1 - i)\big)^{-1/2} \cdot Y_{i,j}^{(2\cdot(\ell+1-i))}$$

for $k = 1, \dots, m$. We add that only $\ell + 1$ out of the 2^{ℓ} summands in the definition of V_k are different from zero. A suitable coupling is achieved via

$$\widetilde{V}_k = \widetilde{V}_{k,m/2} = \sum_{i=0}^{\ell-1} \sum_{j=1}^{2^{\max(i-1,0)}} \big(s_{i,j}(t_{k,m/2}) - s_{i,j}(t_{k-1,m/2})\big) \cdot v\big(2 \cdot (\ell - i)\big)^{-1/2} \cdot Y_{i,j}^{(2\cdot(\ell-i))}$$

for $k = 1, \dots, m/2$. By definition of $Y_{i,j}^{(2\cdot(\ell+1-i))}$ and $Y_{i,j}^{(2\cdot(\ell-i))}$ we get

$$Y_{i,j}^{(2\cdot(\ell-i))} = \Phi^{-1} \circ T^{(2\cdot(\ell-i))} \circ \Phi\big(Y_{i,j}^{(2\cdot(\ell+1-i))}\big),$$

i.e., $\widetilde{V}_k$ can be obtained from V_k by reducing all of the relevant bit numbers by two. The corresponding random bit Euler-Maruyama schemes are denoted by $X_m^\dagger$ and $\widetilde{X}_{m/2}^\dagger$ with approximations $V_{k,m}^\dagger$ and $\widetilde{V}_{k,m/2}^\dagger$ of the respective Brownian increments. We do not write down the whole schemes $X_m^\dagger$ and $\widetilde{X}_{m/2}^\dagger$, since we have just stated the analogs for Scheme 1 in the afore section. Like for Scheme 1 and for the classical Euler-Maruyama scheme, we obtain continuous approximations on the unit interval via linear interpolation of the schemes $X_m^\dagger$ and $\widetilde{X}_{m/2}^\dagger$ based on the respective breakpoints t_k.

The simulation of the joint distribution of $X_m^\dagger$ and $\widetilde{X}_{m/2}^\dagger$ requires $d \cdot (2^{\ell+2} - 2)$ random bits, cf. Theorem 5, and we have $\widetilde{V}_{k,m/2}^\dagger = V_{k,m/2}^\dagger$ by definition. Hence the telescoping sum property (69) is satisfied, i.e., in comparison to Scheme 1 we have no additional bias term for the error for the corresponding multilevel Monte Carlo algorithm. We stress, however, that we do not have the independence of $V_{1,m}^\dagger, \dots, V_{m,m}^\dagger$ or of $\widetilde{V}_{1,m/2}^\dagger, \dots, \widetilde{V}_{m/2,m/2}^\dagger$.

IV.4.3. Strong Error Analysis of Scheme 1. In this section we are interested in estimates with respect to the supremum norm on the unit interval. In the following, we denote the classical Euler-Maruyama scheme by $X_m^\star$ with Brownian increments $V_{k,m}^\star$ for $k = 1, \dots, m$. Analogously, we use the notations $\widetilde{X}_{m/2}^\star$ and $\widetilde{V}_{k,m/2}^\star$ for the coupled approximation. Furthermore, in this section we consider constants, i.e., we do not use the notations $\preccurlyeq$ and $\succcurlyeq$, respectively. In particular, we assume that the global Lipschitz bounds of the drift and diffusion coefficients a and b of the SDE under consideration are bounded from above by $\gamma > 0$.

Regarding the well-known upper bound on the error of the Euler-Maruyama approximate solutions $(X_m^\star)_{m\in\mathbb{N}}$ presented in Theorem 2, we will provide an upper bound for the difference between the classical Euler Maruyama scheme $X_m^\star$ and its random bit approximation $X_m^{(q)}$, that is independent from the number of Euler-Maruyama steps m.

Remark 38. We gather some properties of the scheme $X_m^{(q)}$.

(a) We have independence of $V_{1,m}^{(q)}, \dots, V_{m,m}^{(q)}$.

(b) By Theorem 4, concerning the random bit approximation of the standard normal distribution, there exists a positive constant c such that

$$\Big(\mathrm{E}\big|V_{k,m}^\star - V_{k,m}^{(q)}\big|^2\Big)^{1/2} \le c \cdot m^{-1/2} \cdot 2^{-q/2} \cdot q^{-1/2}$$

and

$$\mathrm{E}\big(V_{k,m}^{(q)}\big) = 0$$

for all $m \in \mathbb{N}$, $k = 1, \dots, m$, and $q \in \mathbb{N}$. Furthermore, we have

$$\sup_{m\in\mathbb{N}} \sup_{k=1,\dots,m} \sup_{q\in\mathbb{N}} \Big(m^{1/2} \cdot \big(\mathrm{E}\big|V_{k,m}^{(q)}\big|^r\big)^{1/r}\Big) < \infty$$

for all $r \ge 1$.

(c) We have $\mathrm{E}\big\|X_m^{(q)}\big\|_{\sup}^2 < \infty$ for every $m \in \mathbb{N}$ and $q \in \mathbb{N}$.

Lemma 14. *There exists a positive constant c such that for all $m, q \in \mathbb{N}$ we have*

$$\max_{k=0,\dots,m} \Big(\mathrm{E}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2\Big)^{1/2} \le c \cdot 2^{-q/2} \cdot q^{-1/2}.$$

PROOF. This proof follows the standard analysis for the classical Euler-Maruyama scheme. In a first step we separate the diffusion terms. For $k = 0, \dots, m-1$ we have

$$X_m^\star(t_{k+1}) - X_m^{(q)}(t_{k+1}) = \xi + \zeta,$$

where

$$\xi = X_m^\star(t_k) - X_m^{(q)}(t_k) + m^{-1} \cdot \big(a\big(X_m^\star(t_k)\big) - a\big(X_m^{(q)}(t_k)\big)\big)$$

and

$$\zeta = b\big(X_m^\star(t_k)\big) \cdot V_{k+1}^\star - b\big(X_m^{(q)}(t_k)\big) \cdot V_{k+1}^{(q)}.$$

For any pair of components ξ_i and ζ_i of ξ and ζ, respectively, we have

$$\mathrm{E}\big(\xi_i \cdot \zeta_i\big) = \mathrm{E}(\xi_i) \cdot \mathrm{E}(\zeta_i) = 0,$$

due to properties (a) – (c) from Remark 38. It follows that

$$\mathrm{E}\big|X_m^\star(t_{k+1}) - X_m^{(q)}(t_{k+1})\big|^2 = \mathrm{E}\big(|\xi|^2\big) + \mathrm{E}\big(|\zeta|^2\big).$$

The Lipschitz continuity of a yields

$$\mathrm{E}\big(|\xi|^2\big)^{1/2} \le (1 + \gamma/m) \cdot \Big(\mathrm{E}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2\Big)^{1/2}. \tag{90}$$

Moreover, adding zero in a smart way and an application of the Jensen-inequality yield

$$\begin{aligned}\mathrm{E}\big(|\zeta|^2\big) \le 2 \cdot \Big(&\mathrm{E}\big|b\big(X_m^\star(t_k)\big) \cdot \big(V_{k+1}^\star - V_{k+1}^{(q)}\big)\big|^2 \\ &+ \mathrm{E}\big|\big(b\big(X_m^\star(t_k)\big) - b\big(X_m^{(q)}(t_k)\big)\big) \cdot V_{k+1}^{(q)}\big|^2\Big).\end{aligned} \tag{91}$$

For the first term property (a) from Remark 38 gives

$$\mathrm{E}\big|b\big(X_m^\star(t_k)\big) \cdot \big(V_{k+1}^\star - V_{k+1}^{(q)}\big)\big|^2 \le \mathrm{E}\big|b\big(X_m^\star(t_k)\big)\big|^2 \cdot \mathrm{E}\big|V_{k+1}^\star - V_{k+1}^{(q)}\big|^2.$$

Using

$$\mathrm{E}\big|b\big(X_m^\star(t_k)\big)\big|^2 \le 2 \cdot \gamma^2 \cdot \mathrm{E}\big|X_m^\star(t_k)\big|^2 + 2 \cdot |b(0)|^2$$

together with property (b) from Remark 38 we obtain existence of a positive constant c_1 such that

$$\mathrm{E}\big|b\big(X_m^\star(t_k)\big) \cdot \big(V_{k+1}^\star - V_{k+1}^{(q)}\big)\big|^2 \le c_1 \cdot m^{-1} \cdot 2^{-q} \cdot q^{-1}$$

for all $m, q \in \mathbb{N}$ and $k = 0, \dots, m-1$. For the second term in (91) property (a) from Remark 38 also gives

$$\mathrm{E}\big|\big(b\big(X_m^\star(t_k)\big) - b\big(X_m^{(q)}(t_k)\big)\big) \cdot V_{k+1}^{(q)}\big|^2 \le \mathrm{E}\big|b\big(X_m^\star(t_k)\big) - b\big(X_m^{(q)}(t_k)\big)\big|^2 \cdot \mathrm{E}\big|V_{k+1}^{(q)}\big|^2.$$

Moreover, the Lipschitz continuity of b implies

$$\mathrm{E}\big|b\big(X_m^\star(t_k)\big) - b\big(X_m^{(q)}(t_k)\big)\big|^2 \le \gamma^2 \cdot \mathrm{E}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2,$$

which yields, together with property (b) from Remark 38, the existence of a positive constant c_2 such that

$$\mathrm{E}\big|\big(b\big(X_m^\star(t_k)\big) - b\big(X_m^{(q)}(t_k)\big)\big) \cdot V_{k+1}^{(q)}\big|^2 \le c_2 \cdot m^{-1} \cdot \mathrm{E}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2$$

for all $m, q \in \mathbb{N}$ and $k = 0, \dots, m-1$. Altogether, it follows that

$$\mathrm{E}\big(|\zeta|^2\big) \le c_3/m \cdot \Big(2^{-q} \cdot q^{-1/2} + \mathrm{E}\big|X_m^\star(t_k) + X_m^{(q)}(t_k)\big|^2\Big) \tag{92}$$

with $c_3 = 2 \cdot \max(c_1, c_2)$. Observe that

$$(1+\gamma/m)^2 = 1 + (2\cdot\gamma + \gamma^2/m)/m \le 1 + (2\cdot\gamma+\gamma^2)/m,$$

and set $c = 2\cdot\gamma + \gamma^2 + c_3$. Then combining the estimates for $\mathrm{E}(|\xi|^2)$ and $\mathrm{E}(|\zeta|^2)$, i.e., (90) and (92), we get

$$\mathrm{E}\big|X_m^\star(t_{k+1}) - X_m^{(q)}(t_{k+1})\big|^2 \le (1+c/m)\cdot \mathrm{E}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2 + c/m\cdot 2^{-q}\cdot q^{-1} \tag{93}$$

for all $m, q \in \mathbb{N}$ and $k = 0, \dots, m-1$.

A discrete Gronwall-inequality or a straightforward computation, namely k-times recursively applying (93), yields

$$\begin{aligned}
&\mathrm{E}\big|X_m^\star(t_{k+1}) - X_m^{(q)}(t_{k+1})\big|^2 \\
&\qquad \le (1+c/m)^{k+1}\cdot \mathrm{E}\big|X_m^\star(t_0) - X_m^{(q)}(t_0)\big|^2 + \sum_{\ell=0}^{k}(1+c/m)^\ell \cdot c/m\cdot 2^{-q}\cdot q^{-1} \\
&\qquad \le (k+1)\cdot(1+c/m)^m \cdot c/m\cdot 2^{-q}\cdot q^{-1}
\end{aligned} \tag{94}$$

with c according to (93). The last inequality is due to $X_m^\star(t_0) = x_0 = X_m^{(q)}(t_0)$ and $(1 + c/m)^k \le (1+c/m)^m$ for all $k \le m$. Since $(1+c/m)^m$ converges to $\exp(c)$ as m tends to infinity, we have existence of a positive constant c_4 such that

$$(1+c/m)^m \le c_4 \cdot \exp(c).$$

Hence for $c_5 = c_4\cdot\exp(c)\cdot c$, the Gronwall argument (94) gives

$$\mathrm{E}\big|X_m^\star(t_{k+1}) - X_m^{(q)}(t_{k+1})\big|^2 \le c_5\cdot(k+1)/m\cdot 2^{-q}\cdot q^{-1}$$

for all $k = 0, \dots, m-1$, and hereby the claim follows. ∎

LEMMA 15. *There exists a positive constant c such that for all $m, q \in \mathbb{N}$ we have*

$$\Big(\mathrm{E}\big\|X_m^\star - X_m^{(q)}\big\|_{\sup}^2\Big)^{1/2} \le c\cdot 2^{-q/2}\cdot q^{-1/2}.$$

PROOF. This proof uses standard martingale arguments to exchange the maximum with the expectation in the error bound of Lemma 14.

Since $X_m^\star$ and $X_m^{(q)}$ are piecewise linear, it suffices to consider the difference of $X_m^\star$ and $X_m^{(q)}$ at their common breakpoints $\{t_k = k/m \colon k = 0, \dots, m\}$. Since the schemes do coincide in t_0, for $k = 1, \dots, m$ we have

$$\begin{aligned}
X_m^\star(t_k) - X_m^{(q)}(t_k) = m^{-1}\cdot &\sum_{\ell=1}^{k}\Big(a\big(X_m^\star(t_{\ell-1})\big) - a\big(\big(X_m^{(q)}(t_{\ell-1})\big)\Big) \\
&+ \sum_{\ell=1}^{k}\Big(b\big(X_m^\star(t_{\ell-1})\big)\cdot V_\ell^\star - b\big(X_m^{(q)}(t_{\ell-1})\big)\cdot V_\ell^{(q)}\Big),
\end{aligned} \tag{95}$$

and therefore we can use the Jensen-inequality to separate the drift terms from the diffusion terms, i.e.,

$$\mathrm{E}\max_{k=0,\dots,m}\big|X_m^\star(t_k) - X_m^{(q)}(t_k)\big|^2 \le 2\cdot \mathrm{E}\max_{k=1,\dots,m}\big|Z_k\big|^2 + 2\cdot\max_{k=1,\dots,m}\big|R_k\big|^2$$

with

$$Z_k = m^{-1}\cdot\sum_{\ell=1}^{k}\Big(a\big(X_m^\star(t_{\ell-1})\big) - a\big(X_m^{(q)}(t_{\ell-1})\big)\Big)$$

and

$$R_k = \sum_{\ell=1}^{k} \Big(b\big(X_m^\star(t_{\ell-1})\big) \cdot V_\ell^\star - b\big(X_m^{(q)}(t_{\ell-1})\big) \cdot V_\ell^{(q)} \Big).$$

We treat the drift terms first. Due to the Jensen-inequality and exploiting the Lipschitz continuity of the drift coefficient a, we get

$$\begin{aligned} \mathrm{E} \max_{k=1,\dots,m} \big|Z_k\big|^2 &\le \mathrm{E} \max_{k=1,\dots,m} \frac{k}{m^2} \cdot \sum_{\ell=1}^{k} \big| a\big(X_m^\star(t_{\ell-1})\big) - a\big(X_m^{(q)}(t_{\ell-1})\big) \big|^2 \\ &\le \frac{\gamma^2}{m} \cdot \sum_{\ell=1}^{m} \mathrm{E} \big| X_m^\star(t_{\ell-1}) - X_m^{(q)}(t_{\ell-1}) \big|^2 \\ &\le \gamma^2 \cdot \max_{k=0,\dots,m} \mathrm{E} \big| X_m^\star(t_k) - X_m^{(q)}(t_k) \big|^2. \end{aligned} \tag{96}$$

For the diffusion terms, we note that $(R_k)_{k=1,\dots,m}$ is a martingale due to properties (b) and (c) from Remark 38. Consequently, the Doob maximal inequality, see Karatzas and Shreve [39, Theorem 1.3.8 (iv)] yields

$$\mathrm{E} \max_{k=1,\dots,m} |R_k|^2 \le 4 \cdot \mathrm{E}\, |R_m|^2.$$

We use (95), the Jensen-inequality, and (96), in this order, to obtain

$$\begin{aligned} \mathrm{E} \big| R_m \big|^2 &= \mathrm{E} \big| X_m^\star(t_m) - X_m^{(q)}(t_m) - Z_m \big|^2 \\ &\le 2 \cdot \mathrm{E} \big| X_m^\star(t_m) - X_m^{(q)}(t_m) \big|^2 + 2 \cdot \mathrm{E} \big| Z_m \big|^2 \\ &\le 2 \cdot (1 + \gamma^2) \cdot \max_{k=0,\dots,m} \mathrm{E} \big| X_m^\star(t_k) - X_m^{(q)}(t_k) \big|^2. \end{aligned}$$

Altogether, we get

$$\mathrm{E} \max_{k=0,\dots,m} \big| X_m^\star(t_k) - X_m^{(q)}(t_k) \big|^2 \le 18 \cdot (1 + \gamma^2) \cdot \max_{k=0,\dots,m} \mathrm{E} \big| X_m^\star(t_k) - X_m^{(q)}(t_k) \big|^2.$$

Apply Lemma 14 to establish the claim for $X_m^\star - X_m^{(q)}$. ∎

Remark 39. Before we continue with the strong error analysis of Scheme 1, we consider the results from Lemma 15 in view of random bit approximations of probability distributions, see Section III.2, particularly Section III.2.3. Recall that $X_m^{(q)}$ is a random bit approximation of X that is based on $d \cdot m \cdot q$ random bits, and hence can be utilized to establish an upper but on $\mathrm{rbit}(\mu, p)$, where μ denotes the distribution of the solution X of the SDE (59) on the separable Banach space $C([0,1], \mathbb{R}^r)$. Since $\mathrm{rbit}(\mu, p)$ is based on the Wasserstein distance d of order 2, combining Lemma 15 with Theorem 2 we get existence of a positive constant c such that

$$\mathrm{rbit}(\mu, p) \le c \cdot \min_{d \cdot m \cdot q \le p} \big(m^{-1/2} \cdot (\ln(m+1))^{1/2} + 2^{-q/2} \cdot q^{-1/2} \big). \tag{97}$$

Actually, it can be shown that the right hand side of (97) is of order $p^{-1/2} \cdot \ln(p)$. In the scalar case, i.e., $r = d = 1$, and under a slightly stronger smoothness assumption on the coefficients, namely a and b are assumed to be differentiable with bounded Lipschitz continuous derivatives, as well as a non-degeneracy assumption on the diffusion coefficient b, which in particular excludes pathological cases yielding a deterministic solution, we have shown in Theorem 6 and Corollary 2 that $\mathrm{rbit}(\mu, p)$ is of the order $p^{-1/2}$ for the separable Banach space $L_2([0,1])$. Hence, at least in the scalar case, the upper bound obtained via

(97) is sharp, up to logarithmic factors, but matching upper and lower bounds seem to be unknown in this case.

Following the proof of Lemma 14 and Lemma 15, one may establish analogous results for the difference $\widetilde{X}^{\star}_{m/2} - \widetilde{X}^{(q)}_{m/2}$. We only formulate the analogue to Lemma 15.

LEMMA 16. *There exists a positive constant c such that for all $q, m \in N$ we have*

$$\Big(\mathrm{E}\big\|\widetilde{X}^{\star}_{m/2} - \widetilde{X}^{(q)}_{m/2}\big\|^2_{\sup}\Big)^{1/2} \leq c \cdot 2^{-q/2} \cdot q^{-1/2}.$$

IV.5. Random Bit MLMC Algorithms

The random bit multilevel Euler-Maruyama algorithms based on Scheme 1 and Scheme 2 are obtained from the classical multilevel Euler algorithm $A_{L,N}$, as described in Section IV.3.1, see in particular (74), and which is henceforth denoted by $A^{\star}_{L,N}$ with coupled Euler-Maruyama schemes $X^{\star}_{2^\ell}$ and $\widetilde{X}^{\star}_{2^{\ell-1}}$, via replacing the approximations $X^{\star}_{2^\ell}$ and $\widetilde{X}^{\star}_{2^{\ell-1}}$ by their random bit approximations $X^{(q)}_{2^\ell}$ and $\widetilde{X}^{(q)}_{2^{\ell-1}}$ or $X^{\dagger}_{2^\ell}$ and $\widetilde{X}^{\dagger}_{2^{\ell-1}}$, respectively. The corresponding random bit multilevel algorithms are denoted by $A^{q}_{L,N}$ and $A^{\dagger}_{L,N}$, respectively. For convenience of the reader we restate the general multilevel Euler algorithm $A_{L,N}$, that might represent any of the three algorithms $A^{\star}_{L,N}, A^{q}_{L,N}$, and $A^{\dagger}_{L,N}$, namely

$$A_{L,N}(f) = \frac{1}{N_0} \cdot \sum_{i=1}^{N_0} f\big(X_{2^0,i}\big) + \sum_{\ell=1}^{L} \frac{1}{N_\ell} \cdot \sum_{i=1}^{N_\ell} \big(f\big(X_{2^\ell,i}\big) - f\big(\widetilde{X}_{2^{\ell-1},i}\big)\big)$$

for an independent family of random elements $(X_{2^\ell,i}, \widetilde{X}_{2^{\ell-1},i})$ with $\ell = 0, \dots, L$ and $i = 1, \dots, N_\ell$ such that

$$\big(X_{2^\ell,i}, \widetilde{X}_{2^{\ell-1},i}\big) \overset{\mathrm{d}}{=} \big(X_{2^\ell}, \widetilde{X}_{2^{\ell-1}}\big).$$

Note that the cost for one realization of $f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})$, clearly, depends on the number r_ℓ of calls to the random number respective random bit generator used by the respective (random bit) Euler-Maruyama scheme, namely

$$r_\ell = d \cdot \begin{cases} 2^\ell, & \text{for the classical Euler,} \\ 2^\ell \cdot q, & \text{for scheme 1,} \\ 2^{\ell+2} - 2, & \text{for scheme 2.} \end{cases} \tag{98}$$

Hence the cost bound (75) for the multilevel Euler algorithm is generalized to

$$\mathrm{cost}\big(A_{L,N}, F\big) \preccurlyeq \sum_{\ell=0}^{L} N_\ell \cdot r_\ell.$$

In the following, we will analyze the particular random bit multilevel algorithm $A^{q}_{L,N}$. The outline is to choose the same highest level L and the same replication numbers N as for the classical multilevel algorithm $A^{\star}_{L,N}$, see Section IV.3.1, and to choose the bit number q on top. This way, we can bound the error of $A^{q}_{L,N}$ from above by the error of $A^{\star}_{L,N}$ and the stochastic L_2-distance between the two schemes $A^{\star}_{L,N}$ and $A^{q}_{L,N}$.

This is, for $\varepsilon \in]0, 1/2[$ we consider the algorithm

$$A^{q}_{\varepsilon,F} = A^{q(\varepsilon,F)}_{L(\varepsilon,F),N(\varepsilon,F)}$$

with maximal level $L = L(\varepsilon, F)$ given by (76), replication numbers $N_\ell = N_\ell(\varepsilon, F)$ for $\ell = 0, \dots, L$ given by (77), and the bit number

$$q = q(\varepsilon, F) = L(\varepsilon, F). \tag{99}$$

THEOREM 11. *Let $F = F_\infty$ or $F = F_p$ with $1 \le p < \infty$. Then there exists a positive constant c such that the random bit multilevel Euler algorithm $A^q_{\varepsilon,F}$ satisfies*

$$\operatorname{error}\bigl(A^q_{\varepsilon,F}, F\bigr) \le c \cdot \varepsilon$$

and

$$\operatorname{cost}\bigl(A^q_{\varepsilon,F}, F\bigr) \le c \cdot \varepsilon^{-2} \cdot \begin{cases} \bigl(\ln(\varepsilon^{-1})\bigr)^3, & \text{if } F = F_p, \\ \bigl(\ln(\varepsilon^{-1})\bigr)^4, & \text{if } F = F_\infty \end{cases}$$

for every $\varepsilon \in]0, 1/2[$.

