Reinhold Kainhofer

reinhold@kainhofer.com
Matrikelnummer 96 13 474

Quasi-Monte Carlo Algorithms with Applications in Numerical Analysis and Finance

Dissertation*

Ausgeführt zum Zwecke der Erlangung
des akademischen Grades eines
Doktors der technischen Wissenschaften

Betreuer: O. Univ.-Prof. Dr. Robert F. Tichy
Eingereicht an der Technischen Universität Graz
Technisch-Naturwissenschaftliche Fakultät

Graz, April 2003

*Supported by the Austrian Science Fund (FWF) FSP-Project S 8308-MAT

Abstract

This thesis is devoted to the development and application of various Quasi-Monte Carlo methods for numerical integration and also for the solution of differential equations. In contrast to Monte Carlo schemes, they employ deterministic sequences with good distribution properties. This has the effect that explicit error bounds can be shown, and the numerical error usually is improved compared to Monte Carlo methods.
First, a dividend barrier model from risk theory is investigated, where the dividend payments and the survival probability can be described by integro-differential equations. Several different schemes for their solution are presented and compared.
Second, a Quasi-Monte Carlo algorithm for the solution of retarded differential equations is developed. While for slowly changing equations conventional methods perform better, for heavily oscillating equations Quasi-Monte Carlo schemes become competitive and might even be applied in unstable regions where conventional schemes fail.
Finally, the problem of numerical integration of singular functions with respect to a given density is explored. A convergence theorem is proved, and an adapted construction schemes for non-uniformly distributed low-discrepancy sequences is presented. As a numerical example from finance the valuation of an Asian option is investigated. Several different scheme for the singular non-uniform integration are compared, and again the Quasi-Monte Carlo approach turns out to be the most beneficial.

Zusammenfassung

Diese Dissertation widmet sich der Entwicklung diverser Quasi-Monte Carlo (QMC) Verfahren zur numerischen Integration sowie zur Lösung von Differentialgleichungen. Im Gegensatz zu Monte Carlo Verfahren basieren diese auf deterministischen Folgen mit guten Verteilungseigenschaften. Dadurch können explizite Fehlerschranken angegeben und numerische Fehler deutlich verbessert werden.

Der erste Teil beschäftigt sich mit einem Modell einer Dividendenschranke in der Risikotheorie, wobei die Dividenden und die Überlebenswahrscheinlichkeit durch Integro-Differentialgleichungen beschrieben werden. Verschiedene Schemata zu deren Lösung werden präsentiert und verglichen.

Im zweiten Teil wird ein QMC Algorithmus für retardierte Differentialgleichungen entworfen. Während für sich langsam ändernde Gleichungen konventionelle Runge-Kutta Verfahren bessere Ergebnisse liefern, können Quasi-Monte Carlo Methoden bei schnell oszillierenden Gleichungen teilweise sogar in Bereichen angewendet werden, wo konventionelle Verfahren versagen.

Im dritten Teil wird das Problem der numerischen Integration von singulären Funktionen bezüglich beliebiger Dichten behandelt. Neben einem Konvergenzbeweis wird eine Folgenkonstruktion von Hlawka and Mück adaptiert. Als Beispiel aus der Finanzwissenschaft dient die Bewertung asiatischer Optionen, wobei mehrere Methoden zur Auswertung des singulären Integrals verglichen werden. Auch hier zeigt sich, dass der QMC Zugang am vorteilhaftesten ist.

Contents

Chapter 1

Introduction

For the solution of high-dimensional integration problems conventional numerical integration techniques like the trapezoidal rule quickly become infeasible as the quality of the error bounds decreases drastically with the number of dimensions. To some extent the occurrence of Monte Carlo methods in the middle of the last century, which employ random numbers to approximate the integral, was able to cure this problem, but for these methods only stochastic error bounds can be given. The error bounds of order $\mathcal{O}\left(\frac{1}{\sqrt{N}}\right)$, where N denotes the number of nodes used in the calculation, however, no longer depend on the dimension of the problem, and the numerical errors can be improved considerably compared to conventional methods. Thus for high-dimensional integration Monte Carlo methods quickly became state-of-the-art tools.
Although random numbers are used in Monte Carlo methods, eventually it became clear that it is not the randomness of the employed point sets, but rather their distribution properties that determine the quality of the numerical calculation. For this reason, Quasi-Monte Carlo methods were suggested, where the used sequence no longer contains any randomness, but follows a clear construction scheme. This has several positive effects, namely that