COROLLARY 5. *In terms of the ε-complexity $\operatorname{comp}^{\mathrm{res}}(\varepsilon, F)$, see Definition 6, Theorem 11 implies an upper bound that is of the same order as $\operatorname{cost}(A^q_{\varepsilon,F}, F)$, i.e., there exists a positive constant c such that*

$$\operatorname{comp}^{\mathrm{res}}\bigl(\varepsilon, F\bigr) \le c \cdot \varepsilon^{-2} \cdot \begin{cases} \bigl(\ln(\varepsilon^{-1})\bigr)^3, & \text{if } F = F_p, \\ \bigl(\ln(\varepsilon^{-1})\bigr)^4, & \text{if } F = F_\infty. \end{cases}$$

PROOF OF THEOREM 11. As in the proof of Theorem 9, we only present the case $F = F_\infty$. At first we establish the error bound. To this end, observe that

$$\operatorname{error}\bigl(A^q_{\varepsilon,F}, F\bigr) \le \sup_{f \in F} \Bigl(\mathrm{E}\bigl|S(f) - A_{\varepsilon,F}(f)\bigr|^2\Bigr)^{1/2} + \sup_{f \in F} \Bigl(\mathrm{E}\bigl|A_{\varepsilon,F}(f) - A^q_{\varepsilon,F}(f)\bigr|^2\Bigr)^{1/2}.$$

Hence, due to Theorem 9, it suffices to show the existence of a positive constant c such that

$$\sup_{f \in F} \Bigl(\mathrm{E}\bigl|A_{\varepsilon,F}(f) - A^q_{\varepsilon,F}(f)\bigr|^2\Bigr)^{1/2} \le c \cdot \varepsilon$$

for every $\varepsilon \in]0, 1/2[$. For each level $\ell = 0, \dots, L$, the $\triangle$-inequality gives

$$\begin{aligned} &\Bigl(\mathrm{E}\Bigl(\frac{1}{N_\ell} \cdot \sum_{i=1}^{N_\ell} f\bigl(X^\star_{2^\ell,i}\bigr) - f\bigl(X^{(q)}_{2^\ell,i}\bigr) - \bigl(f\bigl(\widetilde{X}^\star_{2^{\ell-1},i}\bigr) - \bigl(\widetilde{X}^{(q)}_{2^{\ell-1},i}\bigr)\bigr)\Bigr)^2\Bigr)^{1/2} \\ &\quad \le \frac{1}{N_\ell} \cdot \sum_{i=1}^{N_\ell} \Bigl(\mathrm{E}\Bigl(f\bigl(X^\star_{2^\ell,i}\bigr) - f\bigl(X^{(q)}_{2^\ell,i}\bigr) - \bigl(f\bigl(\widetilde{X}^\star_{2^{\ell-1},i}\bigr) - \bigl(\widetilde{X}^{(q)}_{2^{\ell-1},i}\bigr)\bigr)\Bigr)^2\Bigr)^{1/2} \\ &\quad \le \Bigl(\mathrm{E}\bigl|f\bigl(X^\star_{2^\ell}\bigr) - f\bigl(X^{(q)}_{2^\ell}\bigr)\bigr|^2\Bigr)^{1/2} + \Bigl(\mathrm{E}\bigl|f\bigl(\widetilde{X}^\star_{2^{\ell-1}}\bigr) - f\bigl(\widetilde{X}^{(q)}_{2^{\ell-1}}\bigr)\bigr|^2\Bigr)^{1/2}. \end{aligned}$$

Combining this estimate with the Lipschitz continuity of each $f \in F$ we get

$$\begin{aligned} &\Bigl(\mathrm{E}\bigl|A^\star_{L,N}(f) - A^q_{L,N}(f)\bigr|^2\Bigr)^{1/2} \\ &\qquad \le \sum_{\ell=0}^{L} \Bigl(\mathrm{E}\bigl\|X^\star_{2^\ell} - X^{(q)}_{2^\ell}\bigr\|^2_{\sup}\Bigr)^{1/2} + \sum_{\ell=1}^{L} \Bigl(\mathrm{E}\bigl\|\widetilde{X}^\star_{2^{\ell-1}} - \widetilde{X}^{(q)}_{2^{\ell-1}}\bigr\|^2_{\sup}\Bigr)^{1/2}. \end{aligned}$$

We apply Lemma 15 as well as Lemma 16 to obtain the existence of a positive constant c_1 such that

$$\Bigl(\mathrm{E}\bigl|A^\star_{L,N}(f) - A^q_{L,N}(f)\bigr|^2\Bigr)^{1/2} \le c_1 \cdot L \cdot 2^{-q/2} \cdot q^{-1/2}$$

for all L, q, and N and every $f \in F$. With the particular choice (99) for q this upper bound is of the order ε, as claimed.

To establish the cost bound it remains to observe, analogously to (78), that there exists a positive constant c_2 such that for the particular choice (99) of q we have

$$\sum_{\ell=0}^{L} N_\ell(\varepsilon, F) \cdot 2^\ell \cdot q \leq c_2 \cdot \varepsilon^{-2} \cdot L^4 \leq c_2 \cdot \varepsilon^{-2} \cdot \left(\log_2(\varepsilon^{-1})\right)^4$$

for all ε.

The proof for $F = F_p$ follows step by step the proof for $F = F_\infty$, using the corresponding results from Theorem 9. ∎

Remark 40. To classify the results form Theorem 11 and Corollary 5, recall that for a slightly stronger smoothness assumption and a non-degeneracy assumption on the diffusion coefficient b, we have the lower bound of order ε^{-2} for the ε-complexity for $F = F_p$ and $F = F_\infty$, see Remark 26. Moreover, comparing Corollary 5 to Corollary 4, we observe that concerning the weak asymptotic upper bound on the ε-complexity, we only lose one logarithmic factor in the transition from random numbers to random bits.

It is now a natural question whether we can improve the order of the cost bound in Theorem 11 and hence the ε-complexity in Corollary 5 at least by this logarithmic factor. To this end, observe that the computational bottleneck of $A^q_{\varepsilon,F}$ is the number of calls to the random bit generator. Hence we need to reduce the number of random bits used by $A^q_{\varepsilon,F}$, if we want to lower the order of the cost of $A^q_{\varepsilon,F}$. This can actually be done using a Bakhvalov trick, and will be discussed in Section IV.5.1.

We turn to the main problem of the random bit multilevel Euler algorithm $A^q_{\varepsilon,F}$, namely the à priori choice of the bit number q, see (99). At first, recall that in Remark 27 we already discussed the problem of the à priori choice of the highest level L, which has to be computed in advance, as well as the problem of using the upper bound on the rate of convergence of the Euler-Maruyama scheme, given in Theorem 2, for the classical multilevel Euler algorithm $A^\star_{\varepsilon,F}$. Those problems together with the problems for the numerical estimation of the bias and the variance decay, see Remark 30, let to the adaptive multilevel Euler algorithm $A^{\mathrm{adp}}_{\varepsilon,F}$, see Definition 9.

Consequently, we would like to use a random bit version of the adaptive multilevel Euler algorithm $A^{\mathrm{adp}}_{\varepsilon,F}$. The problem is that each random bit approximation of a Brownian increment $V^{(q)}_{k,2^\ell}$ with $k = 1, \ldots, 2^\ell$ and $\ell = 0, \ldots, L$, see (89), involves the same number $q = L$ of random bits. Hence each time the adaptive random bit multilevel Euler algorithm increases L to $L+1$, in order to establish weak convergence of $f(X_{2^\ell})$ to $f(X)$, we also increase the bit number q by one. That means that the whole simulation up to this point can not be reused, since the random bit Euler-Maruyama schemes $X^{(q+1)}_{2^\ell}$ and $X^{(q)}_{2^\ell}$ as well as $\widetilde{X}^{(q+1)}_{2^{\ell-1}}$ and $\widetilde{X}^{(q)}_{2^{\ell-1}}$, respectively, do not share the same distribution. That actually makes the whole idea of the adaptive multilevel Euler algorithm ridiculous, since the highest level always has to be precomputed, implying additional cost for a practical application of the random bit multilevel Euler algorithm $A^q_{\varepsilon,F}$.

Remark 41. The afore discussed problems lead us to the multilevel Euler algorithm $A^\dagger_{L,N}$. Here, the number of bits used on level $\ell = 0, \ldots, L$, i.e., used by $X^\dagger_{2^\ell}$ and $\widetilde{X}^\dagger_{2^{\ell-1}}$ depends only on the current level ℓ and is in particular independent from the choice of the highest level L. Hence $A^\dagger_{L,N}$ can be modified to an adaptive random bit multilevel Euler algorithm,

denoted by $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, in a straightforward way, namely we only exchange X_{2^ℓ} and $\widetilde{X}_{2^{\ell-1}}$ by $X^\dagger_{2^\ell}$ and $\widetilde{X}^\dagger_{2^{\ell-1}}$ in Definition 9, i.e., in the definition of $A^{\mathrm{adp}}_{\varepsilon,F}$, respectively. The algorithm $A^{\dagger,\mathrm{adp}^\star}_{\varepsilon,F}$ is defined analogously.

Unfortunately, so far, there is no analysis for $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ available. The problem lies in the strong error analysis of Scheme 2, i.e., of $X^\dagger_m$ and $\widetilde{X}^\dagger_{m/2}$, respectively. More precisely, whilst for Scheme 1 we could proceed similar to the analysis of the classical Euler-Maruyama scheme, this is no longer possible, since we no longer have independence of the increments $V^\dagger_{1,m},\dots,V^\dagger_{m,m}$ and $\widetilde{V}^\dagger_{1,m/2},\widetilde{V}^\dagger_{m/2,m/2}$, respectively. This is actually a consequence of the Lévy-Ciesielski construction involving bit approximations of the elements of the sequence of standard normally distributed random variables $Y_{0,1}, Y_{1,1},\dots$, see Remark 2.

However, the consideration of $A^\dagger_{L,N}$ is justified via its extension to the adaptive multilevel Euler algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ on the on hand, and the fact that we use the asymptotically optimal numbers of random bits for the random bit approximation of the distribution of a Brownian bridge in $L_2([0,1])$, see Theorem 5 and Corollary 1 in Section III.2.2, on the other hand, suggesting that at least on the level of constants, $A^\dagger_{L,N}$ might perform better than $A^q_{L,N}$. For numerical studies of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ as well its numerical comparison to $A^q_{L,N}$ see Chapter V.

IV.5.1. A Random Bit MLMC Algorithm Based on Bakhvalov's Trick. In this section we consider a variant of the random bit multilevel Euler algorithm $A^q_{L,N}$ from Section IV.5, which is based on Bakhvalov's trick, see Bakhvalov [2] and also Heinrich et al. [36] and the references therein. In our case this trick yields n^2 pairwise independent random variables, each of which is uniformly distributed on $D^{(q)}$, from $2\cdot n$ independent random variables, each of which is uniformly distributed on $D^{(q)}$, itself. Recall that a uniform distribution on $D^{(q)}$ can be simulated by q random bits, see Section III.2.1.

LEMMA 17. *Let $q\in\mathbb{N}$ be a bit number and let $n\in\mathbb{N}$. Consider an independent family $(G_j)_{j=1,\dots,2\cdot n}$ of random variables that are uniformly distributed on the set of shifted dyadic numbers $D^{(q)}=\{k\cdot 2^{-q}+2^{-(q+1)}\colon k=0,\dots,2^q-1\}$. Then the family $(G_{j_1}+G_{j_2+n}+2^{-(q+1)} \bmod 1)_{j_1,j_2=1,\dots,n}$ is pairwise independent, with each random variable being uniformly distributed on $D^{(q)}$.*

PROOF. Let G be uniformly distributed on $D^{(q)}$, denoted by $G\sim\mathrm{unif}(D^{(q)})$, i.e.,

$$P\big(\{G=k\cdot 2^{-q}+2^{-(q+1)}\}\big)=2^{-q}$$

for all $k=0,\dots,2^q-1$. Then for all $z\in\mathbb{N}(q)=\{i\cdot 2^{-q}\colon i\in\mathbb{N}\}$ we have

$$z+G \mod 1\sim\mathrm{unif}\big(D^{(q)}\big), \tag{100}$$

since for $z=i\cdot 2^{-q}$ it holds

$$\begin{aligned}P\big(\{G+z \bmod 1=k\cdot 2^{-q}+2^{-(q+1)}\}\big)&\\ =P\big(\{G=\big((k-i)\cdot 2^{-q} \bmod 1\big)+2^{-(q+1)}\}\big)&=2^{-q}\end{aligned}$$

for all $k=0,\dots,2^q-1$. Furthermore,

$$G+2^{-(q+1)}\sim\mathrm{unif}\big(D^{(q)}+2^{-(q+1)}\big)$$

with $D^{(q)}+2^{-(q+1)}=\{k\cdot 2^{-1}\colon k=1,\dots,2^q\}\subseteq\mathbb{N}(q)$. Hence, we in particular have $G+2^{-(q+1)}\in\mathbb{N}(q)$, implying $G_{j_2+n}+2^{-(q+1)}\in\mathbb{N}(q)$ and hence

$$H_{j_1,j_2}=\big(G_{j_1}+G_{j_2+n}+2^{-(q+1)} \bmod 1\big)\in D^{(q)} \tag{101}$$

for all $j_2 = 1, \dots, n$.

For notational convenience we set

$$\{G_{j_1}(i,q)\} = \{G_{j_1} = i \cdot 2^{-q} + 2^{-(q+1)}\} \tag{102}$$

for all $i \in \{0, \dots, 2^q - 1\}$. Due to the independence of G_{j_1} and G_{j_2+n} we have for all k, i from $\{0, \dots, 2^q - 1\}$

$$\begin{aligned} &P\big(\{(H_{j_1,j_2} | G_{j_1}(i,q)) = k \cdot 2^{-q} + 2^{-(q+1)}\}\big) \\ &\quad = \frac{P\big(\{((i+1) \cdot 2^{-q} + G_{j_2+n} \mod 1) = k \cdot 2^{-q} + 2^{-(q+1)}\} \cap \{G_{j_1}(i,q)\}\big)}{P\big(\{G_{j_1}(i,q)\}\big)} \\ &\quad = P\big(\{((i+1) \cdot 2^{-q} + G_{j_2+n} \mod 1) = k \cdot 2^{-q} + 2^{-(q+1)}\}\big) = 2^{-q}, \end{aligned}$$

where the last equality is due to (100). Hence the random variable H_{j_1,j_2} conditioned on G_{j_1} is uniformly distributed on $D^{(q)}$. As a consequence, using the law of total probability, we get

$$\begin{aligned} &P\big(\{H_{j_1,j_2} = k \cdot 2^{-q} + 2^{-(q+1)}\}\big) \\ &\qquad = \sum_{i=0}^{2^q-1} P\big(\{(H_{j_1,j_2} = k \cdot 2^{-q} + 2^{-(q+1)}) | G_{j_1}(i,q)\}\big) \cdot P\big(\{G_{j_1}(i,q)\}\big) = 2^{-q} \end{aligned}$$

for every $k \in \{0, \dots, 2^q - 1\}$, i.e.,

$$G_{j_1} + G_{j_2+n} + 2^{-(q+1)} \mod 1 \sim \mathrm{unif}\big(D^{(q)}\big).$$

Let $l_1, l_2 \in \{1, \dots, n\}$. It remains to show that H_{l_1,l_2} and H_{j_1,j_2} given by (101) are independent for $(l_1, l_2) \neq (j_1, j_2)$. The cases $l_1 \neq j_1$ and $l_2 \neq j_2$ are trivial, due to the independence of the family $(G_j)_{j=1,\dots,2\cdot n}$. Therefore, we can assume without loss of generality that $l_1 = j_1$ and $l_2 \neq j_2$. For every choice of $k, \widetilde{k} \in \{0, \dots, 2^q - 1\}$ we have, using the notation (102),

$$\begin{aligned} &P\big(\{(H_{l_1,l_2} | G_{j_1}(i,q)) = k \cdot 2^{-q} + 2^{-(q+1)}\} \cap \{(H_{j_1,j_2} | G_{j_1}(i,q)) = \widetilde{k} \cdot 2^{-q} + 2^{-(q+1)}\}\big) \\ &= P\big(\{G_{l_2+n} + 2^{-(q+1)} = (k-i) \cdot 2^{-q}\} \cap \{G_{j_2+n} + 2^{-(q+1)} = (\widetilde{k} - i) \cdot 2^{-q}\}\big) \\ &= P\big(\{G_{l_2+n} + 2^{-(q+1)} = (k-i) \cdot 2^{-q}\}\big) \cdot P\big(\{G_{j_2+n} + 2^{-(q+1)} = (\widetilde{k} - i) \cdot 2^{-q}\}\big) \\ &= P\big(\{(H_{l_1,l_2} | G_{j_1}(i,q)) = k \cdot 2^{-q} + 2^{-(q+1)}\}\big) \cdot P\big(\{(H_{j_1,j_2} | G_{j_1}(i,q)) = \widetilde{k} \cdot 2^{-q} + 2^{-(q+1)}\}\big) \end{aligned}$$

for all $i \in \{0, \dots, 2^q - 1\}$. Again we can apply the law of total probability to obtain the independence of H_{l_1,l_2} and H_{j_1,j_2}. ■

Of course, the analogon to Lemma 17 is true for the d-dimensional case, with independent components, i.e., with $D^{(q)}$ replaced by $(D^{(q)})^d$.

We construct a random bit multilevel Euler algorithm $A^{q,\mathrm{Bak}}_{L,N}$ that achieves the following properties at a reduced number of random bits compared to $A^q_{L,N}$, described in Section IV.5:

(i) The family $\big(X^{(q),\mathrm{Bak}}_{2^\ell,i}, \widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1},i}\big)_{\ell=0,\dots,L,\, i=1,\dots,N_\ell}$ of random bit Euler-Maruyama schemes involved by $A^{q,\mathrm{Bak}}_{L,N}$ is pairwise independent.

(ii) On every level $\ell = 0, \dots, L$ we have

$$\big(X^{(q),\mathrm{Bak}}_{2^\ell}, \widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1}}\big) \overset{\mathrm{d}}{=} \big(X^{(q)}_{2^\ell}, \widetilde{X}^{(q)}_{2^{\ell-1}}\big).$$

Note that for a family $(Z_k)_{k=1,\dots,n}$ of random variables the variance of the sum satisfies

$$\operatorname{Var}\Big(\sum_{k=1}^{n} Z_k\Big) = \sum_{k=1}^{n}\big(\mathrm{E}\big(Z_k^2\big) - (\mathrm{E}(Z_k))^2\big) + 2\cdot\sum_{k=1}^{n}\sum_{\substack{j=1,\\ j\neq k}}^{n}\big(\mathrm{E}(Z_k\cdot Z_j) - \mathrm{E}(Z_k)\cdot\mathrm{E}(Z_j)\big).$$

Due to this property and the linearity of the expectation, we conclude from (i) and (ii) that the expectation and the variance of $A_{L,N}^{q,\mathrm{Bak}}(f)$ and $A_{L,N}^{q}(f)$ do coincide for every $f\in F$. Since the mean-squared error of any Monte Carlo algorithm can be decomposed into the squared bias and the variance of this algorithm, see Remark 21, we obtain $\operatorname{error}(A_{L,N}^{q,\mathrm{Bak}}, f) = \operatorname{error}(A_{L,N}^{q}, f)$ for every $f\in F$ and hence

$$\operatorname{error}\big(A_{L,N}^{q,\mathrm{Bak}}, F\big) = \operatorname{error}\big(A_{L,N}^{q}, F\big). \tag{103}$$

We describe the construction of $A_{L,N}^{q,\mathrm{Bak}}$ or, equivalently, the distribution of the family $\big(X_{2^\ell,i}^{(q),\mathrm{Bak}}, \widetilde{X}_{2^{\ell-1},i}^{(q),\mathrm{Bak}}\big)_{\ell=0,\dots,L,\, i=1,\dots,N_\ell}$ in detail. Let

$$n_\ell = \Big\lceil N_\ell^{1/2}\Big\rceil$$

for $\ell = 0,\dots,L$. We consider an independent family

$$\mathbf{G} = \big((G_{k,2^\ell})_j\big) = \big(G_{k,2^\ell,j}\big)_{k=1,\dots,2^\ell,\,\ell=0,\dots,L,\,j=1,\dots,2\cdot n_\ell}$$

of random vectors $G_{k,2^\ell,j}$ that are uniformly distributed on $(D^{(q)})^d$, i.e., for fixed k and ℓ we have the d-dimensional counterpart to the family $(G_j)_{j=1,\dots,2\cdot n}$ in Lemma 17. In the following, let $\mathbf{1}\in\mathbb{R}^d$ denote the vector with all components equal to 1. For $\ell = 0,\dots,L$ and $k = 1,\dots,2^\ell$ we construct via

$$\big(U_{k,2^\ell}\big)_{(j_1-1)\cdot n_\ell + j_2} = \big(G_{k,2^\ell}\big)_{j_1} + \big(G_{k,2^\ell}\big)_{j_2+n_\ell} + 2^{-(q+1)}\cdot\mathbf{1} \mod 1, \tag{104}$$

where $j_1, j_2 = 1,\dots,n_\ell$, families $\big(U_{k,2^\ell}\big)_{i=1,\dots,n_\ell^2}$ of n_ℓ^2 pairwise independent random variables, each of which is uniformly distributed on $(D^{(q)})^d$, cf. Lemma 17. Since in general $n_\ell^2 = \lceil N_\ell^{1/2}\rceil^2 > N_\ell$, we define for $\ell = 0,\dots,L$ the families

$$\mathbf{U}_\ell = \big(U_{k,2^\ell,i}\big)_{k=1,\dots,2^\ell\, i=1,\dots,N_\ell}.$$

Before we continue with the construction of $A_{L,N}^{q,\mathrm{Bak}}$, we gather some properties of $(\mathbf{U}_\ell)_{\ell=0,\dots,L}$. To this end, we define

$$\mathbf{U}_{\ell,i} = \big(U_{k,2^\ell,i}\big)_{k=1,\dots,2^\ell}$$

for $\ell = 0,\dots,L$ and $i = 1,\dots,N_\ell$, i.e., $(\mathbf{U}_{\ell,i})_{i=1,\dots,N_\ell}$ is the family of random variables involved on level $\ell = 0,\dots,L$. The following holds true:

(I) The family $(\mathbf{U}_\ell)_{\ell=0,\dots,L}$ is independent, which, as we will see, corresponds to the fact that a multilevel Euler algorithm uses independent direct simulations on each level $\ell = 0,\dots,L$, see Section IV.2.

(II) For all $\ell = 0,\dots,L$ and $i = 1,\dots,N_\ell$ the family $(U_{k,2^\ell,i})_{k=1,\dots,2^\ell}$ is independent. This property will lead to independent Brownian increments for each sample path on level ℓ.

(III) Recall that, by Lemma 17, for every level $\ell = 0,\dots,L$ the family $(\mathbf{U}_{\ell,i})_{i=1,\dots,N_\ell}$ is pairwise independent. This will allow us to apply Bienaymé's formula to each level ℓ.

(IV) Recall that, by Lemma 17, for all $k = 1, \dots, 2^\ell$, $\ell = 0, \dots, L$ and $i = 1, \dots, N_\ell$ the random vector $U_{k,2^\ell,i}$ is uniformly distributed on $(D^{(q)})^d$.

We proceed with the construction of $A^{q,\mathrm{Bak}}_{L,N}$, namely with the bit approximations of the Brownian increments. For $k = 1, \dots, 2^\ell$, $\ell = 0, \dots, L$ and $i = 1, \dots, N_\ell$ let

$$V_{k,2^\ell,i} = 2^{-\ell/2} \cdot \Phi^{-1}\big(U_{k,2^\ell,i}\big),$$

where the function Φ^{-1} is applied to each of the components of the d-dimensional random vector $U_{k,2^\ell,i}$, separately, cf. (88). The bit approximations of the increments of the coupled coarse bit approximation of the Brownian paths on the levels $\ell = 1, \dots, L$ are obtained via

$$\widetilde{V}_{k,2^{\ell-1},i} = V_{2\cdot k-1,2^\ell,i} + V_{2\cdot k,2^\ell,i}$$

for $k = 1, \dots, 2^{\ell-1}$ and $i = 1, \dots, N_\ell$, cf. (89). Observe that the independence properties (I) – (III) carry over to the bit approximations $V_{k,2^\ell,i}$ and $\widetilde{V}_{k,2^{\ell-1},i}$ of the Brownian increments. Finally, for $\ell = 0, \dots, L$ and $i = 1, \dots, N_\ell$ the bit approximation $X^{(q),\mathrm{Bak}}_{2^\ell,i}$ of the classical Euler-Maruyama scheme $X^\star_{2^\ell}$, see Section II.2.1, is obtained via replacing the (classical) Brownian increments $(V^\star_{k,2^\ell})_{k=1,\dots,2^\ell}$ by the bit approximations $(V_{k,2^\ell,i})_{k=1,\dots,2^\ell}$. Likewise, for $\ell = 1, \dots, L$ and $i = 1, \dots, N_\ell$ the bit approximation $\widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1},i}$ of $\widetilde{X}^\star_{2^{\ell-1}}$ is obtained via replacing $(\widetilde{V}^\star_{k,2^{\ell-1}})_{k=1,\dots,2^{\ell-1}}$ by $(\widetilde{V}_{k,2^{\ell-1},i})_{k=1,\dots,2^{\ell-1}}$, and for $\ell = 0$ and $i = 1, \dots, N_0$ we set $\widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1},i} = 0$.