- a calculated result can be exactly reproduced at any time (which is not possible when employing random numbers),
- strict error bounds can be given, and
- these error bounds for the best known sequences have a better asymptotic order, namely $\mathcal{O}\left(\frac{\log^s N}{N}\right)$, than Monte Carlo methods.

The aim of this thesis is to develop such Quasi-Monte Carlo algorithms for several problems that frequently occur in various disciplines, with particular focus on problems from finance, natural sciences and engineering.

The work is structured as follows:

First I will give a short introduction to Quasi-Monte Carlo methods in Chapter 2, where the most important definitions and well-known theorems from discrepancy theory are recalled, as well as the most frequently used low-discrepancy sequences. That chapter is not meant as a comprehensive introduction to Quasi-Monte Carlo methods, but rather to provide only the necessary tools which will be applied later in the remainder of the thesis.

The thesis covers three larger topics, one from risk theory, a second from numerical analysis, and the final one from finance.

In particular, in the first part of this work (Chapters 3 and 4) we will investigate the numerical solution of integro-differential equations at the example of dividend strategies in ruin theory. We model an insurance company with claims that follow some distribution, and if the surplus process exceeds some (parabolic) threshold, dividends are paid out to the shareholders. The resulting integro-differential equations for the expected dividend payments and the survival probability cannot be solved analytically (like in the case of linear dividend barriers), so we iterate these operators and employ Quasi-Monte Carlo integration techniques to the resulting high-dimensional integrals. In Chapter 4 we extend the model to include interest on the surplus, and in a separate model we introduce an absorbing upper barrier for the surplus process. The results presented in these chapters were obtained in collaboration with H. Albrecher and R. Tichy.

The topic of the second part (Chapters 5 and 6) will be the application of Quasi-Monte Carlo techniques to heavily oscillating delayed differential equations (DDE). These are differential equations that do not only involve the function $y(t)$ and its derivatives at time t, but also depend on past values $y(t-\tau(t))$. Such equations appear for example in electronics when one tries to describe the oscillations in circuits with feedback.
For delayed differential equations several solution methods are known, most of which apply Runge-Kutta schemes that extrapolate the new value of the solution at time t from very few nodes at previous time steps. However, if the equation oscillates heavily, a few nodes cannot ensure a good extrapolation, and so the conventional methods fail. Instead of taking only a few function values, we will do a Quasi-Monte Carlo integration over the whole relevant interval so that the oscillations average out and the error stays small. The dependence of the differential equations on past values of the solution will be solved by the use of Hermite interpolation.
As we shall see, for slowly varying equations our Quasi-Monte Carlo method clearly cannot compete with conventional Runge-Kutta schemes, but for heavily varying delay differential equations the Runge-Kutta QMC methods we develop might even be applied in regions where conventional methods become unstable.

The third part of this work is dedicated to the problem of integrating singular functions with respect to a measure other than the uniform distribution. Sobol [64] already treated the uniform QMC integration of functions with a singularity at one vertex of the integration interval. In Chapter 7 this problem is generalized to the evaluation of an integral $\int f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})$, where f is allowed to have singularities at the boundary of the integration domain and H denotes an arbitrary probability density function. Convergence of Quasi-Monte Carlo integration is shown under conditions on the function $f(\mathbf{x})$ and the point sequence that is used for the QMC integration. Additionally, a construction scheme for suitable sequences is presented, which fulfill these conditions better than a scheme proposed by Hlawka and Mück [41, 42, 43].
Finally, in Chapter 8 this method is applied to the problem of valuing an Asian option. Therein, one is faced with two problems: One being the need to generate non-uniformly distributed variates, and the other being the singular integrand function. Several different ways to evaluate the integral are investigated, and the results from Chapter 7 are applied to

enable the use of non-uniform Quasi-Monte Carlo integration of the singular function. These results were obtained in collaboration with J. Hartinger, M. Predota, and R. Tichy.