Observe that for $\ell = 0, \dots, L$ and $i = 1, \dots, N_\ell$ we have

$$\big(V_{k,2^\ell,i}\big)_{k=1,\dots,2^\ell} \overset{\mathrm{d}}{=} \big(V^{(q)}_{k,2^\ell}\big)_{k=1,\dots,2^\ell},$$

which follows from the independence of $(V_{k,2^\ell,i})_{k=1,\dots,2^\ell}$ that is implied by (II), and due to $V_{k,2^\ell,i}$ and $V^{(q)}_{k,2^\ell}$ sharing the same distribution for each $k = 1, \dots, 2^\ell$. The later is a consequence of (IV) and $V^{(q)}_{k,2^\ell} \overset{\mathrm{d}}{=} 2^{-\ell/2} \cdot \Phi^{-1}(U)$ with U uniformly distributed on $(D^{(q)})^d$, see (87) and the discussion thereafter. With the same argument we have

$$\big(\widetilde{V}_{k,2^{\ell-1},i}\big)_{k=1,\dots,2^{\ell-1}} \overset{\mathrm{d}}{=} \big(\widetilde{V}^{(q)}_{k,2^{\ell-1}}\big)_{k=1,\dots,2^{\ell-1}}$$

for $\ell = 1, \dots, L$ and $i = 1, \dots, N_\ell$. Since the Euler-Maruyama schemes, X_{2^ℓ} for $\ell \in N_0$ and $\widetilde{X}_{2^{\ell-1}}$ for $\ell \in \mathbb{N}$, can be interpreted as deterministic functions applied to the sequences of 2^ℓ respective $2^{\ell-1}$ Brownian increments or their respective approximations, we get

$$\big(X^{(q),\mathrm{Bak}}_{2^\ell,i}, \widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1},i}\big) \overset{\mathrm{d}}{=} \big(X^{(q)}_{2^\ell}, \widetilde{X}^{(q)}_{2^{\ell-1}}\big)$$

for $\ell = 0, \dots, L$ and $i = 1, \dots, L$, i.e., property (ii). For property (i) it remains to observe that (I) implies the level-wise independence of the family

$$\big(X^{(q),\mathrm{Bak}}_{2^\ell,i}, \widetilde{X}^{(q),\mathrm{Bak}}_{2^{\ell-1},i}\big)_{\ell=0,\dots,L\, i=1,\dots,N_\ell}.$$

Moreover, property (III) implies for each level $\ell = 0, \dots, L$ the pairwise independence of the realizations $i = 1, \dots, N_\ell$.

The cost of $A^{q,\mathrm{Bak}}_{L,N}$ can be bounded similar as in Section IV.3.1 and Section IV.5. First of all, we note that once we have the uniform distributions on $D^{(q)}$ at hand, the algorithms $A^\star_{L,N}$, $A^q_{L,N}$ and $A^{q,\mathrm{Bak}}_{L,N}$ proceed in the same (deterministic) way. Hence from that point on

the cost of $A_{L,N}^{q,\mathrm{Bak}}$ is bounded by

$$\sum_{\ell=0}^{L} N_\ell \cdot 2^\ell,$$

up to a multiplicative constant. Next, we observe that the uniform distributions on $(D^{(q)})^d$, i.e., the family $(\mathbf{U}_\ell)_{\ell=0,\dots,L}$ can be computed by a separate routine. The number of random bits needed to simulate N_ℓ paths on level $\ell = 0, \dots, L$ is given by $2 \cdot n_\ell \cdot 2^\ell \cdot q \cdot d$ and hence the total number of calls to the random bit generator performed by $A_{L,N}^{q,\mathrm{Bak}}$ is given by

$$\sum_{\ell=0}^{L} 2 \cdot n_\ell \cdot 2^\ell \cdot q \cdot d.$$

In order to obtain a uniform distribution on $D^{(q)}$ from q random bits, we have to carry out $(2 \cdot q + 1)$ arithmetic operations, cf. definition of $D^{(q)}$. Since we need to do this $2 \cdot n_\ell \cdot 2^\ell \cdot d$ times, we again have the cost bound

$$\sum_{\ell=0}^{L} 2 \cdot n_\ell \cdot 2^\ell \cdot q \cdot d,$$

up to a multiplicative constant. Finally, observe that the construction of $(\mathbf{U}_\ell)_{\ell=0,\dots,L}$ requires

$$\sum_{\ell=0}^{L} N_\ell \cdot 2^\ell$$

calls to (104), which has constant cost itself.

Altogether, we conclude that there exists a positive constant c such that

$$\mathrm{cost}\big(A_{L,N}^{q,\mathrm{Bak}}, F\big) \le c \cdot \sum_{\ell=0}^{L} \big(N_\ell \cdot 2^\ell + n_\ell \cdot 2^\ell \cdot q\big).$$

Like in the cases of $A_{L,N}^{\star}$ and $A_{L,N}^{q}$, for $\varepsilon \in]0, 1/2[$ we consider the algorithm

$$A_{\varepsilon,F}^{q,\mathrm{Bak}} = A_{L(\varepsilon,F),N(\varepsilon,F)}^{q(\varepsilon,F),\mathrm{Bak}}$$

with maximal level $L = L(\varepsilon, F)$ given by (76), replication numbers $N_\ell = N_\ell(\varepsilon, F)$ for $\ell = 0, \dots, L$ given by (77), and the bit number $q = q(\varepsilon, F)$ given by (99).

THEOREM 12. *Let $F = F_\infty$ or $F = F_p$ with $1 \le p < \infty$. Then there exists a positive constant c such that the random bit multilevel Euler algorithm $A_{\varepsilon,F}^{q,\mathrm{Bak}}$ satisfies*

$$\mathrm{error}\big(A_{\varepsilon,F}^{q,\mathrm{Bak}}, F\big) \le c \cdot \varepsilon$$

and

$$\mathrm{cost}\big(A_{\varepsilon,F}^{q,\mathrm{Bak}}, F\big) \le c \cdot \varepsilon^{-2} \cdot \begin{cases} \big(\ln(\varepsilon^{-1})\big)^2, & \text{if } F = F_p, \\ \big(\ln(\varepsilon^{-1})\big)^3, & \text{if } F = F_\infty \end{cases}$$

for every $\varepsilon \in]0, 1/2[$.

PROOF. As in the proofs of Theorem 9 and Theorem 11, we only present the case $F = F_\infty$.

The error bound is an immediate consequence of $\mathrm{error}(A_{\varepsilon,F}^{q,\mathrm{Bak}}) = \mathrm{error}(A_{\varepsilon,F}^{q})$, which is due to (103), and Theorem 11, where $A_{\varepsilon,F}^{q}$ is analyzed.

The cost bound follows from the bound on $\sum_{\ell=0}^{L} N_\ell \cdot 2^\ell$ in (78) and the following observation. Let $n_\ell = \lceil N_\ell^{1/2} \rceil$. Then there exist positive constants c, c_1, c_2, and c_3 such that

$$\sum_{\ell=0}^{L} n_\ell \cdot 2^\ell \cdot q \le c \cdot L \cdot \sum_{\ell=0}^{L} N_\ell^{1/2} \cdot 2^\ell \le c_1 \cdot \varepsilon^{-1} \cdot L^{3/2} \cdot \sum_{\ell=0}^{L} 2^{L/2} \cdot \max(\ell, 1)^{1/2}$$
$$\le c_2 \cdot \varepsilon^{-2} \cdot L^{5/2} \cdot \big(\ln(\varepsilon^{-1})\big)^{1/2} \le c_3 \cdot \varepsilon^{-2} \cdot \big(\ln(\varepsilon^{-1})\big)^3.$$

The proof for $F = F_p$ follows step by step the proof for $F = F_\infty$. ■

We conclude that up to a multiplicative constant the error and the cost bounds for the algorithm $A^{q,\mathrm{Bak}}_{\varepsilon,F}$, which solely uses random bits, and the classical multilevel Euler algorithm $A^{\star}_{\varepsilon,F}$ that has access to random numbers from $[0,1]$, do coincide. Hence, in particular, the problems of $A^{q}_{\varepsilon,F}$, discussed in Remark 40 are settled.

CHAPTER V

Numerical Experiments for Random Bit Quadrature

In this section, we numerically investigate the approximation of $\mathrm{E}(f(X))$ by means of the random bit multilevel Euler algorithms $A^q_{\varepsilon,F}$ and $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, for X being the solution of different SDEs of the form

$$\begin{aligned} \mathrm{d}X(t) &= a(X(t))\,\mathrm{d}t + b(X(t))\,\mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= x_0, \end{aligned} \tag{105}$$

and for different functionals $f \in F$.

The main purpose is to compare the two algorithms, for these particular examples, in terms of their respective computational cost in dependence on ε to achieve an error below a given error bound ϵ, respectively. Recall that we have no analysis for $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ but, similarly as in the random number setting, we expect that the algorithm performs better than $A^q_{\varepsilon,F}$, which is analyzed in Section IV.5, cf. also Remark 41. Observe that $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ has the clear advantage of adaptivity, which in particular allows an optimized choice of the highest level $L(\varepsilon,f)$ and the replication numbers $N(\varepsilon,f)$ in the case that the sequence $(X_{2^\ell})_{\ell\in\mathbb{N}_0}$ of Euler-Maruyama schemes, and hence the sequence $(f(X_{2^\ell}))_{\ell\in\mathbb{N}_0}$, has a better order of convergence than the upper bound given in Theorem 2, on which $A^q_{\varepsilon,F}$ is based. Hence we consider examples for which this effect is supposed to kick in as well as examples for which this should not be the case.

However, before we compare the algorithms, we will take a closer look at each one separately, starting with $A^q_{\varepsilon,F}$. Here, we have two aims. One is to fortify the upper bound on the order of the computational cost of $A^q_{\varepsilon,F}$ given in Theorem 11. The other one is to numerically analyze the upper bound on the ϵ-complexity $\mathrm{comp}^{\mathrm{res}}(\epsilon,F)$ induced by $A^q_{\varepsilon,F}(f)$ for the particular choices of the SDE and the functional f. In this context, recall that Theorem 11 yields only the asymptotic order of the computational cost of $A^q_{\varepsilon,F}(f)$ and hence solely an upper bound on the order of the ϵ-complexity and not the ϵ-complexity itself. In particular, so far, we do not know how to choose the parameters $L(\varepsilon,F)$ and $N(\varepsilon,F)$, i.e., how to choose the input ε to obtain an error below a given error bound ϵ. Consequently, we will illustrate the relation between the input ε and the error of $A^q_{\varepsilon,F}(f)$ in a figure, where we plot them against each other for a particular sequence of input parameters ε. Finally, we illustrate the effect of the random bit number q on the approximations of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ on the first few coupled levels $\ell = 1,\dots,L'$. That is, we compare empirical approximations of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ to the corresponding empirical approximations of $|\operatorname{Bias}^{(q)}_\ell(f)| = |\mathrm{E}(f(X^{(q)}_{2^\ell}) - f(\widetilde{X}^{(q)}_{2^{\ell-1}}))|$ and $\operatorname{Var}^{(q)}_\ell(f) = \operatorname{Var}(f(X^{(q)}_{2^\ell}) - f(\widetilde{X}^{(q)}_{2^{\ell-1}}))$, respectively, for particular choices of q to be specified later.

For the second algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ using an equal split of the bias and variance contribution to the error, i.e., $\varepsilon_B^2 = \varepsilon_V^2 = \varepsilon^2/2$, we proceed in a similar way. At first we compare the error and the computational cost of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$, i.e., we numerically analyze the upper bound on $\mathrm{comp}^{\mathrm{res}}(\epsilon,F)$ implied by $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ for the particular SDEs and functionals $f \in F$. The

next quantity of interest is the error of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ in dependence on the input ε, which we again compute for a particular sequence of input parameters ε, against which we plot the corresponding errors for illustrational purposes. In this context, recall that the adaptive multilevel Euler algorithm, denoted by $A^{\star,\mathrm{adp}}_{\varepsilon,F}$, and consequently also $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ is constructed in a way such that it should have an error bounded by or at least close to the input ε. That is, we should either have $\mathrm{error}(A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)) \leq \varepsilon$ or the relative error $(\mathrm{error}(A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)) - \varepsilon)/\varepsilon$ should be rather small. This is the reason why we will use the notion of an *input accuracy* for the input ε. Furthermore, in Section IV.3.3, namely in Remark 37, we have shown that for $f \in F_\infty$ the condition numbers $\kappa(D^T D)$ for the design matrix D in the linear least square fits for the estimation of the parameters of the laws of $(|\operatorname{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\operatorname{Var}_\ell(f))_{\ell\in\mathbb{N}}$, which are supposed to be of the form $c \cdot (2^{-\gamma_1})^\ell \cdot \ell^{\gamma_2}$ for positive constants c, γ_1 and γ_2, see (84) and (85), are rather large, see Figure 1. Especially, they are way larger than the corresponding condition numbers when supposing the law $c \cdot (2^{-\gamma})^\ell$ for positive constants c and γ, see Figure 2, and Remark 37 for the general discussion. Hence we also compare $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ to the adaptive multilevel Euler algorithm $A^{\dagger,\mathrm{adp}^\star}_{\varepsilon,F}(f)$, which uses the split $\varepsilon_B^2 = \varepsilon^2/4$ and $\varepsilon_V^2 = 3/4 \cdot \varepsilon^2$ for the contribution of the bias and the variance to the error of $A^{\dagger,\mathrm{adp}^\star}_{\varepsilon,F}$, respectively. This split corresponds to the split used in programs provided at Giles [26]. Finally, we again illustrate the effect of the random bit approximations on the empirical estimates of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ on the first few levels $\ell = 1, \ldots, L'$. That is, we compare empirical approximations of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ to the corresponding empirical approximations of $|\operatorname{Bias}^\dagger_\ell(f)| = |\operatorname{E}(f(X^\dagger_{2^\ell}) - f(\widetilde{X}^\dagger_{2^{\ell-1}}))|$ and $\operatorname{Var}^\dagger_\ell(f) = \operatorname{Var}(f(X^\dagger_{2^\ell}) - f(\widetilde{X}^\dagger_{2^{\ell-1}}))$, respectively. Note that these approximations also show whether the upper bound on the order of convergence of the Euler-Maruyama schemes in Theorem 2 yields a sharp bound on the order of $(|\operatorname{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\operatorname{Var}_\ell(f))_{\ell\in\mathbb{N}}$, respectively, i.e., whether $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ can exploit a higher order of convergence than $A^q_{\varepsilon,F}(f)$.

For each numerical experiment we do present asymptotic confidence intervals to show the reliability of the result. We will not discuss the underlying replication numbers each time. In general all confidence intervals associated to MLMC algorithms do rely on $2 \cdot 10^3$ to $3 \cdot 10^3$ independent realizations. The numerical experiments concerning the bias and variance decays do rely on 10^6 independent realizations on each level.

Let us finally mention that we do not compare the classical (singlelevel) Euler algorithm to the random bit multilevel Euler algorithms $A^q_{\varepsilon,F}(f)$ and $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. Instead we will solely compare the random bit multilevel Euler algorithms to their counterparts using random numbers from $[0,1]$. These algorithms have been thoroughly studied and compared to the classical Euler algorithm in the literature. For an overview on the subject see Giles [25] and references therein.

The chapter is outlined as follows. We consider four different examples having one thing in common, namely the solution $\operatorname{E}(f(X))$ is known analytically. Hence the error of the involved algorithms can easily be approximated empirically without the necessity of a master computation. That would be a computation using a very accurate approximation scheme $X_{2^{\widetilde{L}}}$ with $\widetilde{L} \gg L(\varepsilon, F)$ and very high replication numbers, i.e., a in general very time demanding computation. The first example will be discussed in a very detailed way, in particular the figures will be explained in much detail. In the three further examples we will not explain the figures in so much detail again, since we illustrate the same quantities, and also the discussions will be kept way more briefly, as long as the results do not substantially differ from those for the first example. To judge the reliability of any random quantity we will give

asymptotic confidence intervals to the confidence level 0.05 for every random quantity, as described in Section IV.2.2, see in particular Remark 25.

As a first elementary example, in Section V.1, we consider the standard one-dimensional Brownian motion and as a functional $f \in F_\infty$ we take the maximum, i.e., $f(x) = \max_{0\le t\le 1} x(t)$ for $x \in C([0,1])$. In the second example, presented in Section V.2, we take a closer look to a geometric Brownian motion, again with the maximum as functional f. Geometric Brownian motions, are, e.g., used to model stock prices in financial mathematics. The third example, considered in Section V.3, deals with the Ornstein-Uhlenbeck process and uses the functional f that evaluates at the final time point, i.e., $f(x) = x(1)$. The Ornstein-Uhlenbeck process is, e.g., used to model the dynamics in the Vasicek interest rate model, see, e.g., Shreve [61, Example 4.4.10]. Last but not least, in Section V.4, we consider the Cox-Ingersoll-Ross (CIR) process, again with the functional f that evaluates the process at the final time point. The CIR process was first introduced for modeling interest rates, and is used to model the instantaneous variance in the Heston model, a model from financial mathematics used to describe the time-dependent price and volatility of an asset, see, e.g., Shreve [61, Example 4.4.11].

V.1. Standard Brownian Motion

We consider a one-dimensional standard Brownian motion. In terms of our stochastic differential equation (105), that is, we choose the constant drift and diffusion coefficients $a = 0$ and $b = 1$, as well as, the initial value $x_0 = 0$, i.e.,

$$\begin{aligned} \mathrm{d}X(t) &= 1 \cdot \mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= 0. \end{aligned}$$

Moreover, we consider the functional $f(x) = \max_{0\le t\le 1} x(t)$ for $x \in C([0,1])$. Hence the current setting is covered by our theoretical results in Section IV.5.

Before we turn to the numerical results, we first compute $\mathrm{E}(f(X))$ analytically. To this end observe that the reflection principle for Brownian motion gives

$$\max_{0\le t\le 1} X(t) \overset{\mathrm{d}}{=} |X(1)|, \tag{106}$$

see Mörters and Peres [46, Theorem 2.21]. That is, $f(X)$ shares the same distribution as the absolute value of a standard normally distributed random variable. Hence we have

$$\begin{aligned} \mathrm{E}(f(X)) &= 2 \cdot \int_0^\infty y \cdot (2\,\pi)^{-1/2} \cdot \exp(-y^2/2)\,\mathrm{d}y = (2/\pi)^{1/2} \cdot \int_0^\infty y \cdot \exp(-y^2/2)\,\mathrm{d}y \\ &= (2/\pi)^{1/2} \cdot [-\exp(-y^2/2)]_0^\infty = (2/\pi)^{1/2}. \end{aligned}$$

Observe that with this result at hand it is straightforward to empirically estimate the error of a multilevel Euler (Monte Carlo) algorithm.

V.1.1. MLMC Quadrature Based on Scheme 1. In this section, we present numerical results for the non-adaptive random bit multilevel algorithm $A^q_{\varepsilon,F}(f)$. In the first figure, Figure 1, we present the relation between the error and the cost of $A^q_{\varepsilon,F}(f)$ as well as the relation of the error and the cost of $A^\star_{\varepsilon,F}$, in terms of the number of calls to the random bit respective the random number generator. For simplicity we will use the general notions of calls to the random generator and random generator calls. Moreover, we will use the notation ϵ for the error of the algorithms under consideration. The error is measured on the x-axis, decreasing from right to left and the number of random generator calls is measured on the

y-axis. In this context, recall that the number of calls to the random generator dominates the overall cost of each of the algorithms, respectively. Note that in this comparison the algorithm $A^{\star}_{\varepsilon,F}$ has clearly a lesser cost than $A^{q}_{\varepsilon,F}$ since each bit approximation $V^{(q)}_{k,2^\ell}$ of a Brownian increment $V^{\star}_{k,2^\ell}$ involves q random bits whilst the Brownian increment itself uses only one random number, which formally corresponds to a whole sequence of random bits.

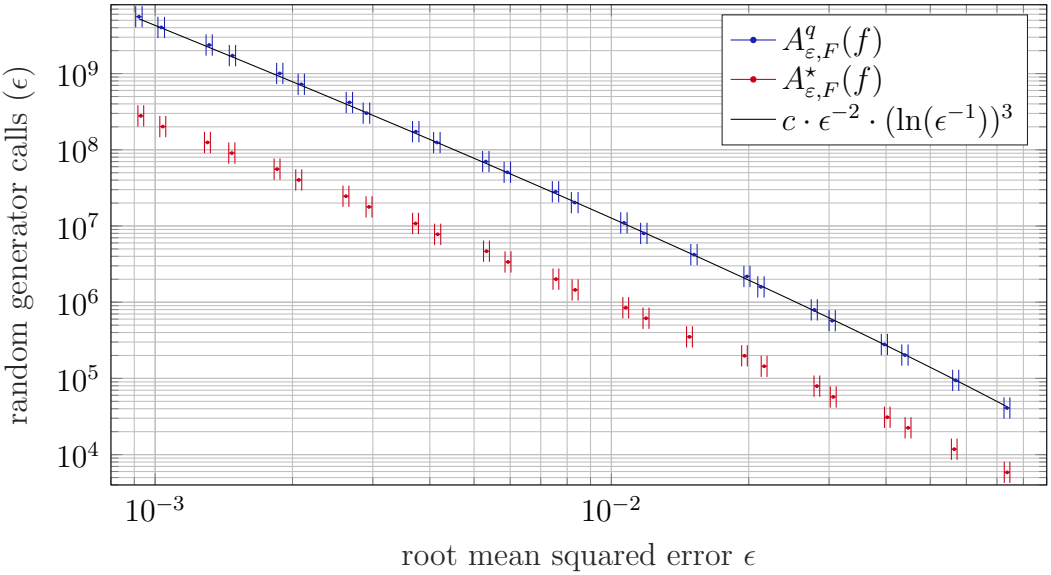

FIGURE 1. Error-cost relation of the non-adaptive MLMC Euler algorithms (random bits: blue, random numbers: red) with asymptotic confidence intervals to the confidence level 0.95. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

One last thing to be mentioned before we discuss the figure is the estimation of the asymptotic behaviour of the number of random generator calls. Similar to what we are doing in the estimates of the asymptotic behaviour of $(|\operatorname{Bias}_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\operatorname{Var}_\ell(f))_{\ell\in\mathbb{N}}$ for adaptive multilevel algorithms, see Remark 28 and Remark 29 as well as the discussion of the modifications of the adaptive multilevel algorithm in Section IV.3.3, it stands to reason to use linear least-square fits. Unfortunately, this turns out to be a too unstable method in the sense that a slightly perturbation in the data leads to a substantial impact on the exponent of the logarithmic factor, respectively. Recall that Theorem 11 suggests the law $c \cdot \epsilon^{-\gamma_1} \cdot (\ln(\epsilon^{-1}))^{\gamma_2}$ for positive constants c, γ_1 and γ_2. The problem is that working in a log-log-setting, the logarithmic factor is almost linear in ϵ^{-1} on the interval of interest. Hence the condition number $\kappa(D^T D)$ of the design matrix D is rather large and therefore, the solution is too affective to small perturbations, see Remark 43. For example choosing 25 equidistant points form $5 \cdot 10^{-2}$ to 10^{-3} leads to the condition number $3,26 \cdot 10^4$. Of course one could ask, whether 10^{-3} is small enough and whether the condition number would significantly decrease for smaller error bounds. The answer is no, taking again 25 equidistant points from $5 \cdot 10^{-2}$ to 10^{-4} and from $5 \cdot 10^{-2}$ to 10^{-6} we have the condition numbers $1.05 \cdot 10^4$ and $4,3 \cdot 10^3$, respectively. For comparison see also Remark 37, where a similar effect for the bias and variance estimations is discussed. If one really wants to decrease the condition numbers, one has to consider significantly bigger values of ϵ. But then the algorithms will no longer give reliable results, since the replication numbers get too small for good estimations of the variance and also the asymptotic estimates used for the choice of the highest level L and

the replication numbers N might no longer be valid. As a consequence we plot the function $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^{\gamma}$ with a parameter choice for c and γ that seems to fit the empirical data.

We discuss Figure 1. First of all we observe that we have relatively small confidence intervals. Hence we assume our numerical data, i.e., the empirical estimations of the error ϵ of the algorithms under consideration, to be reliable. Furthermore, the number of generator calls and hence the cost of $A^q_{\varepsilon,F}(f)$ seems to obey the law $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^3$, which is indeed one logarithmic order better than the upper bound from Theorem 1. However, this suggests that the upper bound from Theorem 1 is not sharp for the particular case of the standard Brownian motion and its maximum. Moreover, the comparison to $A^\star_{\varepsilon,F}(f)$ shows that the asymptotic behaviour of both the random bit and the random number algorithm seem to be quite similar, up to a multiplicative constant. Regarding this constant, recall that the random bit algorithm $A^q_{\varepsilon,F}(f)$ is, as discussed earlier in this section, punished by the factor $q = q(\varepsilon, F) = L(\varepsilon, F)$. Indeed the factor between the random generator calls of both algorithms is only slightly larger than q. This can be seen combining the definition of $L(\varepsilon, F)$, see (81), and the relation between the input parameter ε and the corresponding error ϵ of $A^\star_{\varepsilon,F}(f)$, given in Figure 2.

We consider Figure 2 in more detail. First of all, we again observe that the asymptotic confidence intervals are sufficiently small to justify working with the empirical values for the error ϵ of $A^q_{\varepsilon,F}(f)$. The figure illustrates the relation between the input parameter ε, which determines the whole algorithm $A^q_{\varepsilon,F}(f)$, and the empirical error of $A^q_{\varepsilon,F}(f)$. To improve the readability of the figure it also contains the identity function $\mathrm{Id}\colon \mathbb{R} \to \mathbb{R}, x \mapsto x$.

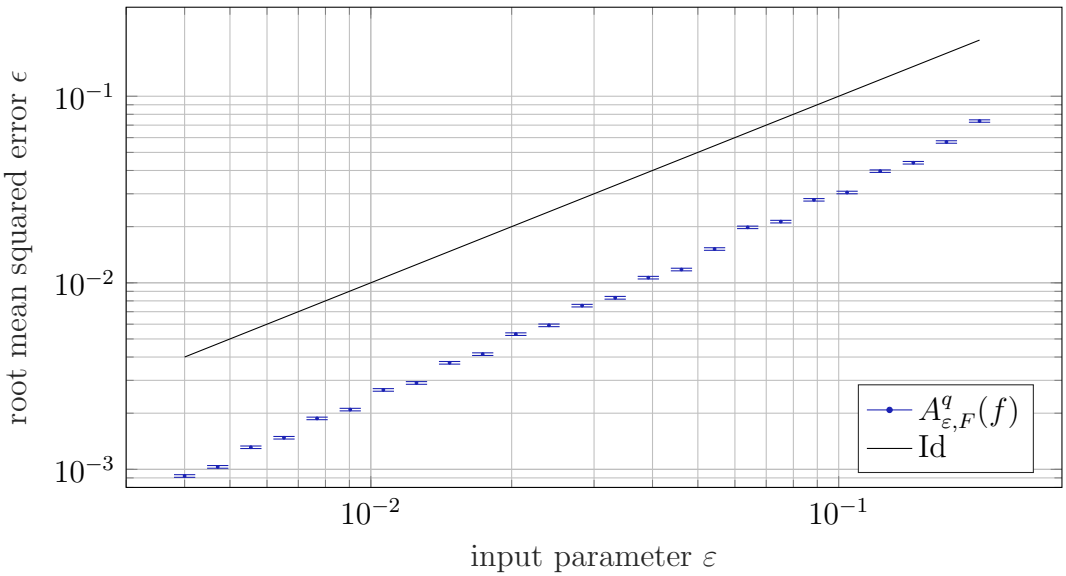

FIGURE 2. Error of the non-adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

This figure has two purposes. One is to relate the cost of $A^q_{\varepsilon,F}(f)$ in terms of its random generator calls, to the input parameter ε and the second, the main purpose is to give an intuition on how to choose the input parameter ε such that $A^q_{\varepsilon,F}(f)$ falls below a given error bound ϵ.

Finally, we turn to Figure 3, which is dedicated to the effect of the bit approximations on the empirical values of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ on the first few levels, starting with $\ell = 1$, since there is no coupling on level $\ell = 0$. More precisely, we compare empirical estimates of $|\operatorname{Bias}^q_\ell(f)|$ and $\operatorname{Var}^q_\ell(f)$ to those of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ for the particular bit numbers

$q = 3$ and $q = 6$. The level is layed on the x-axis and the empirical estimates on the y-axis, respectively.

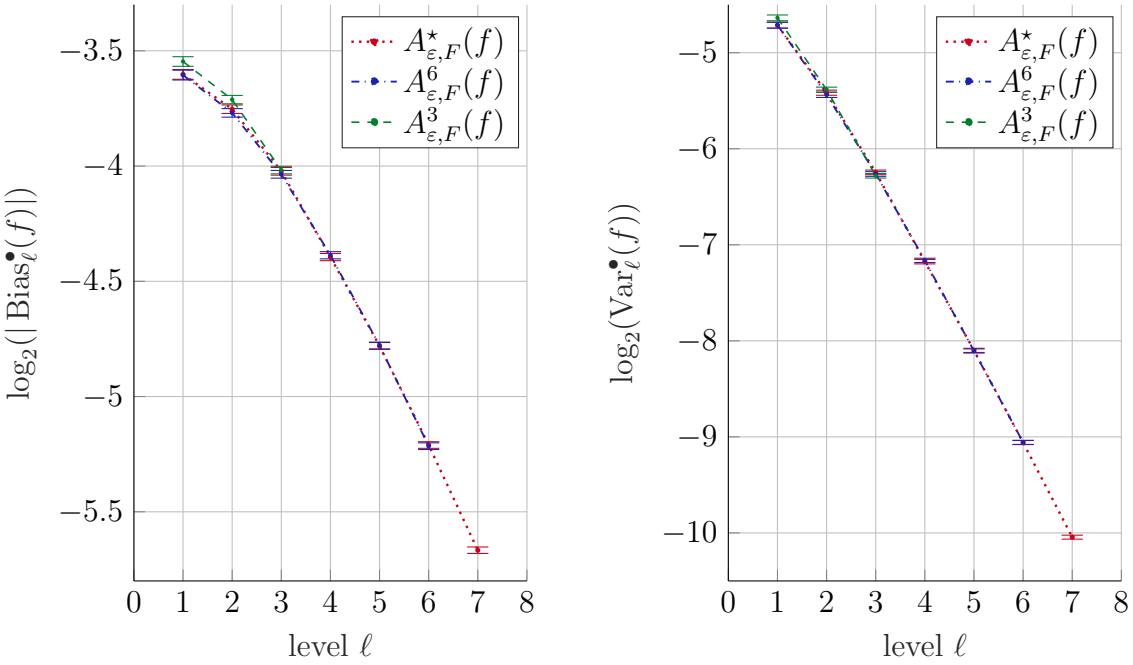

FIGURE 3. Empirical evolution of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ for different bit numbers compared to random numbers. Here, $\bullet \in \{q, \star\}$ indicates the affiliation to the particular algorithm.

There are two main observations. We start with the discussion of $|\operatorname{Bias}^q_\ell(f)|$. Using only $q = 3$ bits we see that the estimated values of $|\operatorname{Bias}^q_\ell(f)|$ are already relatively close to those of $|\operatorname{Bias}_\ell(f)|$, and for $q = 6$ bits, the two quantities do almost coincide. That is, the effect of the bit approximations has a negligible impact on the mean of the differences $f(X_{2^\ell}) - f(\widetilde{X}_{2^{\ell-1}})$, especially since $A^q_{\varepsilon,F}(f)$ already chooses the highest level $L = 5$ for the input $\varepsilon = 0.1$, which corresponds to an error ϵ of the magnitude 10^{-2}, see Figure 2. The same holds true for the difference of the empirical estimates of $\operatorname{Var}^q_\ell(f)$ and $\operatorname{Var}_\ell(f)$. The second observation is that especially the law of the decay of $(|\operatorname{Bias}^q_\ell(f)|)_{\ell\in\mathbb{N}}$ clearly involves a logarithmic factor, indicating that this factor should indeed be respected when estimating the laws for the decays of $(|\operatorname{Bias}^\dagger_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\operatorname{Var}^\dagger_\ell(f))_{\ell\in\mathbb{N}}$ in the adaptive multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$, which is discussed in Section IV.3.3.

V.1.2. MLMC Quadrature Based on Scheme 2. In this section, we present numerical results for the adaptive random bit multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. In the first figure, Figure 4, we present, analogously to Figure 1, the relation between the error and the cost of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ and we compare it to the corresponding relation between the error and the cost of the classical adaptive multilevel algorithm $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$, in terms of the number of random generator calls. In difference to the non-adaptive algorithms $A^q_{\varepsilon,F}(f)$ and $A^\star_{\varepsilon,F}(f)$, here the number of calls to the random generator is not deterministic in the input parameter ε, instead it is a random quantity. Hence we present the empirical mean of the number of random generator calls together with asymptotic confidence intervals to the confidence level 0.05. Again the error ϵ is measured on the x-axis, decreasing from right to left, and the number of generator calls is measured on the y-axis. As in Figure 1, in the current setting, the random bit algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ is punished by a factor of about 4 in the comparison with the

classical random number algorithm $A^{\star,\mathrm{adp}}_{\varepsilon,F}$, see (98). Moreover, with the same reasoning as in Section V.1.1 we illustrate the supposable asymptotic behaviour of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}$ by the function $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^{\gamma}$ with a parameter choice for c and γ that seems to fit the empirical data.

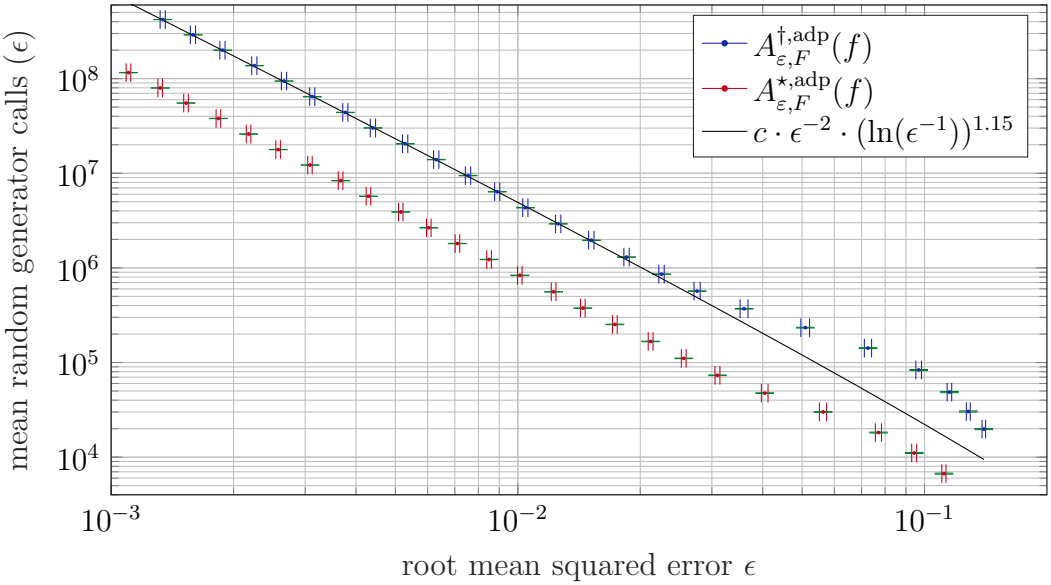

FIGURE 4. Error-cost relation of the adaptive random bit MLMC Euler algorithm (blue) and its random number counterpart (red) with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

We discuss Figure 4. The results are similar to those for $A^{q}_{\varepsilon,F}(f)$ in Figure 1. In particular we do have sufficiently small confidence intervals to stick to the empirical approximations of the error ϵ and to the number of random generator calls of the algorithms under consideration. The latter, the confidence intervals for the empirical estimation of the number of random generator calls, printed in green, actually almost vanish. The asymptotic behaviour of the random bit algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ seems to obey the law $c \cdot \epsilon^{-2}(\ln(\epsilon^{-1}))^{1.15}$, which seems to coincide with the law of the classical random number algorithm $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$, up to a multiplicative constant that seems to be close to 5. As we have already noticed that estimating the cost of the algorithms via counting their random generator calls leads to a factor of about 4 by which $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ is inferior to $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$, we conclude that both algorithms seem to have a very similar performance. That is, random bits seem to have an almost negligible effect on the adaptive multilevel Euler algorithm in the case of the standard Brownian motion and its maximum. The logarithmic exponent of 1.15 seems reasonable since Figure 6 shows that the exponential decay of $(\mathrm{Var}^{\dagger}_{\ell}(f))_{\ell\in\mathbb{N}}$ and $(\mathrm{Var}^{\star}_{\ell}(f))_{\ell\in\mathbb{N}}$ is close to 1, cf. Theorem 10. Finally, we consider the behaviour for errors ϵ in the range from $4\cdot 10^{-2}$ to $2 \cdot 10^{-1}$. It is very likely, that the replication numbers $N^{\dagger}$ and $N^{\star}$ used by the adaptive algorithms $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ and $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$, respectively, do simply not yield sufficiently good empirical approximations of $|\,\mathrm{Bias}^{\dagger}_{\ell}(f)|$ and $\mathrm{Var}^{\dagger}_{\ell}(f)$ respective $|\,\mathrm{Bias}^{\star}_{\ell}(f)|$ and $\mathrm{Var}^{\star}_{\ell}(f)$ leading to a precipitate termination of the respective algorithms, i.e., before the weak convergence is established, cf. Step 5 in Definition 9 together with Remark 35. This speculation is fortified by the fact the $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ seems to behave even more odd than $A^{\star,\mathrm{adp}}_{\varepsilon,F}$, coinciding with the unusual behaviour, i.e., the non-monotonicity, of $|\,\mathrm{Bias}^{\dagger}_{\ell}(f)|$ on the levels $\ell = 1, \ldots, 3$, cf. Figure 6.

Recall that one of the main motivations for the construction of adaptive multilevel algorithms is that for a given input accuracy ε, the error ϵ of the algorithm should be slightly smaller than and in no way substantially bigger than ε. Moreover, it should not be substantially smaller than ε, too, to avoid immense computational overhead. Hence in Figure 5 we illustrate the relation between the input accuracy ε, laid on the x-axis, and the empirical error ϵ of $A^{\dagger\,\mathrm{adp}}_{\varepsilon,F}(f)$ measured on the y-axis. For convenience of the reader the figure also contains the identity function $\mathrm{Id}\colon \mathbb{R} \to \mathbb{R}, x \mapsto x$.

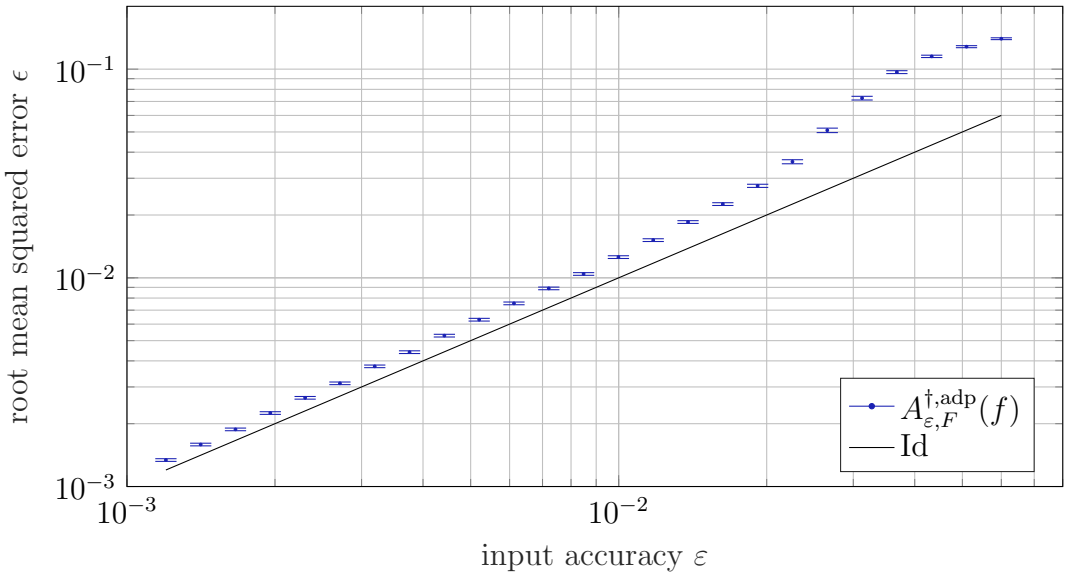

FIGURE 5. Error of the adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

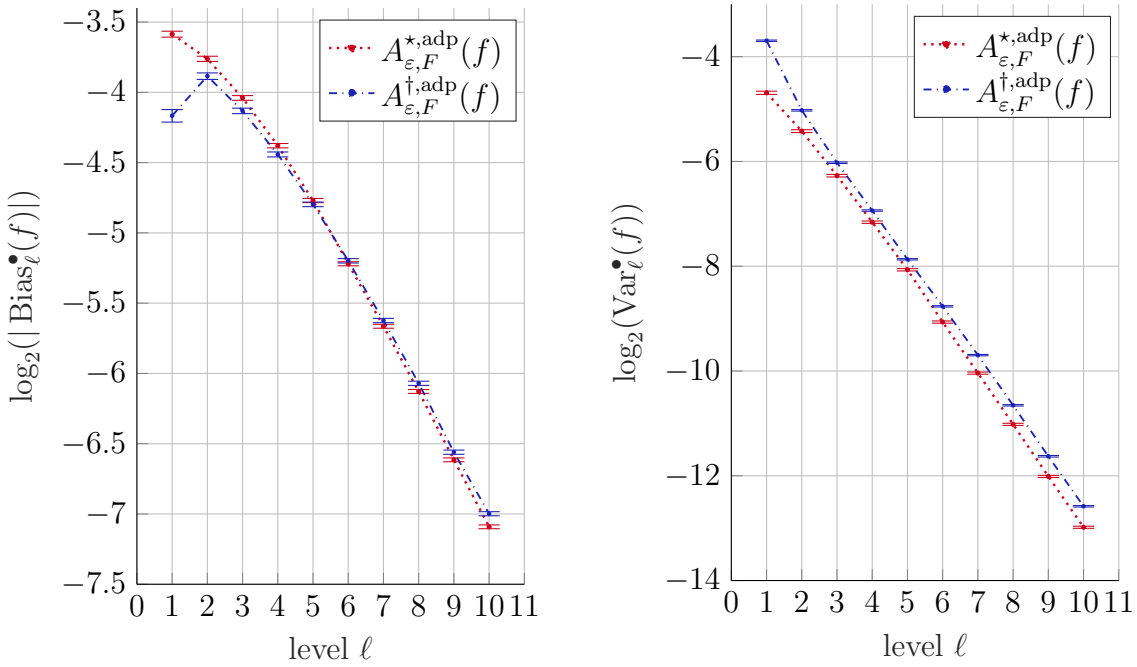

FIGURE 6. Empirical evolution of $|\,\mathrm{Bias}_{\ell}(f)|$ and $\mathrm{Var}_{\ell}(f)$ for random bits (blue) compared to random numbers (red).

Figure 5 shows that as soon as the input accuracy is sufficiently small such that $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ seems to perform properly, cf. the discussion of Figure 4, the error ϵ actually almost coincides with the input accuracy ε. In that sense, $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ fulfills its purpose.

Like for the non-adaptive algorithm $A^{q}_{\varepsilon,F}(f)$, we take a look at the effect of the bit approximations on the empirical estimates of $|\operatorname{Bias}^{\star}_{\ell}(f)|$ and $\operatorname{Var}^{\star}_{\ell}(f)$ on the first few levels, again starting with $\ell = 1$. That is, we compare $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$ to $|\operatorname{Bias}^{\star}_{\ell}(f)|$ and $\operatorname{Var}^{\dagger}_{\ell}(f)$ to $\operatorname{Var}^{\star}_{\ell}(f)$, respectively. As in Figure 3, the level is laid on the x-axis and the empirical estimate on the y-axis, respectively.

There are two main observations. The first one is that $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$ and $|\operatorname{Bias}^{\star}_{\ell}(f)|$ as well as $\operatorname{Var}^{\dagger}_{\ell}(f)$ and $\operatorname{Var}^{\star}_{\ell}(f)$ share almost the same asymptotic order of convergence and differ only by an almost negligible small constant. The second observation is that $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$ behaves somehow unexpected on the first level $\ell = 1$, namely the monotonicity of $|\operatorname{Bias}^{\star}_{\ell}(f)|$ is violated. Probably a consequence of the bit approximations using only two bits on the first level, because the confidence intervals suggest that is rather not a consequence of the empirical estimation of $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$.

Finally, we compare $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ to the adaptive random bit multilevel algorithm $A^{\dagger,\mathrm{adp}^{\star}}_{\varepsilon,F}(f)$,

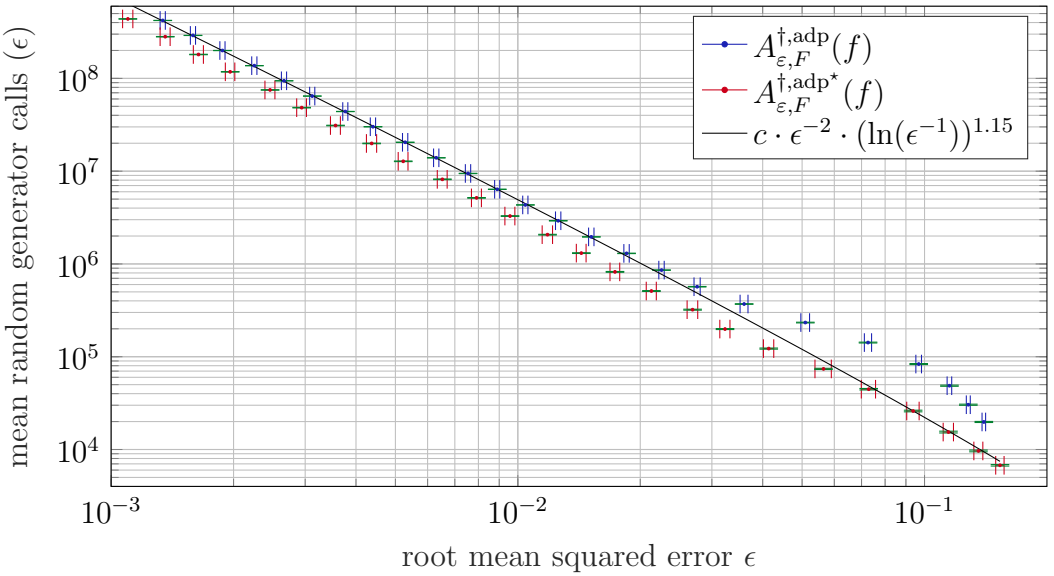

FIGURE 7. Error-cost relation of the adaptive random bit MLMC Euler algorithm without (blue) and with (red) a modified bias and variance estimation, both with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the algorithms.

that uses the coupled random bit Euler approximations $(X^{\dagger}_{2^{\ell}}, \widetilde{X}^{\dagger}_{2^{\ell-1}})_{\ell\in\mathbb{N}_0}$ and the construction from Definition 9 together with the modifications described in Remark 35 and Remark 36, as well as the split $\varepsilon_B^2 = \varepsilon^2/4$ and $\varepsilon_V^2 = 3/4\cdot\varepsilon^2$, i.e., the random bit counterpart of the multilevel Euler algorithm presented in Giles [25] for $f \in F_p$. As discussed in the introduction to the current chapter, cf. also Remark 37, this algorithm involves way smaller condition numbers associated to the design matrix D in the least-square fits for $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$ and $\operatorname{Var}^{\dagger}_{\ell}(f)$. This comparison has two purposes. One is to check whether $A^{\dagger,\mathrm{adp}^{\star}}_{\varepsilon,F}(f)$ performs better than $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$, and the second purpose is to get a better intuition on the observations for input accuracies in the range $[4\cdot 10^{-2}, 2\cdot 10^{-1}[$. In the discussion of Figure 4 we already made the guess that the curve shape in this range is due to too bad fits of the parameters in the

estimation of $|\mathrm{Bias}^{\dagger}_{\ell}(f)|$ and $\mathrm{Var}^{\dagger}_{\ell}(f)$, respectively, leading to an overhasty termination of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. This interpretation is fortified by Figure 7, since $A^{\dagger,\mathrm{adp}^\star}_{\varepsilon,F}(f)$ seems to behave more steady then $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ in combination with asymptotic confidence intervals sufficiently small not to credit this effect to the empirical estimation of the error ϵ.