All theoretical results in this work are accompanied by numerical examples, where it will become clear that Quasi-Monte Carlo methods are superior to other methods in most cases. We will also compare some popular low-discrepancy sequences, the Halton, Sobol and Faure sequences, as well as Niederreiter's (t, s)-nets. In some cases we will see that there is hardly any difference in their performance, while in other cases their results differ considerably (but perform still better then crude Monte Carlo methods).
All parts of this thesis were submitted for publication in mathematical journals, most of them are already published.

Graz, April 2003 Reinhold Kainhofer

Chapter 2

Quasi-Monte Carlo methods

Contents

In this chapter I will quickly lay out the notations as well as several basic facts about Quasi-Monte Carlo methods and low-discrepancy sequences, which will be applied in the rest of this work. This short introduction is not meant to be a comprehensive survey of this area at the intersection of number theory and numerical mathematics, but rather to provide the reader with those facts that are necessary for the further treatment in the remaining chapters.
Often, Quasi-Monte Carlo (or QMC in short) methods are referred to as deterministic versions of Monte Carlo schemes. The use of deterministic uniformly distributed point sequences instead of pseudo-random sequences in crude Monte Carlo has proven to be an efficient extension of the classical Monte Carlo method.
I will particularly deal with problems, where the desired quantity can be expressed as an integral, or the solution of an integral equation. In all these cases, the numerical evaluation of the integral through Quasi-Monte Carlo methods will turn out to be beneficial compared to conventional as well as Monte Carlo methods.

2.1 Monte Carlo methods

In Monte Carlo integration, the idea is to approximate the integral (possibly after a transformation to the desired unit interval) as the arithmetic mean of a number N of sample points $\mathbf{x}_n \in [0,1]^s$, $1 \leq n \leq N$:

$$\int_{[0,1]^s} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \approx \frac{1}{N}\sum_{n=1}^{N} f(\mathbf{x}_n).$$

In contrast to conventional integration techniques, the nodes $\mathbf{x}_n$ are chosen randomly in the unit interval, and the quality of the estimator obtained by this technique depends heavily on the quality of the random number generator. The most important advantage of Monte Carlo

methods over conventional techniques is the (stochastic) error bound of $\mathcal{O}\left(\frac{1}{\sqrt{N}}\right)$, which does not depend on the dimension of the integration problem. Thus the "curse of dimensionality" in conventional methods, where the number of nodes grows exponentially with the dimension, is removed in Monte Carlo methods. Drawbacks of this method are, however, the stochastic error bound, and the fact that a Monte Carlo result cannot be exactly (due to the use of random numbers).

2.2 Quasi-Monte Carlo integration

If one can improve the integration error by using random but well distributed sample nodes, shouldn't it be possible to obtain even better results by using specifically chosen sequences, which are as uniformly distributed as possible? Or in other words, is the randomness of the nodes in Monte Carlo methods really needed, and aren't the distribution properties of the sequence the important factor in this techniques?
This line of thought leads to the Quasi-Monte Carlo approach, where the random sequences are replaced by deterministic sequences, which exhibit very good distribution properties. Typically, these are net-like constructions.

2.2.1 Discrepancy

A well-known measure for the uniformness of the distribution of a sequence $\{\mathbf{x}_n\}_{1\leq n\leq N}$ in $U^s := [0,1)^s$ is the *star-discrepancy*

$$D_N^*(\mathbf{x}_n) = \sup_{I\in J_0^s} \left| \frac{A(\mathbf{x}_n; I)}{N} - \lambda_s(I) \right| , \tag{2.1}$$

where J_0^s is the set of all intervals of the form $[0,\vec{y}) = [0,y_1)\times[0,y_2)\times\ldots\times[0,y_s)$ with $0 \leq y_i < 1$, $i = 1,\ldots,s$ and $A(\mathbf{x}_n; I)$ is the number of points of the sequence $\{\mathbf{x}_n\}_{1\leq n\leq N}$ that lie in I. $\lambda_s(I)$ denotes the s-dimensional Lebesgue-measure of I.
The star discrepancy just describes the maximum error one can get if we calculate the volume of an arbitrary interval containing the origin. Another widely used discrepancy is the *total discrepancy* $D_N(\mathbf{x}_n)$, where the maximum in (2.1) is extended over all subintervals of $[0,1]^s$. Clearly, these two discrepancy types are connected by the inequality