V.1.3. The Comparison. In this section, we compare $A^{q}_{\varepsilon,F}(f)$ to $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. As mentioned several times throughout the thesis, we expect the adaptive algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ to perform better than the non-adaptive algorithm $A^{q}_{\varepsilon,F}(f)$.

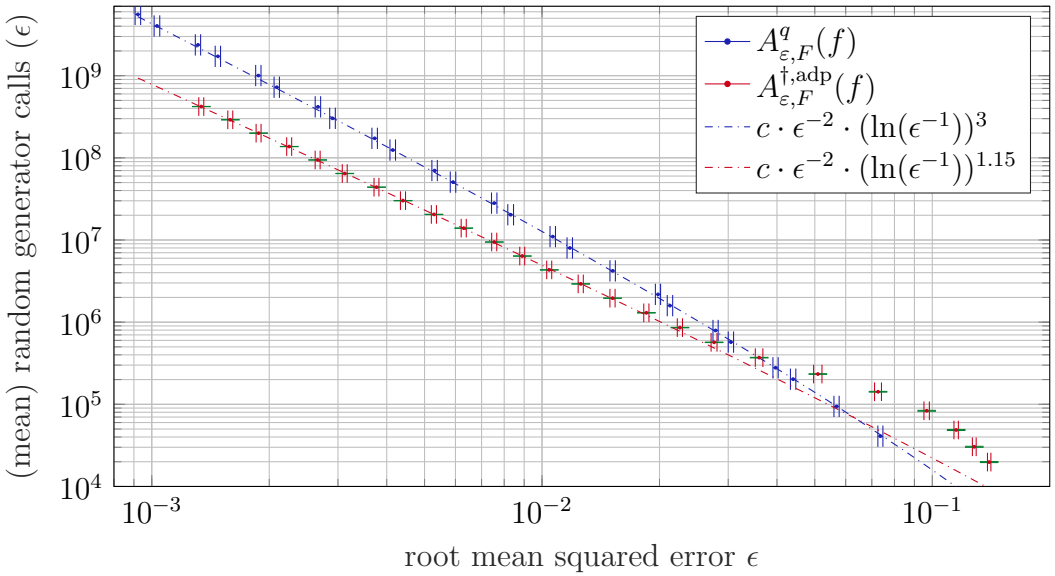

FIGURE 8. Error-cost relation of the non-adaptive (blue) and of the adaptive (red) random bit MLMC Euler algorithm with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The dashed line shows the supposed asymptotic behavior of the algorithm, respectively.

We present the empirical number of random generator calls of each algorithm measured on the y-axis plotted against their errors ϵ, measured on the x-axis. We add functions of the type $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^{\gamma}$ with parameter choices for c and γ that seem to describe the asymptotic laws of the algorithms in matching colors, respectively. Indeed it turns out that $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ shows a better asymptotic order of the number of random generator calls by almost two logarithmic orders. Furthermore, it suggests the superiority of the adaptive algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ for errors ϵ less than $4 \cdot 10^{-2}$.

V.2. Geometric Brownian Motion

In this section we consider a one-dimensional geometric Brownian motion, more precisely, the solution process $X = (X(t))_{t\in[0,1]}$ to the SDE with drift coefficient $a(x) = 0.02 \cdot x$ and diffusion coefficient $b(x) = 0.2 \cdot x$ as well as the initial value $x_0 = 1$, i.e.,

$$\begin{aligned} \mathrm{d}X(t) &= 0.02 \cdot X(t)\,\mathrm{d}t + 0.2 \cdot X(t)\,\mathrm{d}W(t), \quad t \in [0,1], \\ \mathrm{d}X(0) &= 1. \end{aligned}$$

This choice of the coefficient b corresponds to the choice in Giles [24, Section 6.1] and the coefficient a is chosen such that $\mathrm{E}(f(X))$ can easily be computed analytically. As for the

Brownian motion, we consider the maximum $f(x) = \max_{0\le t\le 1} x(t)$ for $x \in C([0,1])$. Consequently, the current setting is, again, covered by our theoretical results from Section IV.5.

We proceed in the same way as for the Brownian motion in Section V.1. First of all, we compute the reference solution $\mathrm{E}(f(X))$ analytically. For this purpose, observe that the solution X at time $t \in [0,1]$ of the above SDE is given by

$$X(t) = \exp(0.2 \cdot W(t)),$$

see, e.g., Taylor and Karlin [63, Section VIII.4.2, i.p. equation (4.17)] or Wiersema [65, Section 5.3]. Hence from (106) we get

$$f(X) = \max_{0\le t\le 1} \exp(0.2 \cdot W(t)) = \exp\bigl(0.2 \cdot \max_{0\le t\le 1} W(t)\bigr) \overset{\mathrm{d}}{=} \exp(0.2 \cdot |W(1)|),$$

implying

$$\begin{aligned}\mathrm{E}(f(X)) &= 2 \cdot \int_0^\infty \exp(0.2 \cdot y) \cdot (2\cdot\pi)^{-1/2} \cdot \exp(-y^2/2)\,\mathrm{d}y \\ &= (2/\pi)^{1/2} \cdot \int_0^\infty \exp(0.2 \cdot y - y^2/2)\,\mathrm{d}y = 1.1819\ldots.\end{aligned}$$

Hence, the empirical error estimations of the multilevel algorithms are again straightforward.

The numerical results are presented in three separate sections, following the outline in Section V.1. Since the results are very similar to those for the Brownian motion, presented in Section V.1, we will not explain and discuss all figures in detail. Instead we emphasize the most important differences.

V.2.1. MLMC Quadrature Based on Scheme 1. In this section we present numerical results for the non-adaptive random bit multilevel algorithm $A^q_{\varepsilon,F}(f)$. The main difference to the results for the standard Brownian motion is located in the relation between the number of random generator calls carried out by $A^q_{\varepsilon,F}(f)$ and the empirical error ϵ of $A^q_{\varepsilon,F}(f)$.

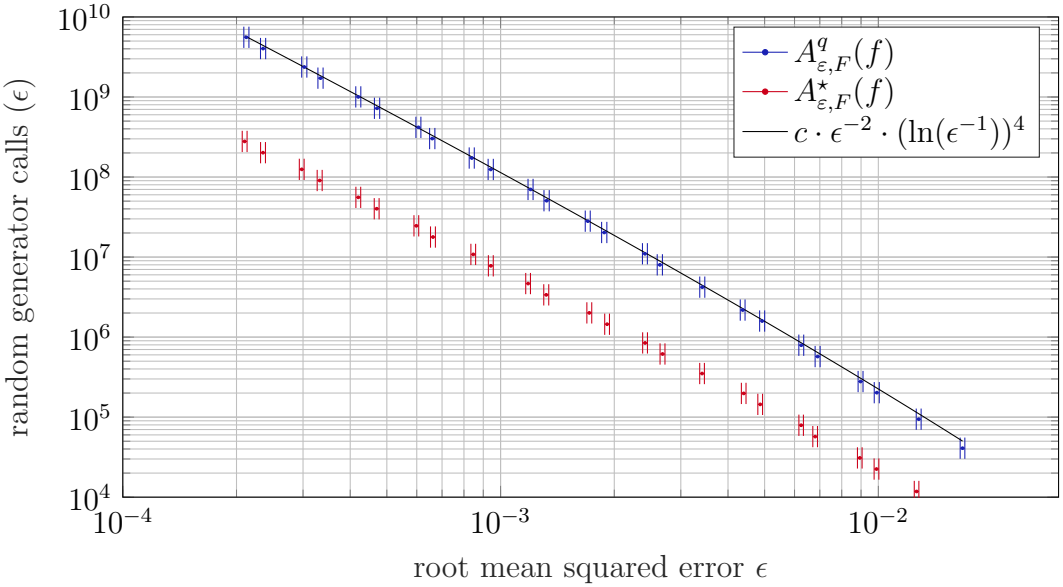

FIGURE 9. Error-cost relation of the non-adaptive MLMC Euler algorithms (random bits: blue, random numbers: red) with asymptotic confidence intervals to the confidence level 0.95. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

Actually, the number of random generator calls seems to obey the law $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^4$ for some positive constant c. Recall that, in comparison, for the maximum of the standard Brownian motion Figure 1 suggests the asymptotic law $c_1 \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^3$ for some positive constant c_1. However, this one logarithmic order we lose for the geometric Brownian motion $(X(t))_{t\in[0,1]}$ is actually good news in the sense that it implies that the general upper bound on the cost of $A^q_{\varepsilon,F}(f)$ in Theorem 11 is sharp. At least for the construction of $A^q_{\varepsilon,F}(f)$ not having the Bakhvalov trick from Section IV.5.1 at its disposal.

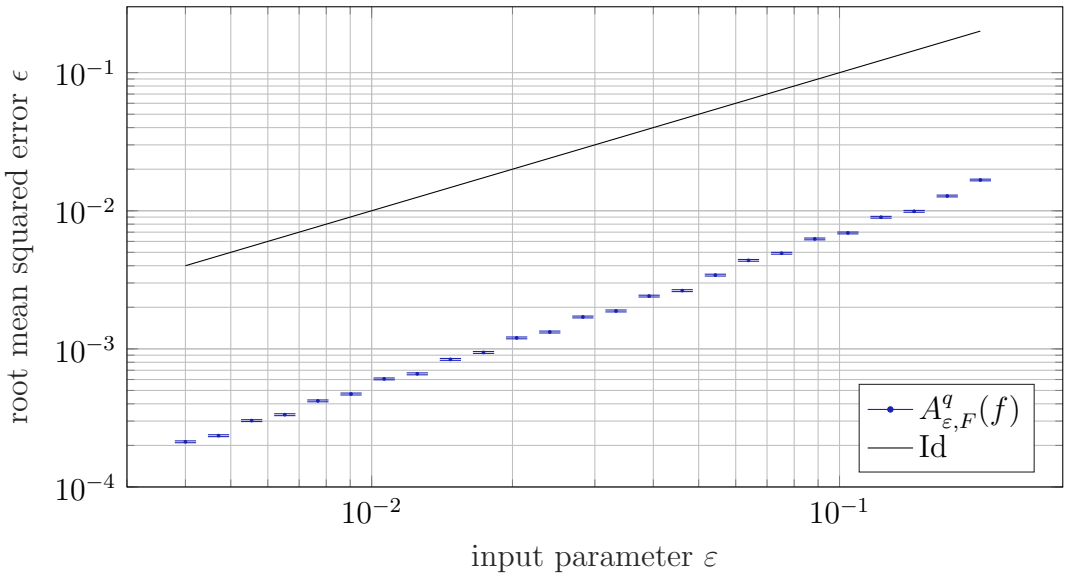

FIGURE 10. Error of the non-adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

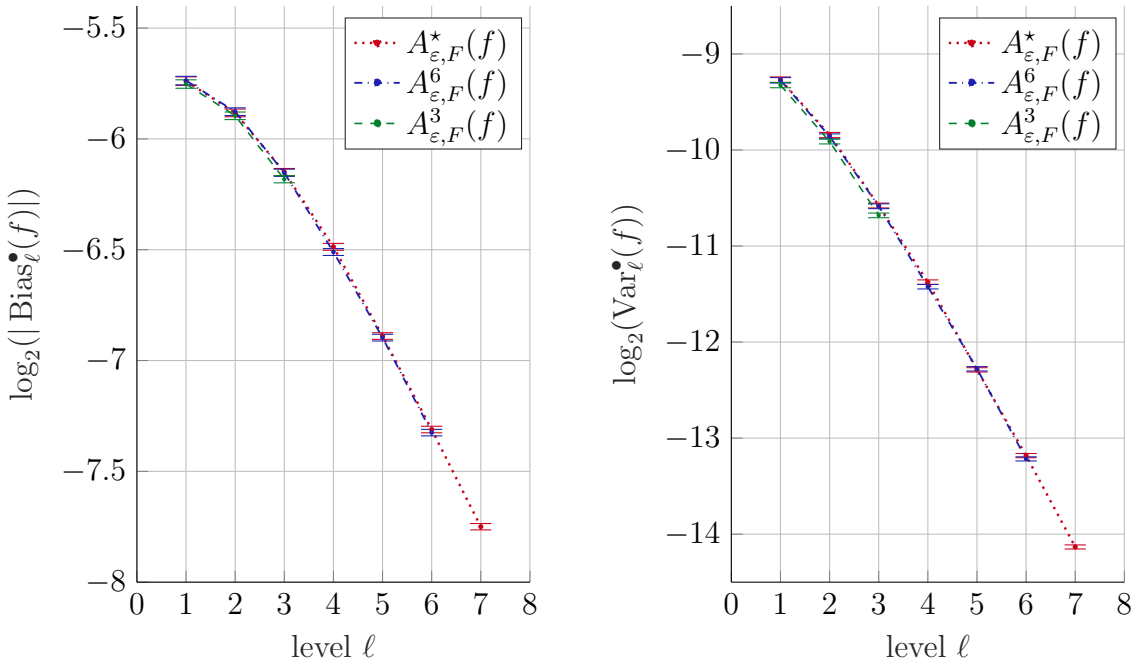

FIGURE 11. Empirical evolution of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ for different bit numbers compared to random numbers. Here, $\bullet \in \{q, \star\}$ indicates the affiliation to the particular algorithm.

Figure 11 once again clearly indicates the presence of a logarithmic factor in the laws of $(|\operatorname{Bias}_\ell^\bullet(f)|)_{\ell\in\mathbb{N}}$ for $\bullet \in \{q, *\}$, respectively. For the corresponding variances $(\operatorname{Var}_\ell^\bullet(f))_{\ell\in\mathbb{N}}$ this might also be the case but not as distinctive as implied by the empirical estimates of $|\operatorname{Bias}_\ell^\bullet(f)|$ in Figure 11.

V.2.2. MLMC Quadrature Based on Scheme 2. In this section, we present numerical results for the adaptive random bit multilevel algorithm $A_{\varepsilon,F}^{\dagger,\mathrm{adp}}(f)$.

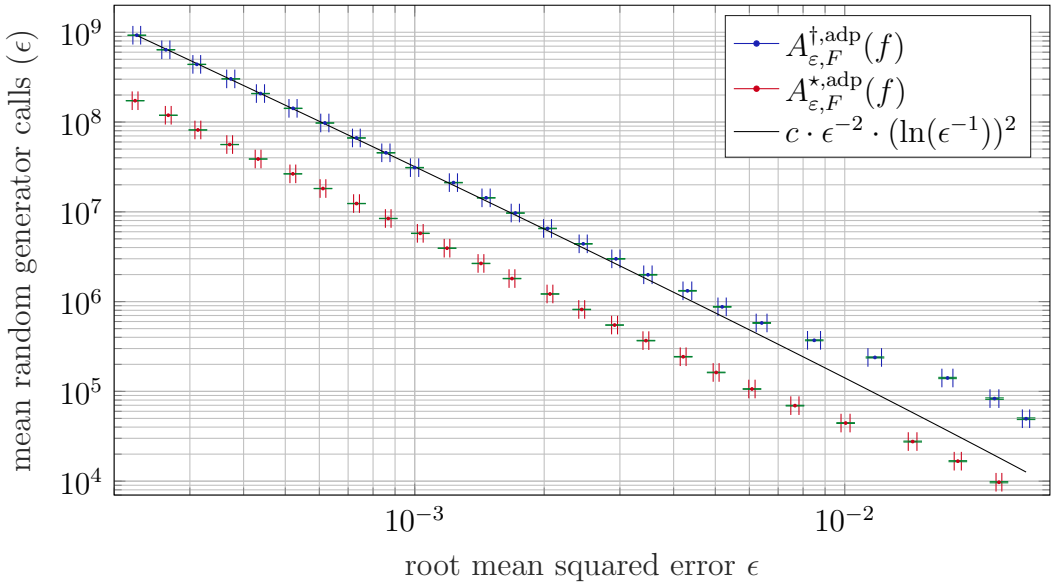

FIGURE 12. Error-cost relation of the adaptive random bit MLMC Euler algorithm (blue) and its random number counterpart (red) with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

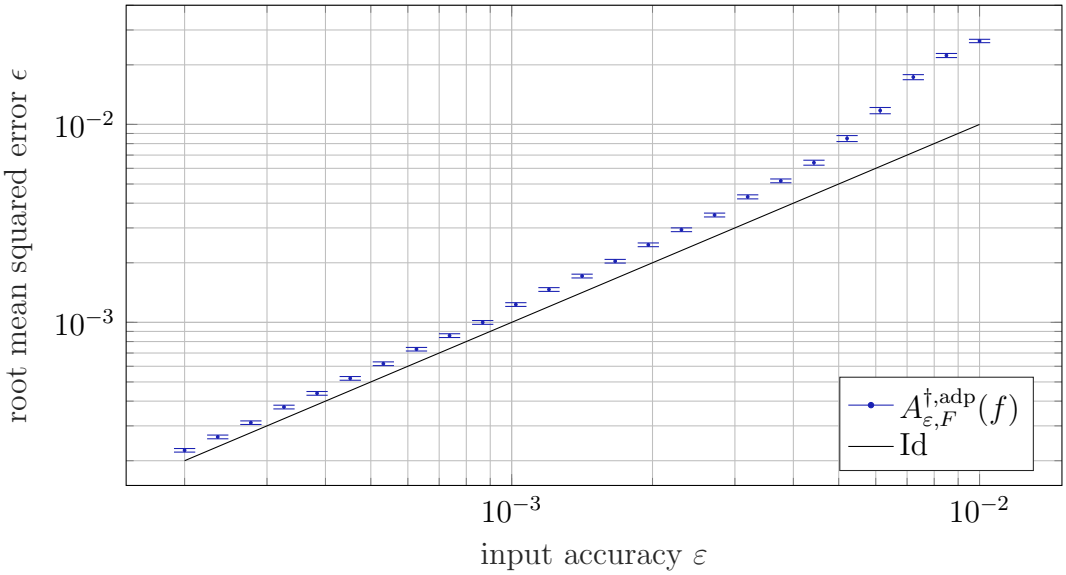

FIGURE 13. Error of the adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

The main difference to the results for the standard Brownian motion lies, as for the non-adaptive algorithm $A_{\varepsilon,F}^{q}(f)$, in the relation between the number of random generator calls

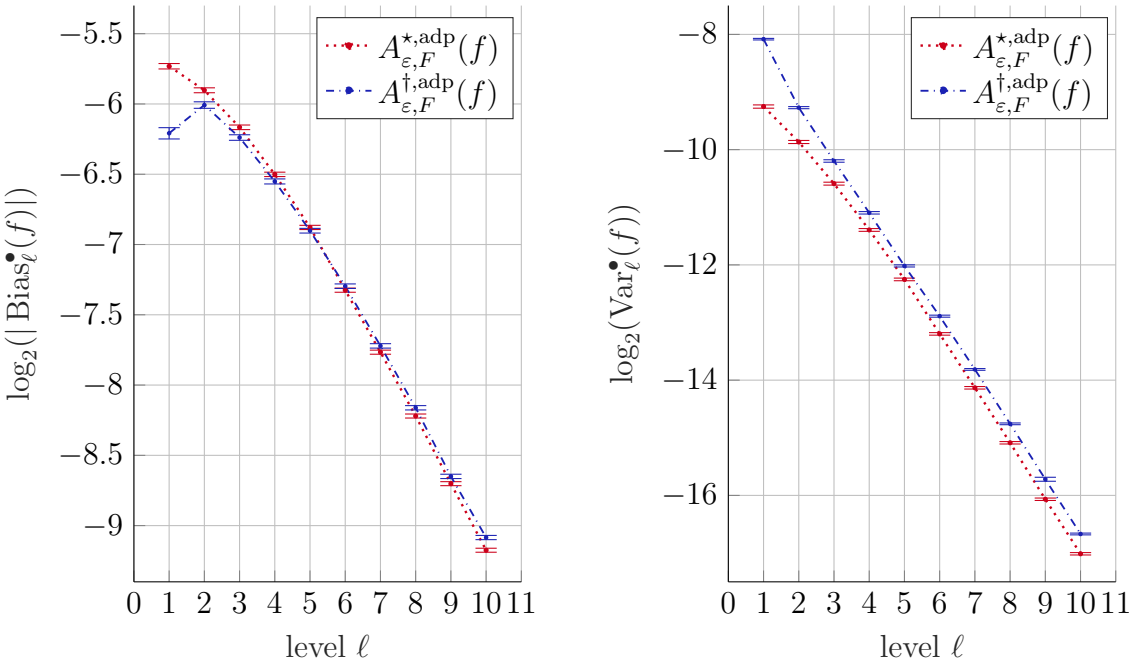

FIGURE 14. Empirical evolution of $|\,\mathrm{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ for random bits (blue) compared to random numbers (red).

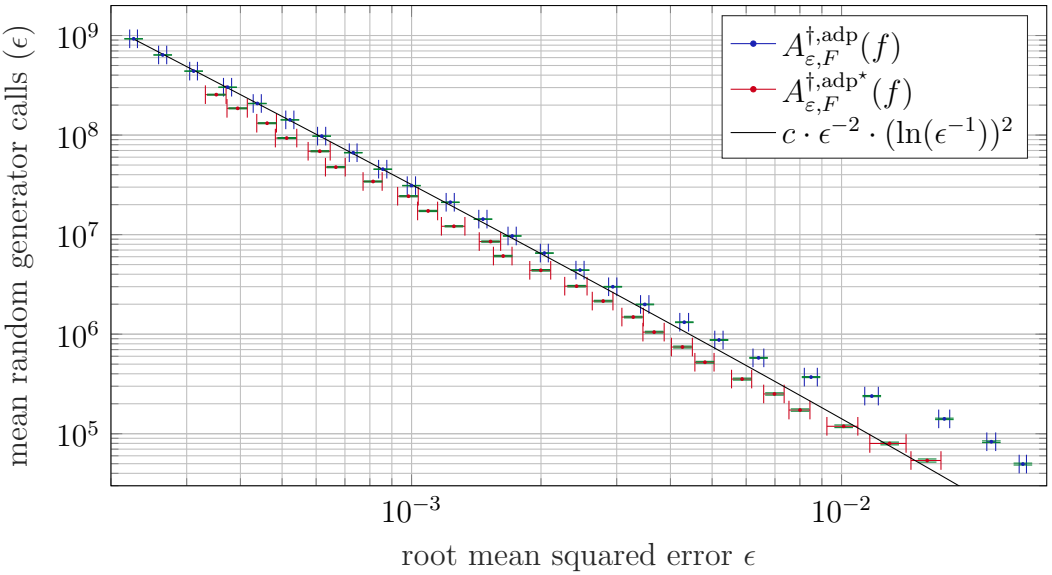

FIGURE 15. Error-cost relation of the adaptive random bit MLMC Euler algorithm without (blue) and with (red) a modified bias and variance estimation, both with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the algorithms.

and the empirical error ϵ of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. Actually, this relation seems to have the asymptotic law $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^2$ for a positive constant c. Recall that in the case of the standard Brownian motion we have observed the logarithmic factor 1.15, cf. Figure 4. That is we lose almost one logarithmic order, which corresponds to the effect that we have observed for the non-adaptive algorithm $A^{q}_{\varepsilon,F}(f)$ in Section V.2.1. However, we stress that the logarithmic exponent of 2 seems to also hold for the classical adaptive multilevel Euler algorithm $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$

and corresponds to the (weak) asymptotic upper bound on its cost in dependence on the input ε, that is presented in Theorem 10.

Finally, we observe that Figures 13, 14 and 15 fortify the interpretation of the observations for the standard Brownian motion in Section V.1.2, hence referred to for their discussion.

V.2.3. The Comparison. In this section, we compare $A^{q}_{\varepsilon,F}(f)$ to $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$.

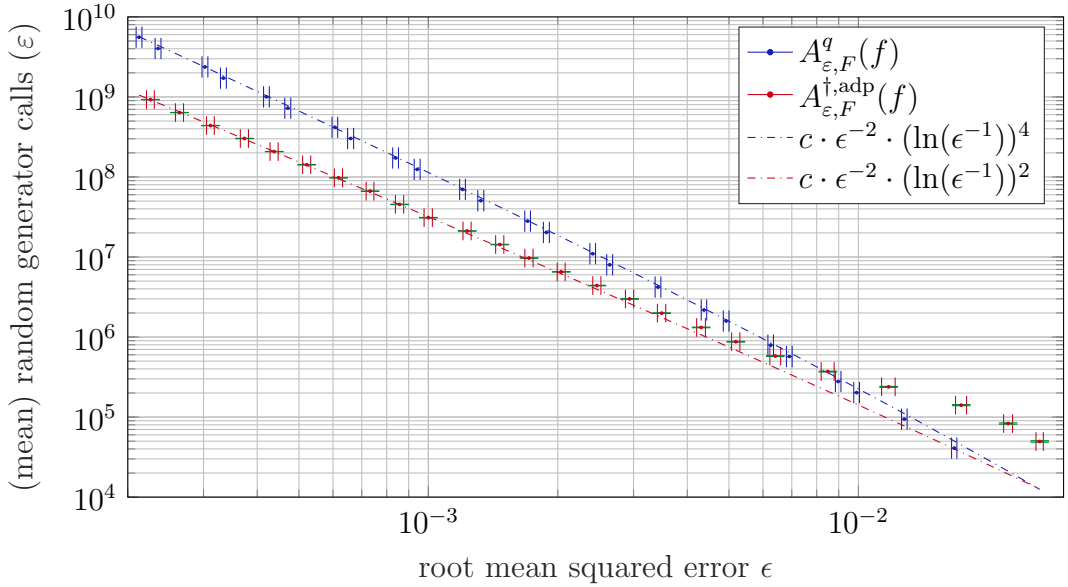

FIGURE 16. Error-cost relation of the non-adaptive (blue) and of the adaptive (red) random bit MLMC Euler algorithm with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The dashed line shows the supposed asymptotic behavior of the algorithm, respectively.

Similar as for the Brownian motion, see Section V.1.3, we observe that the adaptive algorithm outruns the non-adaptive algorithm for small errors ϵ by two logarithmic orders. Actually, Figure 16 suggests the superiority of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ for errors ϵ less than 10^{-2}, coinciding with the error range for which $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ seems to become more stable regarding the involved linear least-square fits for $|\operatorname{Bias}^{\dagger}_{\ell}(f)|$ and $\operatorname{Var}^{\dagger}_{\ell}(f)$, cf. the discussions of Figure 4 and Figure 7.

V.3. Ornstein-Uhlenbeck Process

We consider an Ornstein-Uhlenbeck process $X = (X(t))_{t\in[0,1]}$, i.e., the solution of a stochastic differential equation of the form

$$\begin{aligned} \mathrm{d}X(t) &= (a_1 - a_2 \cdot X(t))\,\mathrm{d}t + b_1\,\mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= x_0 \end{aligned}$$

with positive constants a_1, a_2 and b_1. More precisely, we are interested in the ad hoc parameter choice $a_1 = 1$, $a_2 = 1$, and $b_1 = 1$ as well as the initial value $x_0 = 1$. In our notation, cf. (105), this corresponds to the drift coefficient $a(x) = 1 - x$ and the diffusion coefficient $b(x) = 1$, i.e.,

$$\begin{aligned} \mathrm{d}X(t) &= (1 - X(t))\,\mathrm{d}t + \mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= 1. \end{aligned} \tag{107}$$

Observe that X can be interpreted as a, as we will call it, tamed Brownian motion, since the Brownian motion is pushed back towards its initial value by the drift coefficient a.

In contrast to the first two examples, here we are interested in the functional f that evaluates the process X at the final time point $t = 1$, i.e., $f(x) = x(1)$ for $x \in C([0,1])$. Hence the current setting is, once again, covered by our theoretical results from Section IV.5.

We proceed in the same way as in Section V.1 and Section V.2. At first we compute $\mathrm{E}(f(X))$ analytically. From Shreve [61, Example 4.4.10] we obtain that $X(t)$ is normally distributed with expectation

$$\mathrm{E}(X(t)) = \exp(-a_2 \cdot t) \cdot X(0) + a_1/a_2 \cdot (1 - \exp(-a_2 \cdot t))$$

for all $t \in [0,1]$. Hence for our particular parameter choice, we have

$$\mathrm{E}(f(X)) = \exp(-1) + (1 - \exp(-1)) = 1.$$

For us, the most interesting aspect of the Ornstein-Uhlenbeck process is that X is the solution process of an SDE with additive noise. Hence the Euler scheme coincides with the Milstein scheme, cf. Remark 6. Since the functional f evaluates the solution of the SDE in the final time point only, Theorem 3 yields that the Euler-Maruyama scheme has a weak rate of convergence of at least 1 instead of 1/2. Consequently, the adaptive multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ should be able to exploit this higher order of convergence in contrast to $A^{q}_{\varepsilon,F}(f)$, which is based on the upper bound on the order of convergence of the Euler-Maruyama scheme from Theorem 2.

As in the previous examples we present the numerical results in three separate sections. Again, we focus on differences to the results for the previous examples, mainly the standard Brownian motion in Section V.1, and we do not discuss similarities in detail.

V.3.1. MLMC Quadrature Based on Scheme 1. We present numerical results for the non-adaptive multilevel algorithm $A^{q}_{\varepsilon,F}(f)$.

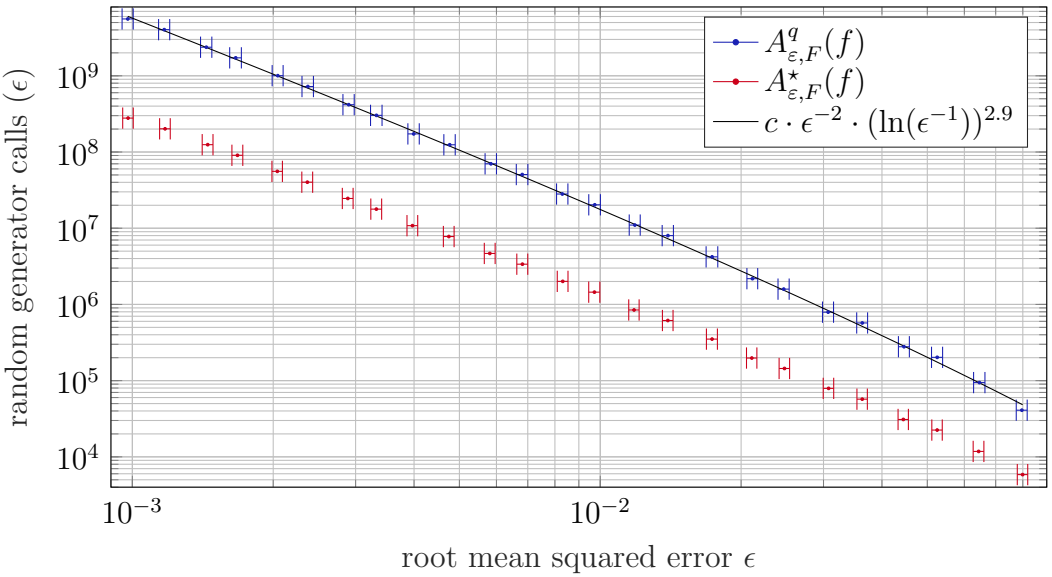

FIGURE 17. Error-cost relation of the non-adaptive MLMC Euler algorithms (random bits: blue, random numbers: red) with asymptotic confidence intervals to the confidence level 0.95. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

The first thing we notice is that the asymptotic relation between the error ϵ of $A^q_{\varepsilon,F}(f)$ and the number of random generator calls does almost coincide with the corresponding relation observed for the standard Brownian motion, cf. Figure 1. In the estimated logarithmic order, namely 2.9 vs 3, as well as on the level of constants. Actually, this is not really surprising, due to our interpretation of X as a tamed Brownian motion, see (107) and the comment thereafter, together with the fact that the maximum of a Brownian motion has the same distribution as the absolute value in the final time point, see (106), at which the functional f evaluates the process X in the current example.

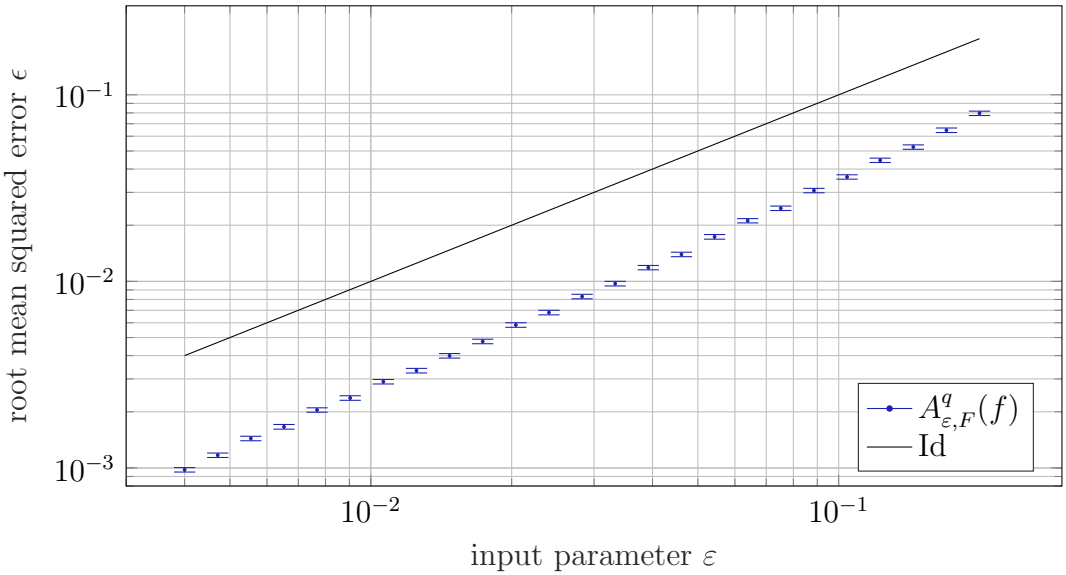

FIGURE 18. Error of the non-adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

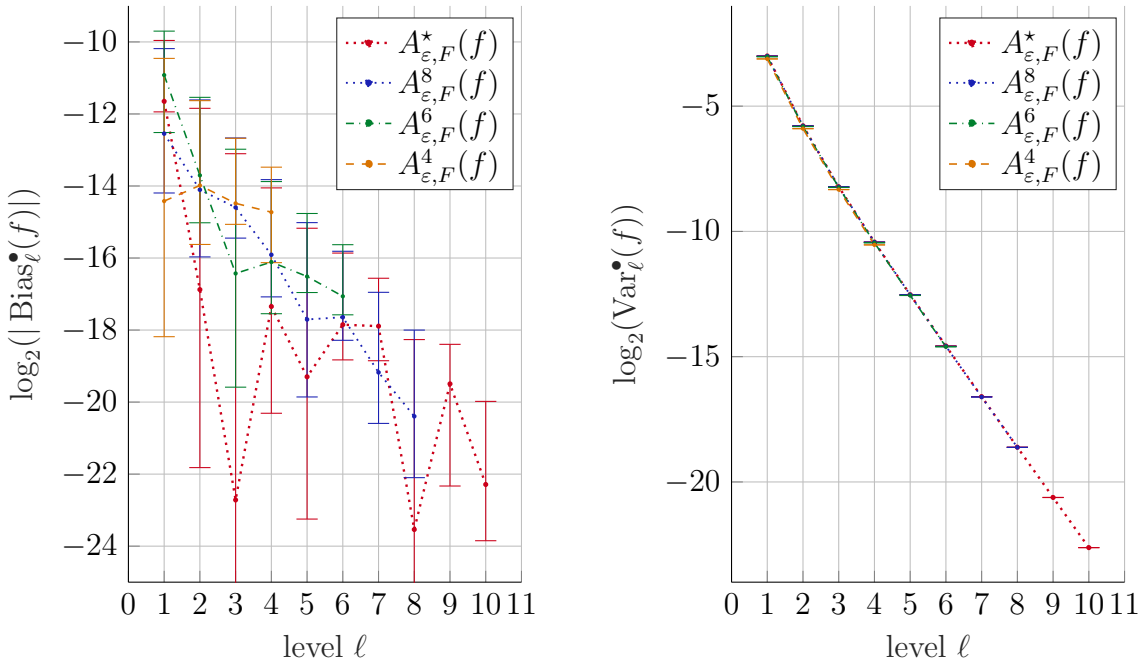

FIGURE 19. Empirical evolution of $|\,\mathrm{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ for different bit numbers compared to random numbers. Here, $\bullet \in \{q, \star\}$ indicates the affiliation to the particular algorithm.

Since the input parameter ε completely determines $A^q_{\varepsilon,F}(f)$, as a consequence of the afore discussion, also the relation between the input parameter ε and the error ϵ of $A^q_{\varepsilon,F}(f)$, presented in Figure 18, is almost the same as for the standard Brownian motion, cf. Figure 2.

The first substantial difference to the numerical results for the standard Brownian motion is in the exponential decay of the empirical estimates of $(|\operatorname{Bias}^\bullet_\ell(f)|)_{\ell\in\mathbb{N}}$ and $(\operatorname{Var}^\bullet_\ell(f))_{\ell\in\mathbb{N}}$ with $\bullet \in \{q,\star\}$. Though $|\operatorname{Bias}^\bullet_\ell(f)|$ has very large confidence intervals and hence one can not say that the results are very reliable, there is a clear tendency that $(|\operatorname{Bias}^\bullet_\ell(f)|)_{\ell\in\mathbb{N}}$ decreases with exponential order of about 1, which is close to the upper bound for the Milstein scheme given in Theorem 3. Of course it might be that the replication numbers chosen on each level $\ell = 1,\dots,10$ are to small, but we are already using 10^6 realizations of $f(X^\bullet_{2^\ell}) - f(\widetilde{X}^\bullet_{2^{\ell-1}})$, i.e., the simulations are already time demanding, especially for the larger choices of ℓ. The explanation is probably to be found in the symmetry of the Brownian motion, i.e., the same likelihood of being positive and negative. Therefore, in the empirical estimation of $|\operatorname{Bias}^\bullet_\ell(f)|$ the values for the realizations of $f(X^\bullet_{2^\ell}) - f(\widetilde{X}^\bullet_{2^{\ell-1}})$ are supposed to cancel out in the mean, which coincides with the relatively small values of the empirical estimates of $|\operatorname{Bias}^\bullet_\ell(f)|$ observed on the first levels, starting with a value very likely less than 2^{-10} for $\ell = 1$. In comparison the corresponding value for the standard Brownian motion is estimated to be of the magnitude $2^{-3.5}$, cf. Figure 3. However, we will not go into detail of this observation, since the results for $A^q_{\varepsilon,F}(f)$ seem to be very reliable in the sense that the confidence intervals for the empirical error ϵ are relatively small. Furthermore, the empirical values for the relation between the error ϵ and the number of random generator calls, see Figure 17, do almost coincide with $c\cdot\epsilon^{-2}\cdot(\ln(\epsilon^{-1}))^{2.9}$ for some positive constant c. Moreover, the confidence intervals for the empirical variances are almost negligible small and the values for the bit approximations $\operatorname{Var}^q_\ell(f)$ do almost coincide with those for $\operatorname{Var}^\star_\ell(f)$.

Finally, note that $(\operatorname{Var}^\star_\ell(f))_{\ell\in\mathbb{N}}$ is decreasing with an exponential order of about 2. Hence we can expect the asymptotic law $c\cdot\epsilon^{-2}$, with some positive constant c, for the adaptive random bit multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$, that we consider in the next section.

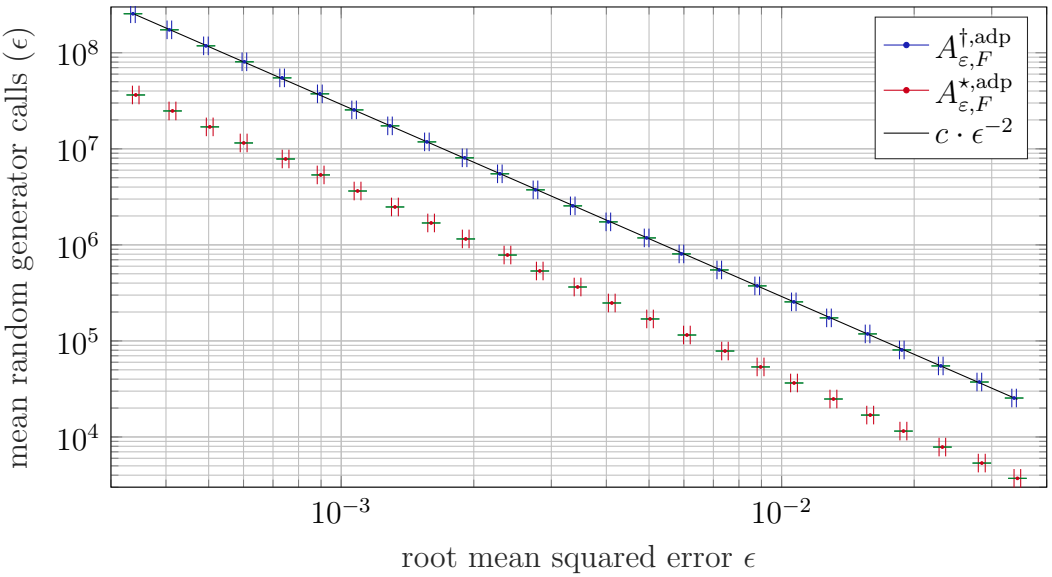

FIGURE 20. Error-cost relation of the adaptive random bit MLMC Euler algorithm (blue) and its random number counterpart (red) with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

V.3.2. MLMC Quadrature Based on Scheme 2. We present numerical results for the adaptive random bit multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$.

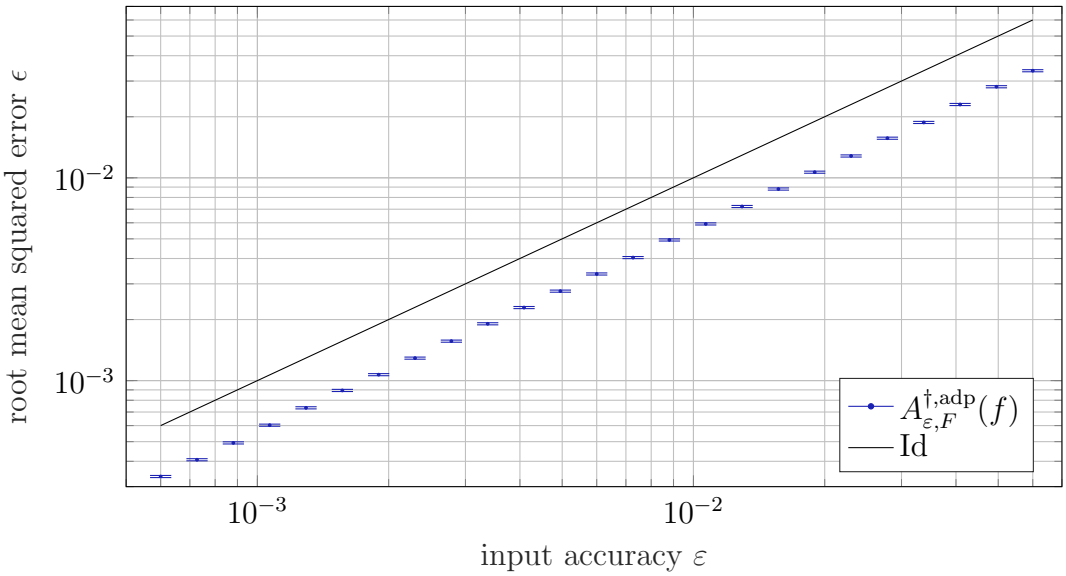

FIGURE 21. Error of the adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

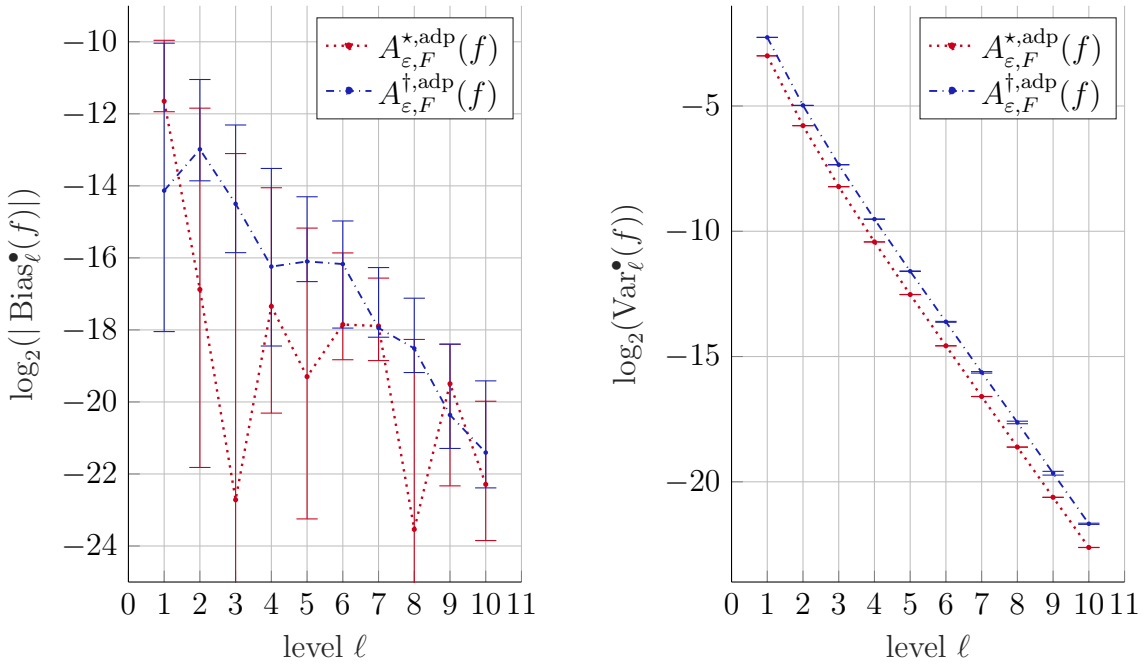

FIGURE 22. Empirical evolution of $|\,\mathrm{Bias}_{\ell}(f)|$ and $\mathrm{Var}_{\ell}(f)$ for random bits (blue) compared to random numbers (red).

Here, we observe a distinctive difference to the results for the standard Brownian motion in Section V.1.2. Namely the relation between the error ϵ and the number of random generator calls of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ seems to obey the law $c \cdot \epsilon^{-2}$ for some positive constant c. This corresponds to the exponent we were hoping for, regarding the level-variances $(\mathrm{Var}^{\dagger}_{\ell}(f))_{\ell\in\mathbb{N}}$ decaying with an order of about 2, see Figure 22, which actually corresponds to the case

$\beta > 1$ in Theorem 10. Observe that by Remark 26 the lower bound for any Monte Carlo algorithm is of the form $c_1 \cdot \epsilon^{-2}$ for some positive constant c_1.

The relation to the classical random number multilevel algorithm $A^{\star,\mathrm{adp}}_{\varepsilon,F}(f)$ is similar to the relations for the Brownian motion thoroughly discussed in Section V.1.2.