$$D_N^*(\mathbf{x}_N) \leq D_N(\mathbf{x}_N) \leq 2^s D_N^*(\mathbf{x}_N)\,.$$

To measure the distribution properties of a given sequence with respect to measures other than the uniform distribution, this definition of discrepancy can be generalized in a straightforward way. This will be done in Chapter 7, where we will take a look at QMC integration with respect to arbitrary measures.

2.2.2 Error bounds: Koksma-Hlawka's inequality

The notion of discrepancy is particularly useful for obtaining an upper bound for the error of quasi-Monte Carlo integration:

Lemma 2.1 (Koksma-Hlawka Inequality). *Let the function $f : [0,1)^s \to \mathbb{R}$ be of bounded variation $V_{[0,1)^s}(f)$ in the sense of Hardy and Krause. Then for any point set $\{x_1, \ldots, x_N\} \subset [0,1)^s$*

$$\left| \frac{1}{N} \sum_{n=1}^{N} f(x_n) - \int_{[0,1)^s} f(u)du \right| \leq V([0,1)^s, f) D_N^* (x_1, \ldots, x_N) \ . \tag{2.2}$$

For a proof of this famous inequality we refer to [24] and [51]. This error bound is deterministic (opposed to error bounds obtainable for crude Monte Carlo). Especially for s not too large, certain Quasi-Monte Carlo sequences have turned out to be superior to pseudo-Monte Carlo sequences in many applications.
Again, this inequality can easily be generalized to arbitrary densities (e.g. in [70] for general distributions H, but using a slight restriction to variation in the measure sense), and it will be the basis for the convergence theorem of QMC integration of singular integrands in Chapter 7.

2.2.3 Low-discrepancy point sets and sequences

This is in particular the case for so-called *low discrepancy sequences*, i.e. sequences for which

$$D_N^* (x_1, \ldots, x_N) \ \leq C_s \frac{(\log N)^s}{N}, \tag{2.3}$$

with an explicitly computable constant C_s, holds. It is conjectured, but not yet proven, that the order of this bound in N is best possible. Bounds for C_s are usually pessimistic and often the actual error made by Quasi-Monte Carlo integration is much lower than the bound implied by C_s (see e.g. [16]). The dependence on s of the constant C_s varies for the known bounds, and for a detailed investigation of the dependence of multi-dimensional integration on s we refer to [54].

Some low discrepancy sequences will be given in the sequel:

- The *Halton sequence* [35] is defined as a sequence of vectors in U^s based on the digit representation of n in base p_i

  $$\xi_n = (b_{p_1}(n), b_{p_2}(n), \ldots, b_{p_s}(n)), \tag{2.4}$$

  where the p_i are pair-wise coprime nubmers and $b_p(n)$ is the digit reversal function for base p given by

  $$b_p(n) = \sum_{k=0}^{\infty} n_k p^{-k-1}, \qquad n = \sum_{k=0}^{\infty} n_k p^k,$$

  where the n_k are integers. Halton [35] showed a discrepancy bound of

  $$D_N^*(S) < \frac{s}{N} + \frac{1}{N} \prod_{i=1}^{s} \left(\frac{p_i - 1}{2 \ln p_i} \ln N + \frac{p_i + 1}{2} \right)$$

  for the Halton sequence S in pairwise co-prime bases $p_1, \ldots, p_s \geq 2$. According to this bound, the smallest discrepancy is obtained by using the first s prime numbers.

Better error bounds can be obtained for low-discrepancy sequences based on so-called (t, m, s)-nets or nets for short. These nets are based on the b-adic representation of vectors in U^s.