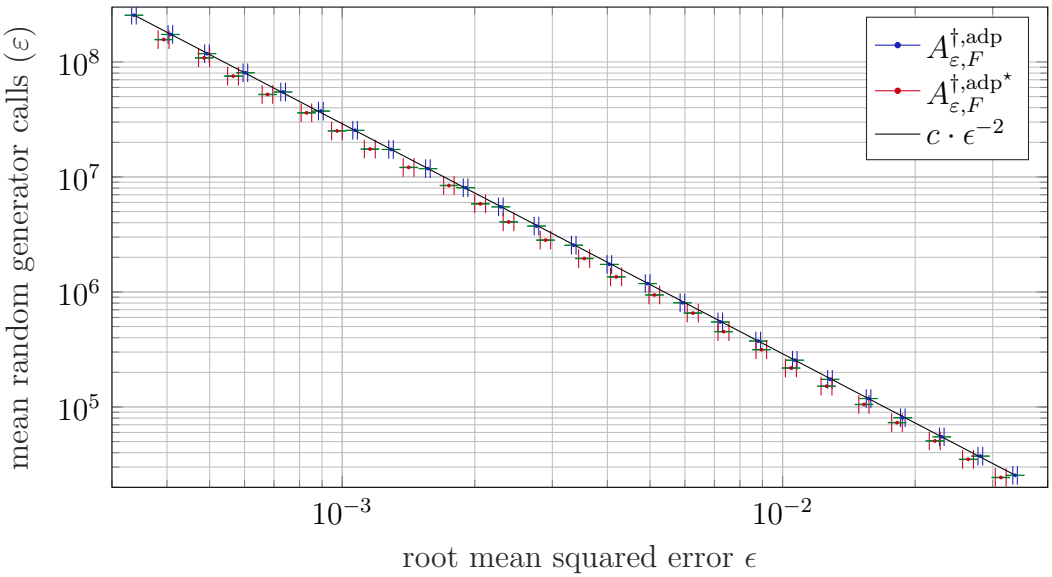

FIGURE 23. Error-cost relation of the adaptive random bit MLMC Euler algorithm without (blue) and with (red) a modified bias and variance estimation, both with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the algorithms.

Let us finally mention that in the current setting $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ achieves an error ϵ that is strictly less than the input accuracy ε, see Figure 21.

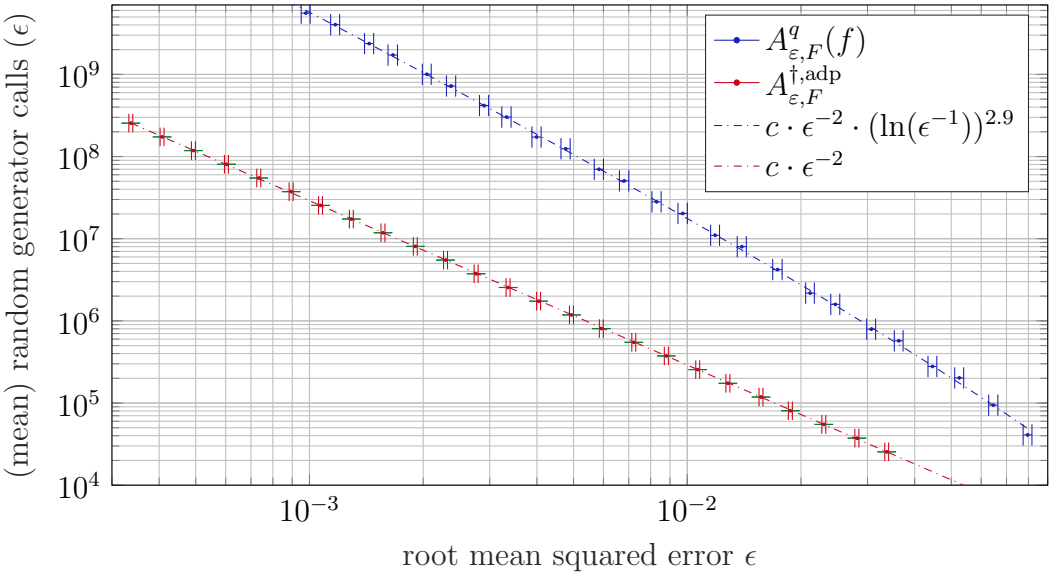

FIGURE 24. Error-cost relation of the non-adaptive (blue) and of the adaptive (red) random bit MLMC Euler algorithm with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The dashed line shows the supposed asymptotic behavior of the algorithm, respectively.

V.3.3. The Comparison. We compare the non-adaptive algorithm $A^{q}_{\varepsilon,F}(f)$ to the adaptive algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. In the previous examples we observed that the adaptive algorithm outruns the non-adaptive algorithm for small errors ϵ, and they both showed a similar performance for an error ϵ of the magnitude 10^{-2}, at least in terms of their numbers of random generator calls. In the current setting $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ has a way lower number of generator calls than $A^{q}_{\varepsilon,F}(f)$ on the whole interval under consideration, i.e., already for an error ϵ of $3 \cdot 10^{-2}$, $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ performs by a factor > 10 better than $A^{q}_{\varepsilon,F}(f)$. This clearly demonstrates the superiority of the adaptive algorithm over the non-adaptive algorithm.

V.4. Cox-Ingersoll-Ross (CIR) Process

We consider the Cox-Ingersoll-Ross process $X = (X(t))_{t\in[0,1]}$, i.e., the solution of a stochastic differential equation, which we call CIR SDE, of the form

$$\begin{aligned} \mathrm{d}X(t) &= (a_1 - a_2 \cdot X(t))\,\mathrm{d}t + b_1 \cdot \sqrt{X(t)}\,\mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= x_0 \end{aligned}$$

for positive constants a_1, a_2 and b_1. Existence and uniqueness of the solution are well-known results, see, e.g., Ekström and Tysk [21]. We consider the particular parameter choice $a_1 = 3$, $a_2 = 1$, and $b_1 = 2$, as well as the initial value $x_0 = 1$. In our notation, see (105), this corresponds to the drift coefficient $a(x) = 3 - x$ and the diffusion coefficient $b(x) = 2 \cdot \sqrt{x}$, i.e.,

$$\begin{aligned} \mathrm{d}X(t) &= (3 - X(t))\,\mathrm{d}t + 2 \cdot \sqrt{X(t)}\,\mathrm{d}W(t), \quad t \in [0,1], \\ X(0) &= 1. \end{aligned}$$

As for the Ornstein-Uhlenbeck process in Section V.3 we are interested in the functional $f(x) = x(1)$ for $x \in C([0,1])$. Observe that the diffusion coefficient is not globally Lipschitz continuous, as $\lim_{x\to 0} b'(x) = \infty$. Hence the current setting is not covered by our theoretical results in Section IV.5. However, for our parameter choice the CIR SDE should behave similar as if it would fulfill the global Lipschitz condition, since by Lamberton and Lapeyre [40, Proposition 6.2.4] the naturally to the time interval $[0,\infty[$ extended CIR process X satisfies

$$2 \cdot a_1 \geq b_1^2 \Rightarrow P\big(\inf\{t \geq 0 \colon X(t) = 0\} = \infty\big) = 1. \tag{108}$$

For the analytical computation of the reference solution $\mathrm{E}(f(X))$, Shreve [61, Example 4.4.11] gives

$$\mathrm{E}(X(t)) = \exp(-a_2 \cdot t) \cdot X(0) + a_1/a_2 \cdot (1 - \exp(-a_2 \cdot t))$$

for $t \in [0,1]$, i.e., the same expectation as for the Ornstein-Uhlenbeck process. Hence we have

$$\mathrm{E}(f(X)) = \exp(-1) + 3 \cdot (1 - \exp(-1)) = 2.2642\ldots.$$

for our particular choice of a_1 and a_2.

From a numerical point of view, the CIR SDE has the problem, that the square-root in the diffusion coefficient b requires non-negativity. This is not warranted by the classical Euler-Maruyama scheme and hence also not by its random bit approximations. Therefore,

throughout the section, we use the modified Euler-Maruyama scheme that takes in each step the absolute value, i.e.,

$$X^\star_m(t_{0,m}) = x_0,$$
$$X^\star_m(t_{k,m}) = \big|X^\star_m(t_{k-1,m}) + m^{-1} \cdot a\big(X^\star_m(t_{k-1,m})\big) + b\big(X^\star_m(t_{k-1,m})\big) \cdot V^\star_{k,m}\big|$$

for $k = 1, \dots, m$, as suggested in Diop [20], and analogously for the coupled scheme $\widetilde{X}^\star_{m/2}$. Likewise we consider the modified random bit Euler-Maruyama schemes $X^{(q)}_m$ and $\widetilde{X}^{(q)}_{m/2}$ as well as $X^\dagger_m$ and $\widetilde{X}^\dagger_{m/2}$. Since, in this section, we only consider the modified algorithms and moreover only in this section, we do not introduce a new notation. Furthermore, we do not discuss the modified schemes. A comparison on different modified Euler-Maruyama schemes used to approximate CIR processes can, e.g., be found in Alfonsi [1]. For recent results, as well as a general overview, on strong convergence rates for CIR processes, we do refer to Hefter and Herzwurm [32]. In this article, polynomial rates of convergence are established using a Milstein-type approximation scheme that is suitably truncated close to zero.

As in the previous examples, the numerical results are presented in three separate sections. Once again, we focus on differences to the results for the previous examples, mainly on differences to the Brownian motion in Section V.1, which we discussed in detail, and on differences to the Ornstein-Uhlenbeck process in Section V.3, which is closely related to the CIR process.

V.4.1. MLMC Quadrature Based on Scheme 1. We present numerical results for the non-adaptive random bit multilevel algorithm $A^q_{\varepsilon,F}(f)$. Actually, the results are quite similar to those for the standard Brownian motion in Section V.1.1 and to those for the geometric Brownian motion in Section V.2.1. The relation between the error ϵ and the number of random generator calls seems to obey the law $c \cdot \epsilon^{-2} \cdot (\ln(\epsilon^{-1}))^{2.6}$ for some positive constant c with asymptotic confidence intervals sufficiently small to trust the empirical estimations.

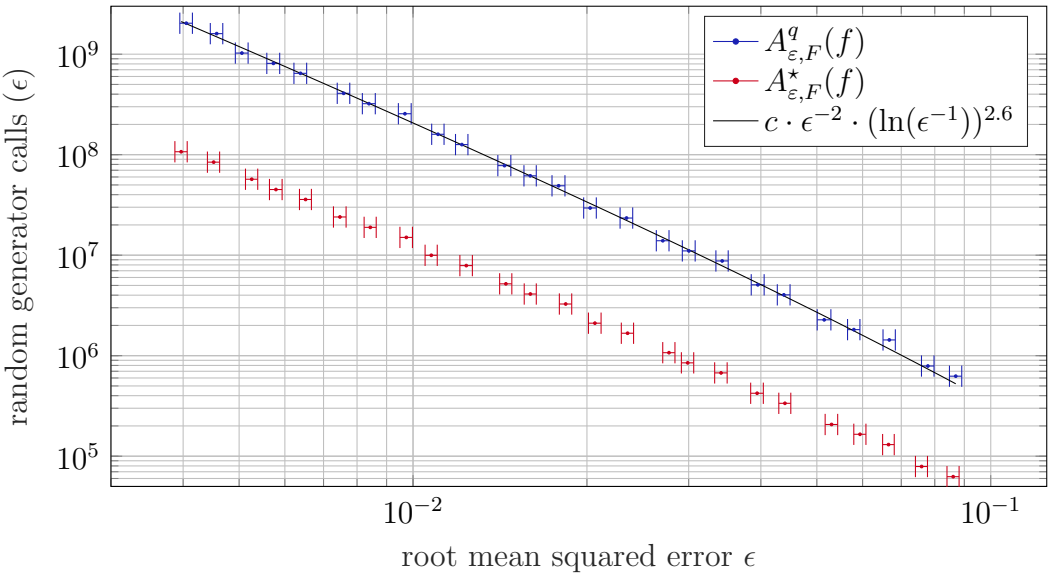

FIGURE 25. Error-cost relation of the non-adaptive MLMC Euler algorithms (random bits: blue, random numbers: red) with asymptotic confidence intervals to the confidence level 0.95. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

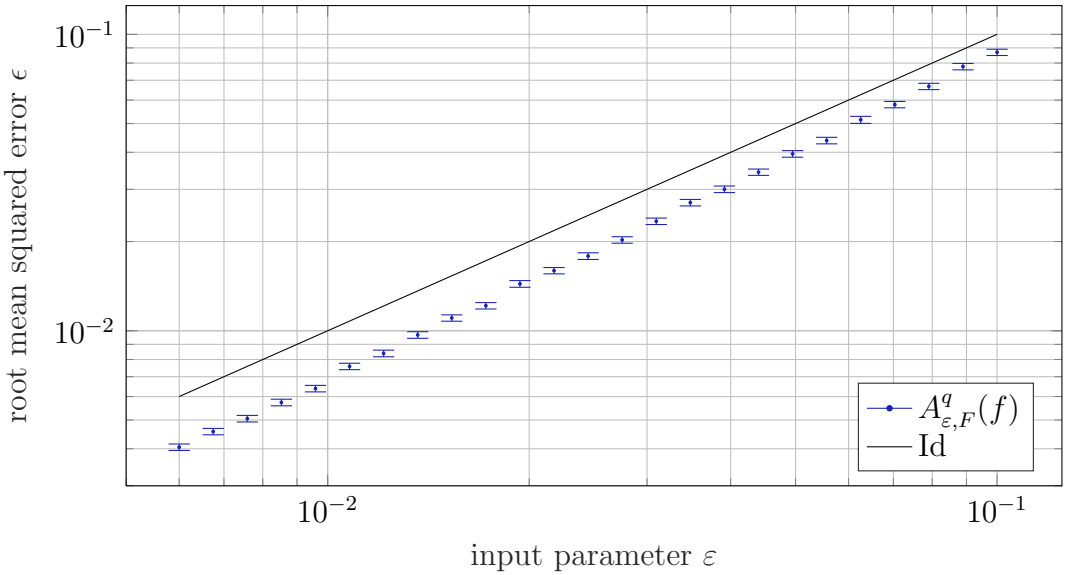

FIGURE 26. Error of the non-adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

Indeed, here this is also the case for the empirical estimation of $|\operatorname{Bias}^\bullet_\ell(f)|$ for $\bullet \in \{q, \star\}$, in difference to what we observed for the Ornstein-Uhlenbeck process in Section V.3.1, where we used the same functional f, cf. Figure 19. Following the discussion of Figure 19 this should be a consequence of the strict positivity of X, see (108).

Let us finally mention that the input parameter ε and the error ϵ of $A^q_{\varepsilon,F}(f)$ differ only by a very small factor.

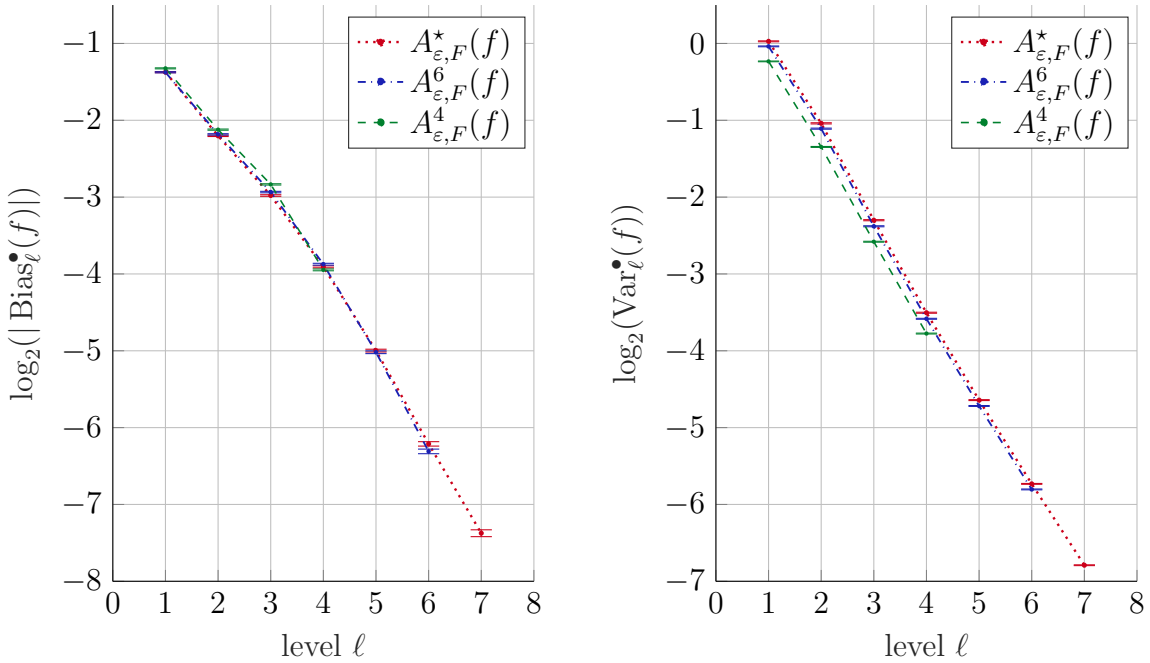

FIGURE 27. Empirical evolution of $|\operatorname{Bias}_\ell(f)|$ and $\operatorname{Var}_\ell(f)$ for different bit numbers compared to random numbers. Here, $\bullet \in \{q, \star\}$ indicates the affiliation to the particular algorithm.

V.4.2. MLMC Quadrature Based on Scheme 2. We present numerical results for the adaptive random bit multilevel algorithm $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. The results are similar to those in the previous sections, in particular to those for the Brownian motion in Section V.1.2, where we provided a detailed discussion. Hence, we solely mention that the asymptotic confidence intervals give reason to trust the empirical estimations, and that the input parameter ε and the error ϵ of $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$ do almost coincide, which is desirable in the sense that there is in particular no overestimation, which would lead to an unnecessarily long runtime of the algorithm.

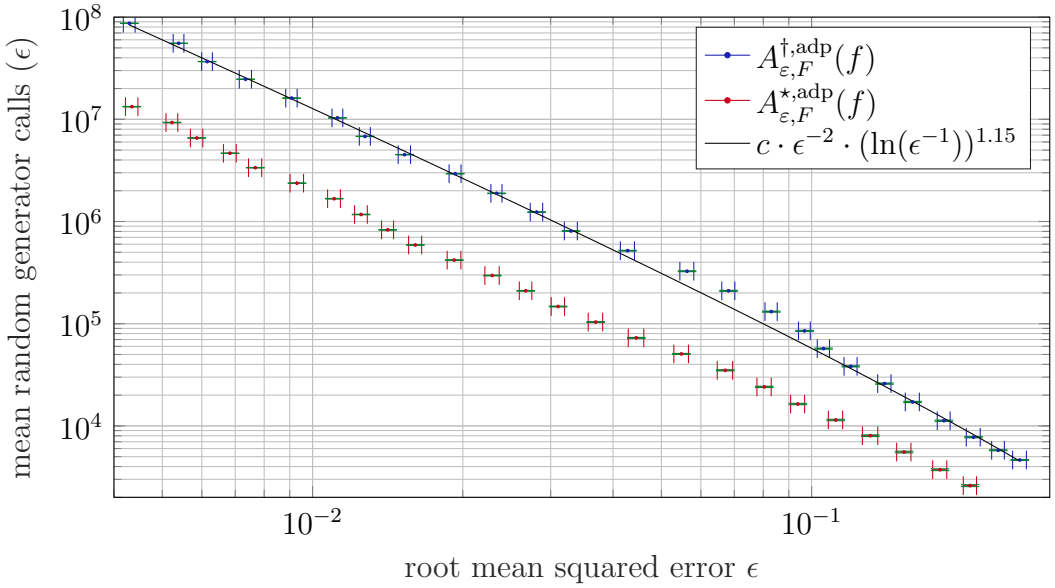

FIGURE 28. Error-cost relation of the adaptive random bit MLMC Euler algorithm (blue) and its random number counterpart (red) with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the random bit algorithm.

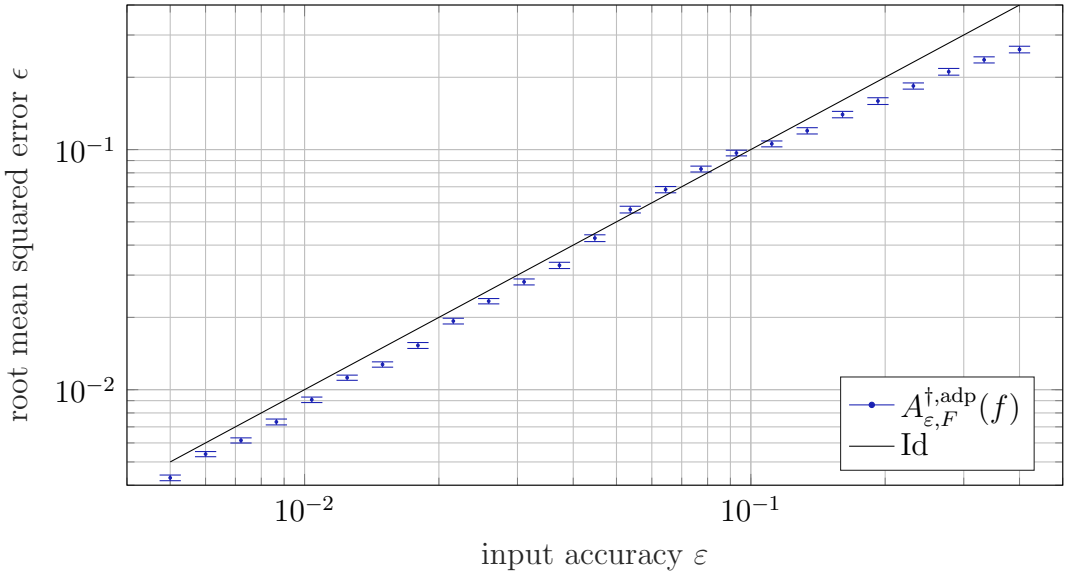

FIGURE 29. Error of the adaptive random bit MLMC Euler algorithm for given input parameter ε, with asymptotic confidence intervals to the confidence level 0.95.

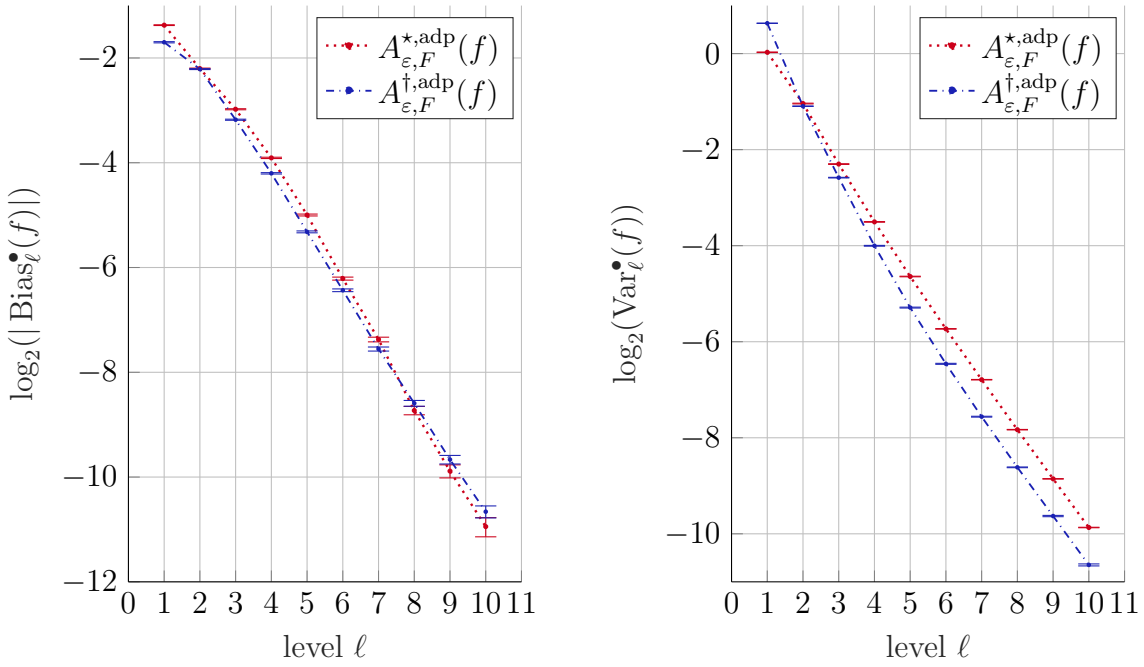

FIGURE 30. Empirical evolution of $|\,\mathrm{Bias}_\ell(f)|$ and $\mathrm{Var}_\ell(f)$ for random bits (blue) compared to random numbers (red).

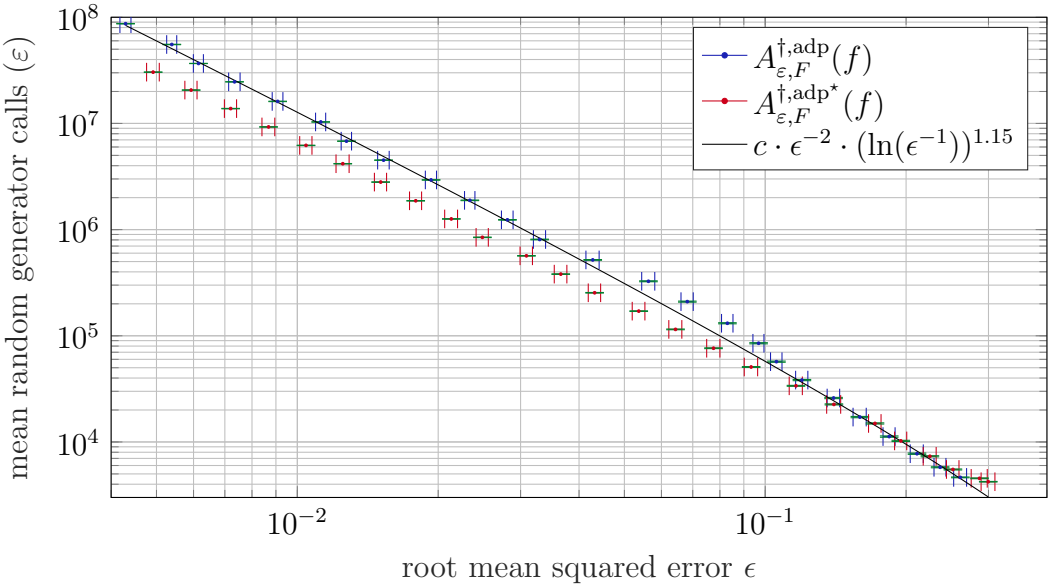

FIGURE 31. Error-cost relation of the adaptive random bit MLMC Euler algorithm without (blue) and with (red) a modified bias and variance estimation, both with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The solid line shows the supposed asymptotic behavior of the algorithms.

V.4.3. The Comparison. We compare the non-adaptive to the adaptive multilevel algorithm, i.e., $A^{q}_{\varepsilon,F}(f)$ to $A^{\dagger,\mathrm{adp}}_{\varepsilon,F}(f)$. The observed relation is similar to the observations for the Ornstein-Uhlenbeck process in Figure 24. Namely, the adaptive algorithm has a lower cost, at least in terms of the number of random generator calls, for the whole error range under consideration. Moreover, it has the better, i.e., slower growing, asymptotic logarithmic exponent of about 1.5. That is, we once again see a clear superiority of the adaptive multilevel algorithm.

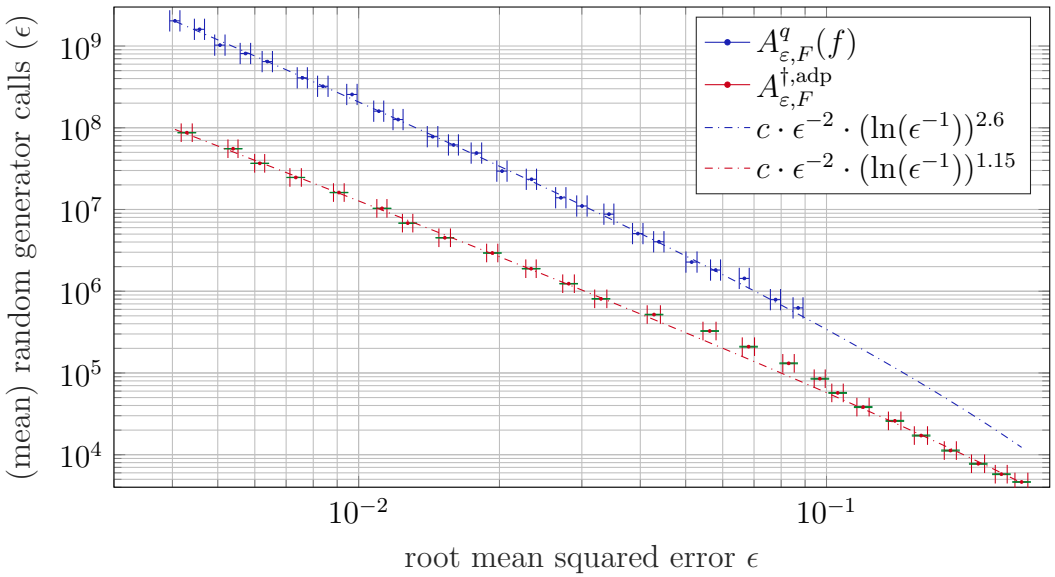

FIGURE 32. Error-cost relation of the non-adaptive (blue) and of the adaptive (red) random bit MLMC Euler algorithm with asymptotic confidence intervals to the confidence level 0.95 in both dimensions. The dashed line shows the supposed asymptotic behavior of the algorithm, respectively.

Appendix

A.1. Basic Definition and Facts Infrequently Used in the Thesis

In this section we provide the definition of a *metric projection*. Moreover, we do recall basic facts on the *inverse of the cumulative distribution function* and the *composition of convex functions*.

DEFINITION 10. Let (M, d) be a metric space and S a closed subset of M. For every $x \in M$ we define the set

$$S_x = \big\{y \in S \colon d(x, y) = \inf\{d(x, z) \colon z \in S\}\big\}$$

of elements which yield the best approximation of $x \in M$ among all elements from S. The mapping $P_S \colon M \to S, x \mapsto P_S(x) = S_x$ is called a *metric projection* of M onto S.

For a proof of the following well-known result, see, e.g., Devroye [19, Theorem 2.1].

PROPOSITION 6. *Let F be the cumulative distribution function of a probability measure μ on $\mathbb{R}$ with inverse distribution function F^{-1} given by*

$$F^{-1}(u) = \inf\big\{x \in \mathbb{R} \colon F(x) = u\big\},$$

for $0 < u < 1$. Furthermore, let U be a random variable that is uniformly distributed on the unit interval. Then we have

$$F^{-1}(U) \sim \mu,$$

i.e., $F^{-1}(U)$ has distribution function F. Conversely, if a random variable X is distributed according to μ, i.e., has distribution function F, than $F(X)$ is uniformly distributed on $[0, 1]$.

LEMMA 18. *Let $f \colon I \to \mathbb{R}$ and $g \colon \mathbb{R} \to \mathbb{R}$ be convex functions. Moreover, let $g_{|f(I)}$ be monotonically increasing. Then $g \circ f$ is a convex function on I.*

PROOF. Let $a, b \in I$ with $a < b$. Then $g(a) \leq g(b)$ due to g monotonically increasing on $f(I)$. Consequently, for $x, y \in I$ and $\lambda \in [0, 1]$ the convexity of f yields

$$g\big(f(\lambda \cdot x + (1 - \lambda) \cdot y)\big) \leq g\big(\lambda \cdot f(x) + (1 - \lambda) \cdot f(y)\big).$$

The convexity of g finishes the proof. ■

A.2. Method of Linear Least-Squares and Stability

Due to the importance of linear regressions, i.e., fitting linear functions to data, for the numerics of multilevel Monte Carlo algorithms, we present the method of linear least-square approximations, following the approach in Schwarz and Köckler [60, Section 6.1]. Since in practical applications, the data is often *noisy* we consider the connection of linear least-square fits to the condition number of a matrix, see, e.g., Schwarz and Köckler [60, Section 2.2.2]. This number can be interpreted as an indicator for the stability of the solution of the linear equation system involved in the linear least-square approximation.

Let $L, n \in \mathbb{N}$ with $L > n$. Let $s \in \mathbb{R}^L$ be a vector of statistical data, i.e., the observed values in a numerical experiment, s is often called *regressand* or *response variable*. Furthermore, let $D \in \mathbb{R}^{L \times n}$ be the *design matrix* consisting of n linearly independent columns, i.e., with full rank n, which are also called *regressors*. Our aim is to determine an n-dimensional *parameter vector* x such that x minimizes the *error equation*

$$r = D \cdot x - s,$$

where $r \in \mathbb{R}^L$ is called *residuum*. In this context, looking for a least-square approximation means to minimize the sum of the squares of the components r_i of r, i.e., we minimize $r^T r$. To do this, we observe that $x^T D^T s = s^T D x$ and hence

$$r^T r = x^T D^T D x - 2 \cdot \left(D^T s\right) x + s^T s.$$

In order to minimize $r^T r$ (with respect to x) we consider the gradient $\nabla_x(r^T r)$, which is due to the symmetry of $D^T D$ given by

$$\nabla_x(r^T r) = 2 \cdot \left(\sum_{k=1}^{n} \left(D^T D\right)_{1,k} \cdot x_k - \left(D^T s\right)_1, \ldots, \sum_{k=1}^{n} \left(D^T D\right)_{L,k} \cdot x_k - \left(D^T s\right)_L \right)^T.$$

This yields the necessary condition

$$\left(D^T D\right) \cdot x = D^T s \tag{109}$$

for x to be a minimizer of $r^T r$. Since D has rank n we know that $D^T D$ is an invertible matrix, and hence we obtain that

$$x = \left(D^T D\right)^{-1} \cdot \left(D^T s\right)$$

is an extremal point for $r^T r$. To show that x is really a minimizer, it remains to observe that the Hesse-matrix of $r^T r$ with respect to x, which is given by $2 \cdot D^T D$, is positive definite. Indeed we have for all $x \in R^n$

$$x^T D^T D x = \left(D x\right)^T \left(D x\right) \geq 0$$

and since D has full rank n, we also get $D x = 0$ if and only if $x = 0$.

Remark 42. We mention that from a computational point of view it is not necessary to invert $D^T D$ in order to compute x. Instead one can, e.g., use the Cholesky decomposition $L L^T$ of $D^T D$, recall $D^T D$ is positive definite. Then it remains to solve the linear equation systems $L \cdot y = D^T s$ and $L^T \cdot x = y$, which is quite easy due to the triangular structure of L and L^T, respectively.

As already mentioned in the beginning of this section, we would like to have some estimate on the accuracy of the approximation of x based on the data s. For the following results we refer to Schwarz and Köckler [60, Section 2.2.2]

DEFINITION 11. Let M be an invertible matrix and let $\|M\|$ denote a matrix norm of M. Then

$$\kappa(M) = \|M\| \cdot \|M^{-1}\|,$$

which can be shown to be independent from the chosen matrix norm, is called the *condition number* of M.

To use the notion of a condition number in the context of the linear equation system $D \cdot x = s$, we recall that x can be computed as the solution to $(D^T D) \cdot x = (D^T s)$, see (109). For now, we assume x to be the (correct) solution we are interested in and we let $\widetilde{x}$ denote a linear least-square approximation of x with $\widetilde{r} = D^T D\, \widetilde{x} - D^T s$. Then the relative error of $\widetilde{x}$ satisfies

$$\frac{\|x - \widetilde{x}\|}{\|x\|} \leq \kappa\big(D^T D\big) \cdot \frac{\|\widetilde{r}\|}{\|D^T s\|}$$

for any to the matrix norm compatible vector norm, due to $\|D^T s\| \leq \|D^T D\| \cdot \|x\|$ and $\|x - \widetilde{x}\| = \| - (D^T D)^{-1} \widetilde{r}\| \leq \|(D^T D)^{-1}\| \cdot \|\widetilde{r}\|$.

REMARK 43. We conclude that a small residuum vector r only implies a small relative error for $\widetilde{x}$ if the condition number $\kappa(D^T D)$ is also small. Or in other words, if the condition number $\kappa(D^T D)$ is large, then even a small perturbation in the data s and therefore in $D^T s$ may lead to a large impact on $\widetilde{x}$, i.e., a rather bad approximation $\widetilde{x}$. In that sense, the considered linear equation system is instable.

References

[1] A. Alfonsi. On the Discretization Schemes for the CIR (and Bessel Squared) Processes. *Monte Carlo Methods and Applications mcma*, 11(4):355–384, 2005.

[2] N. S. Bakhvalov. Optimal Convergence Bounds for Quadrature Processes and Integration Methods of Monte Carlo Type for Classes of Functions,(Russian). *Ž. Vyčisl. Mat. i Mat. Fiz.*, 4:5–63, 1964.

[3] D. Belomestny and T. Nagapetyan. Multilevel Path Simulation for Weak Approximation Schemes with Application to Lévy-driven SDEs. *Bernoulli*, 23(2):927–950, 2017.

[4] A. Berger and C. Xu. Best Finite Approximations of Benford's Law. *Journal of Theoretical Probability*, pages 1–29, 2018.

[5] P. Billingsley. *Probability and Measure*. Wiley series in probability and mathematical statistics. Wiley New York, 1995.

[6] J. Binia, M. Zakai, and J. Ziv. On the ε-entropy and the Rate-Distortion Function of Certain Non-Gaussian Processes. *IEEE Transactions on Information Theory*, 20(4): 517–524, 1974.

[7] C. Brugger, C. de Schryver, N. Wehn, S. Omland, M. Hefter, K. Ritter, A. Kostiuk, and R. Korn. Mixed Precision Multilevel Monte Carlo on Hybrid Computing Systems. In *Computational Intelligence for Financial Engineering & Economics (CIFEr), 2104 IEEE Conference on*, pages 215–222. IEEE, 2014.

[8] J. Bucklew and G. Wise. Multidimensional Asymptotic Quantization Theory with r-th Power Distortion Measures. *IEEE Transactions on Information Theory*, 28(2):239–247, 1982.

[9] J. Chevallier. Uniform Decomposition of Probability Measures: Quantization, Classification, Rate of Convergence. *arXiv preprint arXiv:1801.02871*, 2018.

[10] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo Methods and Applications to Elliptic PDEs with Random Coefficients. *Computing and Visualization in Science*, 14(1):3, 2011.

[11] J. C. Cox, J. E. Ingersoll Jr, and S. A. Ross. A Theory of the Term Structure of Interest Rates. *Econometrica*, 53(2):385–407, 1985.

[12] J. Creutzig, S. Dereich, T. Müller-Gronbach, and K. Ritter. Infinite-dimensional Quadrature and Approximation of Distributions. *Foundations of Computational Mathematics*, 9(4):391–429, 2009.

[13] S. Dereich. *High Resolution Coding of Stochastic Processes and Small Ball Probabilities*. PhD thesis, FU Berlin, 2003.

[14] S. Dereich. The Coding Complexity of Diffusion Processes under $L^p[0,1]$-norm Distortion. *Stochastic Processes and Their Applications*, 118(6):938–951, 2008.

[15] S. Dereich. The Coding Complexity of Diffusion Processes under Supremum Norm Distortion. *Stochastic Processes and Their Applications*, 118(6):917–937, 2008.

[16] S. Dereich. Asymptotic Formulae for Coding Problems and Intermediate Optimization Problems: A Review. In *Trends in Stochastic Analysis (J. Blath, P. Moerters, M.*

Scheutzow, eds.), pages 187–232. Cambridge University Press, 2009.

[17] S. Dereich and M. Scheutzow. High-Resolution Quantization and Entropy Coding for Fractional Brownian Motion. *Electron. J. Probab*, 11(28):700–722, 2006.

[18] S. Dereich, F. Fehringer, A. Matoussi, and M. Scheutzow. On the Link Between Small Ball Probabilities and the Quantization Problem for Gaussian Measures on Banach Spaces. *Journal of Theoretical Probability*, 16(1):249–265, 2003.

[19] L. Devroye. *Non-Uniform Random Variate Generation*. Springer, New York, 1986.

[20] A. Diop. *Sur la Discrétisation et le Comportement à Petit Bruit d'EDS Multi-dimensionnelles dont les Coefficients sont à Dérivées Singulières*. PhD thesis, École doctorale Sciences fondamentales et appliquées (Nice), 2003.

[21] E. Ekström and J. Tysk. Existence and Uniqueness Theory for the Term Structure Equation. *Preprint*, 2008.

[22] W. Gao, Y. Peixin, and W. Hui. Optimal Error Bound of Restricted Monte Carlo Integration on Anisotropic Sobolev Classes. *Progress in Natural Science*, 16(6):588–593, 2006.

[23] M. Giles, K. Debrabant, and A. Rößler. Numerical Analysis of Multilevel Monte Carlo Path Simulation Using the Milstein Discretisation. *arXiv preprint arXiv:1302.4676v2*, 2018.

[24] M. B. Giles. Multilevel Monte Marlo Path Simulation. *Operations Research*, 56(3): 607–617, 2008.

[25] M. B. Giles. Multilevel Monte Carlo Methods. *Acta Numerica*, 24:259–328, 2015.

[26] M. B. Giles. Matlab Code for Multilevel Monte Carlo Computations. https://people.maths.ox.ac.uk/gilesm/acta/, 2018.

[27] M. B. Giles, M. Hefter, L. Mayer, and K. Ritter. Random Bit Quadrature and Approximation of Distributions on Hilbert Spaces. *Foundations of Computational Mathematics*, pages 1–34, 2018.

[28] M. B. Giles, M. Hefter, L. Mayer, and K. Ritter. Random Bit Multilevel Algorithms for Stochastic Differential Equations. *arXiv:1808.10623*, 2018.

[29] S. Graf and H. Luschgy. *Foundations of Quantization for Probability Distributions*. Lecture Notes in Mathematics; 1730. Springer, Berlin, 2000.

[30] A. Haar. Zur Theorie der Orthogonalen Funktionensysteme. *Mathematische Annalen*, 69:331–371, 1910.

[31] A.-L. Haji-Ali, F. Nobile, and R. Tempone. Multi-index Monte Carlo: When Sparsity Meets Sampling. *Numerische Mathematik*, 132(4):767–806, 2016.

[32] M. Hefter and A. Herzwurm. Strong Convergence Rates for Cox–Ingersoll–Ross Processes — Full Parameter Range. *Journal of Mathematical Analysis and Applications*, 459(2):1079–1101, 2018.

[33] S. Heinrich. Monte Carlo Complexity of Global Solution of Integral Equations. *Journal of Complexity*, 14(2):151–175, 1998.

[34] S. Heinrich. Multilevel monte carlo methods. In S. Margenov, J. Waśniewski, and P. Yalamov, editors, *Large-Scale Scientific Computing*, pages 58–67. Springer, 2001.

[35] S. Heinrich and E. Sindambiwe. Monte Carlo Complexity of Parametric Integration. *Journal of Complexity*, 15(3):317–341, 1999.

[36] S. Heinrich, E. Novak, and H. Pfeiffer. How Many Random Bits Do We Need for Monte Carlo Integration? In *Monte Carlo and Quasi-Monte Carlo Methods 2002*, pages 27–49. Springer, 2004.

[37] S. L. Heston. A Closed-form Solution for Options with Stochastic Volatility with Applications to Bond and Currency Options. *The review of financial studies*, 6(2):327–343, 1993.

[38] O. Kallenberg. *Foundations of Modern Probability*. Springer Series in Statistics. Probability and Its Application. Springer, New York, 1997.

[39] I. Karatzas and S. Shreve. *Brownian Motion and Stochastic Calculus*. Springer, New York, 1988.

[40] D. Lamberton and B. Lapeyre. *Introduction au Calcul Stochastique Appliqué à la Finance*. Ellipses Marketing, 2012.

[41] H. Luschgy and G. Pagès. Functional Quantization of Gaussian Processes. *Journal of Functional Analysis*, 196(2):486–531, 2002.

[42] H. Luschgy and G. Pagès. Functional Quantization of a Class of Brownian Diffusions: A Constructive Approach. *Stochastic Processes and their Applications*, 116(2):310–336, 2006.

[43] H. Luschgy and G. Pagès. Sharp Asymptotics of the Functional Quantization Problem for Gaussian Processes. *The Annals of Probability*, 32(2):1574–1599, 2004.

[44] X. Mao. *Stochastic Differential Equations and Applications*. Elsevier, 2007. Second Edition.

[45] G. N. Milstein and M. V. Tretyakov. *Stochastic Numerics for Mathematical Physics*. Springer Berlin, 2004.

[46] P. Mörters and Y. Peres. *Brownian Motion*, volume 30. Cambridge University Press, 2010.

[47] E. H. Müller, R. Scheichl, and T. Shardlow. Improving Multilevel Monte Carlo for Stochastic Differential Equations with Application to the Langevin Equation. *Proceedings of the Royal Society A*, 471(2176), 2015.

[48] T. Müller-Gronbach and K. Ritter. A Local Refinement Strategy for Constructive Quantization of Scalar SDEs. *Foundations of Computational Mathematics*, 13(6):1005–1033, 2013.

[49] T. Müller-Gronbach, E. Novak, and K. Ritter. *Monte Carlo-Algorithmen*. Springer-Verlag, 2012.

[50] E. Novak. Eingeschränkte Monte Carlo-Verfahren zur Numerischen Integration. In *Proc. of the 4th Pannonian Symp. on Math. Stat., Bad Tatzmannsdorf, Austria, W. Grossmann et al.(eds.)*, pages 269–282, 1983.

[51] E. Novak. *Deterministic and Stochastic Error Bounds in Numerical Analysis*. Lecture Notes in Mathematics; 1349. Springer, Berlin, 1988.

[52] E. Novak. The Real Number Model in Numerical Analysis. *Journal of Complexity*, 11 (1):57–73, 1995.

[53] E. Novak. Quantum Complexity of Integration. *Journal of Complexity*, 17(1):2–16, 2001.

[54] E. Novak and H. Pfeiffer. Coin Tossing Algorithms for Integral Equations and Tractability. *Monte Carlo Methods and Applications*, 10(3-4):491–498, 2004.

[55] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, Berlin, 1998.

[56] S. Omland. *Mixed Precision Multilevel Monte Carlo Algorithms for Reconfigurable Hardware Systems*. PhD thesis, Technische Universität Kaiserslautern, 2016.

[57] S. Omland, M. Hefter, K. Ritter, C. Brugger, C. De Schryver, N. Wehn, and A. Kostiuk. Exploiting Mixed-precision Arithmetics in a Multilevel Monte Carlo Approach on FPGAs. In *FPGA Based Accelerators for Financial Applications*, pages 191–220. Springer, 2015.

[58] D. Pollard. Quantization and the Method of k-means. *IEEE Transactions on Information Theory*, 28(2):199–205, 1982.

[59] K. Ritter. *Average-Case Analysis of Numerical Problems*. Lecture Notes in Mathematics; 1733. Springer, Berlin, 2000.

[60] H.-R. Schwarz and N. Köckler. *Numerische Mathematik*. Springer-Verlag, 2007.

[61] S. E. Shreve. *Stochastic Calculus for Finance II: Continuous-time Models*, volume 11. Springer Science & Business Media, 2004.

[62] C. P. Simon and L. Blume. *Mathematics for Economists*, volume 7. Norton New York, 1994.

[63] H. M. Taylor and S. Karlin. *An Introduction to Stochastic Modeling*, volume 3. Gulf Professional Publishing, 1998.

[64] J. F. Traub and H. Woźniakowski. The Monte Carlo Algorithm with a Pseudorandom Generator. *Mathematics of Computation*, 58(197):323–339, 1992.

[65] U. F. Wiersema. *Brownian Motion Calculus*. John Wiley & Sons, 2008.

[66] A. Winkelbauer. Moments and Absolute Moments of the Normal Distribution. *arXiv preprint arXiv:1209.4340*, 2012.

[67] C. Xu and A. Berger. Best Finite Constrained Approximations of One-dimensional Probabilities. *arXiv preprint arXiv:1704.07871*, 2017.

[68] P. Ye and X. Hu. Optimal Integration Error on Anisotropic Classes for Restricted Monte Carlo and Quantum Algorithms. *Journal of Approximation Theory*, 150(1):24–47, 2008.

Scientific career

2009	Abitur Christian-Wirth-Schule Usingen
2009 – 2104	Study in Mathematics Technische Universität Kaiserslautern (Oktober 2012: Bachelor of Science) (Dezember 2014: Master of Science)
2015	Starting Ph.D. program in Mathematics Technische Universität Kaiserslautern Group of Prof. Dr. Klaus Ritter
2015 – 2017	Member DFG-Graduiertenkolleg 1932

Wissenschaftlicher Werdegang

2009	Abitur Christian-Wirth-Schule Usingen
2009 – 2104	Studium Mathematik Technische Universität Kaiserslautern (Oktober 2012: Bachelor of Science) (Dezember 2014: Master of Science)
2015	Beginn Promotion in Mathematik Technische Universität Kaiserslautern Arbeitsgruppe Prof. Dr. Klaus Ritter
2015 – 2017	Mitglied DFG-Graduiertenkolleg 1932