Instead of optimizing the discrepancy itself, one considers the discrepancy with respect to elementary intervals J in base b only, i.e. $J = \prod_{i=1}^{s}[a_i b^{-d_i}, (a_i+1)b^{-d_i})$ with integers $d_i \geq 0$ and integers $0 \leq a_i < b^{d_i}$ for $1 \leq i \leq s$, and tries to construct point sequences in U^s such that the discrepancy with respect to these intervals J is optimal for subsequences of length $N = b^m$.
Let $\#(J, N)$ denote the number of points of a sequence $\{\mathbf{x}_n\}_{1 \leq n \leq N}$ that lie in J. A point set $\mathcal{P}$ with $\text{card}(\mathcal{P}) = b^m$ is now called a (t, m, s)-net, if

$$\#(J, b^m) = b^t$$

for every elementary interval J with $\lambda_s(J) = b^{t-m}$. The parameter t is a quality parameter. For $t = 0$ we have the minimal discrepancy of the point set $\mathcal{P}$ with respect to the family of elementary intervals.

Definition: Let $t \geq 0$ be an integer. A sequence $\xi_1, \xi_2, \ldots$ of points in U^s is called a (t, s)-sequence in base b, if for all integers $k \geq 0$ and $m > t$, the point set consisting of the ξ_n with $kb^m < n \leq (k+1)b^m$ is a (t, m, s)-net in base b.
The discrepancy of the (t, s)-sequence is minimal for t being as small as possible, but unfortunately, $(0, s)$-nets do not exist for all bases b. See for example [53] for lower bounds of t for given pairs (s, b).
Examples of (t, m, s)-nets are

- *Sobol sequences* are (t, s)-sequences in base 2 with values for t that depend on s. They were initially proposed by Sobol [63], but other choices for the direction numbers used in the construction have also been proposed (e.g. [57] for the application of a high-dimensional Sobol sequence).

- *Faure sequences* [26] are $(0, s)$-sequences in a base b which is the smallest prime number fulfilling $b \geq s$. The s-dimensional Faure sequence is defined by

  $$\left\{\phi_b(n), P(\phi_b(n)), \ldots, P^{s-1}(\phi_b(n))\right\}$$

  where the function P is defined for a b-adic rational $x \in [0, 1]$ with expansion $x = \sum_{j=0}^{\infty} x_j b^{-j-1}$ as $P(x) = \sum_{j=0}^{\infty} \xi(x_j) b^{-j-1}$ with $\xi(x_j) = \sum_{i \geq j} \binom{i}{j} x_i \mod b$.

- *Niederreiter sequences* [53] are a generalization of Sobol's and Faure's sequences and yield general (t, s)-sequences for arbitrary bases p with the restrictions on t mentioned above. Among them there are $(0, s)$-sequences in prime power bases $b \geq s$. In particular, for Niederreiter sequences the constant C_s in (2.3) tends to zero for $s \to \infty$.

  For various different constructions see [53].

Part I

QMC Solutions of Iterated Integral Operators for Sublinear Dividend Models in Risk Theory

Chapter 3

Risk theory with a non-linear dividend barrier

Contents

In the framework of classical risk theory we investigate a surplus process in the presence of a non-linear dividend barrier and derive equations for two characteristics of such a process, the probability of survival and the expected sum of discounted dividend payments. Number-theoretic solution techniques are developed for approximating these quantities and numerical illustrations are given for exponential claim sizes and a parabolic dividend barrier.
This chapter is based on a joint work [2] with H. Albrecher.

3.1 Introduction

Let us consider the classical risk process $R_t = u + c\,t - \sum_{i=1}^{N(t)} X_i$, where c is a constant premium intensity, $N(t)$ denotes a homogeneous Poisson process with intensity λ which counts the claims up to time t and the claim amounts X_i are iid random variables with distribution function $F(y)$. In this context R_t represents the surplus of an insurance portfolio at time t (for an introduction to classical risk theory see for instance Gerber [29] and Thorin [67] or more recently Asmussen [8]). As usual we assume $\mu = E(X_i) < \infty$ and $c > \lambda \int_0^\infty y\, dF(y)$. A reasonable modification of this model is the introduction of a dividend barrier b_t, i.e. whenever the surplus R_t reaches b_t, dividends are paid out to the shareholders with intensity $c - db_t$

and the surplus remains on the barrier, until the next claim occurs. This means that the risk process develops according to

$$dR_t = c\,dt - dS_t \quad \text{if} \quad R_t < b_t \tag{3.1}$$

$$dR_t = db_t - dS_t \quad \text{if} \quad R_t = b_t, \tag{3.2}$$

where we have used the abbreviation $S_t = \sum_{i=1}^{N(t)} X_i$. Together with the initial capital $R_0 = u$, $0 \leq u < b_0 < \infty$, this determines the risk process $\{R_t,\, t \geq 0\}$ (cf. Figure 3.1).

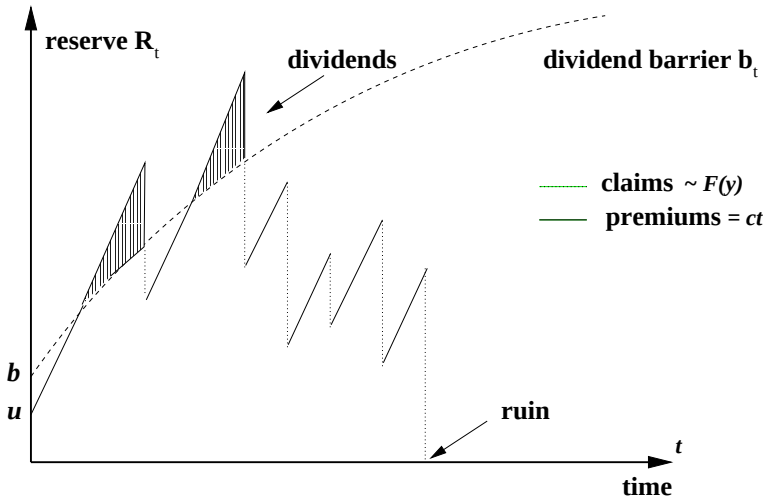

Figure 3.1: *A typical sample path of* R_t

Two quantities of particular interest in this context are the probability of survival $\phi(u,b) = Pr\,(R_t \geq 0\ \forall t > 0 | R_0 = u, b_0 = b)$ (or alternatively the probability of ruin $\psi(u,b) = 1 - \phi(u,b)$) and the expected sum of discounted dividend payments $W(u,b)$.

Dividend barrier models have a long history in risk theory (see e.g. [19, 15, 29]). For a survey on the relation between dividend payments and tax regulations we refer to [7, 10].
Gerber [28] showed that barrier dividends constitute a complete family of Pareto-optimal dividends. In the case of a horizontal dividend barrier $b_t \equiv b_c$ =const., it is easy to see that $\phi(u,b) = 0\ \forall\ 0 \leq u \leq b$. Segerdahl [61] used the technique of integro-differential equations to derive the characteristic function of the time to ruin in the presence of a horizontal dividend barrier for exponentially distributed claims. This approach was generalized by Gerber and Shiu [31]. Paulsen and Gjessing [58] calculated the optimal value of b_c that maximizes the expected value of the discounted dividend payments in an economic environment. Recently Irbäck [44] developed an asymptotic theory for a high horizontal dividend barrier.
If one allows for monotonically increasing b_t in the model, a positive probability of survival can be achieved. The case of linear dividend barriers is fairly well understood: Gerber [27] derived an upper bound for the probability of ruin for $b_t = b + at$ by martingale methods and in [30] he obtained exact solutions for the probability of ruin and the expected sum of discounted dividend payments $W(u,b)$ in terms of infinite series in the case of exponentially distributed claim amounts. This result was generalized to arbitrary Erlang claim amount distributions in Siegl and Tichy [62] by developing a suitable solution algorithm. The convergence of this algorithm was proved by Albrecher and Tichy [5].

Apart from mathematical simplicity there is no compelling reason to restrict the model to linear dividend barriers. Moreover, simulations indicate that by choosing an appropriate dividend barrier, the expected value of discounted dividend payments $W(u,b)$ can be increased, while the probability of survival $\phi(u,b)$ stays constant (cf. ALEGRE ET AL. [6]).

In this chapter non-linear dividend barrier models are investigated. In Section 3.2 we derive integro-differential equations for $\phi(u,b)$ and $W(u,b)$ and discuss the existence and uniqueness of the corresponding solutions. Our main focus is on the development of efficient numerical algorithms to obtain those quantities. More precisely, we adapt number-theoretic solution methods in the spirit of [68] to the current situation (Section 3.3). Finally Section 3.4 gives numerical results for the special case of a parabolic dividend barrier and exponential claim amount distributions.

3.2 The model

Model A: We consider a classical risk process extended by a dividend barrier of type

$$b_t = \left(b^m + \frac{t}{\alpha}\right)^{1/m} \qquad (\alpha, b > 0, m > 1).$$

Note that $m = 1$ corresponds to the linear barrier case.
The probability of survival $\phi(u,b)$ can then be expressed as a boundary value problem in the following way: Conditioning on the occurrence of the first claim, we get for $u < b$

$$\phi(u,b) = (1-\lambda dt)\phi\left(u + c\,dt, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) + \\ + \lambda\,dt \int_0^{u+c\,dt} \phi\left(u + c\,dt - z, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) dF(z). \quad (3.3)$$

Taylor series expansion of the functions ϕ on the right-hand side of (3.3) and division by dt shows that ϕ satisfies the equation

$$c\,\frac{\partial \phi}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\,\frac{\partial \phi}{\partial b} - \lambda\,\phi + \lambda \int_0^u \phi(u-z,b)dF(z) = 0, \quad (3.4)$$

which, for reasons of continuity, is valid for $0 \le u \le b$. For $u = b$ we get along the same line of arguments

$$\phi(u,b) = (1-\lambda dt)\phi\left(\left(b^m + \frac{dt}{\alpha}\right)^{1/m}, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) + \\ + \lambda\,dt \int_0^{\left(b^m+\frac{dt}{\alpha}\right)^{1/m}} \phi\left(\left(b^m + \frac{dt}{\alpha}\right)^{1/m} - z, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) dF(z), \quad (3.5)$$

from which it follows that

$$\frac{1}{\alpha m\, b^{m-1}}\,\frac{\partial \phi}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\,\frac{\partial \phi}{\partial b} - \lambda\,\phi + \lambda \int_0^u \phi(u-z,b)dF(z) = 0. \quad (3.6)$$

Comparing (3.4) and (3.6) we thus obtain the boundary condition

$$\left.\frac{\partial \phi}{\partial u}\right|_{u=b} = 0. \quad (3.7)$$

A further natural requirement is

$$\lim_{b\to\infty} \phi(u,b) = \phi(u), \tag{3.8}$$

where $\phi(u)$ is the probability of survival in absence of the barrier.

Contrary to ruin, the crossing of the dividend barrier is a much desired event. For equal slopes of the barrier at time 0, the expected time until the first crossing of the dividend barrier will be considerably less for sub-linear barriers as introduced above than for the linear case. A quantitative result in this direction follows from BOOGAERT ET AL. [12] who used a martingale technique to derive upper bounds for the probability $Pr(D > t)$ that the surplus process does not reach a given barrier before time t. Adapting these results to our situation, we obtain

$$Pr(D > t) \leq \frac{\lambda\mu\, t}{u - (b^m + t/\alpha)^{1/m} + ct}$$

for all t that satisfy $u + ct > (b^m + t/\alpha)^{1/m}$.

Let furthermore $W(u,b)$ denote the expected present value of the future dividend payments, which are discounted with a constant intensity δ, and stop when ruin occurs. Then, in a similar way to (3.3) and (3.5), one can derive the integro-differential equation

$$c\frac{\partial W}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\frac{\partial W}{\partial b} - (\delta+\lambda)\, W + \lambda\int_0^u W(u-z,b)dF(z) = 0, \tag{3.9}$$

with boundary condition

$$\frac{\partial W}{\partial u}\Big|_{u=b} = 1. \tag{3.10}$$

In the actuarial literature [27, 69] there has been some interest in models where dividends can also be paid after a ruin event (this makes sense since ruin of a portfolio is a technical term used in decision making and does not necessarily imply bankruptcy). If we allow for dividend payments after ruin in our model, then along the same line of arguments as above, we obtain the following equation for the expected value $V(u,b)$ of the discounted dividend payments

$$c\frac{\partial V}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\frac{\partial V}{\partial b} - (\delta+\lambda)\, V + \lambda\int_0^\infty V(u-z,b)dF(z) = 0, \tag{3.11}$$

and the initial condition $\frac{\partial V}{\partial u}\Big|_{u=b} = 1$. Note that for a linear dividend barrier the corresponding integro-differential equation was much simpler, because V could be expressed as a function of one variable only (cf. [69]); for a non-linear barrier this is no longer the case.

Model B: In addition to Model A, we will also consider a "finite-horizon" version of the model, namely we introduce an absorbing upper barrier $b_{max} \equiv const.$ If the surplus process $R_t \geq b_{max}$ for some $t > 0$, it is absorbed and the company is considered to have survived. From an economic point of view this can be interpreted that the company will then decide to pursue other forms of investment strategies. Mathematically, this model has some nice features (e.g. the process stops in finite time with probability 1). The boundary value problem for the probability of survival can now be formulated by (3.4), (3.7) and

$$\phi(u, b_{max}) = \frac{\phi(u)}{\phi(b_{max})}, \tag{3.12}$$

where $0 \leq u \leq b \leq b_{max}$ and as before $\phi(u)$ is the probability of survival in absence of the barrier.

Example: For exponentially distributed claim amounts ($F(z) = 1 - e^{-z}$), equation (3.4) can be expressed as a hyperbolic partial differential equation with variable coefficients

$$c\frac{\partial^2\phi}{\partial u^2} + \frac{1}{\alpha\, m\, b^{m-1}}\,\frac{\partial^2\phi}{\partial b\,\partial u} + (c-\lambda)\frac{\partial\phi}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\,\frac{\partial\phi}{\partial b} = 0 \tag{3.13}$$

and with boundary conditions (3.7) and

$$\left(c\,\frac{\partial\phi}{\partial u} + \frac{1}{\alpha\, m\, b^{m-1}}\,\frac{\partial\phi}{\partial b} - \lambda\,\phi\right)\Bigg|_{u=0} = 0. \tag{3.14}$$

Since $\phi_0(u,b) = e^{-sb^m - r(s)u}$ is a solution of (3.13), where $r(s)$ satisfies

$$cr^2 + \Big(s/\alpha + \lambda - c\Big)r - s/\alpha = 0, \tag{3.15}$$

one can try to obtain a solution of the form

$$\phi(u,b) = \int_0^\infty e^{-sb^m} A_1(s) e^{-r_1(s)u}\,ds + \int_0^\infty e^{-sb^m} A_2(s) e^{-r_2(s)u}\,ds + \phi(u),$$

where $r_1(s), r_2(s)$ are the solutions of (3.15) and the $A_i(s)$ have to be determined according to (3.7) and (3.14). However, this turns out to be an intricate problem.
Similarly, the integro-differential equations for $W(u,b)$ and $V(u,b)$ can be expressed as second-order PDE's in the case of exponentially distributed claims.

3.3 Solution techniques

The above example shows that even for the simple case of exponentially distributed claim amounts it is a delicate problem to obtain analytical solutions. Thus there is a need for effective algorithms to obtain numerical solutions to these problems. Here we focus on the development of number-theoretic solution methods.

Following a procedure developed by GERBER [30] for the case of linear barriers, we first show that the boundary value problem (3.9) together with (3.10) has a unique bounded solution. For that purpose, we define an operator A by

$$\begin{aligned} Ag(u,b) = &\int_0^{t^*} \lambda e^{-(\lambda+\delta)t} \int_0^{u+ct} g\left(u+ct-z, \left(b^m + \frac{t}{\alpha}\right)^{1/m}\right) dF(z)dt + \\ &+ \int_{t^*}^\infty \lambda e^{-(\lambda+\delta)t} \int_0^{\left(b^m+\frac{t}{\alpha}\right)^{1/m}} g\left(\left(b^m+\frac{t}{\alpha}\right)^{1/m} - z, \left(b^m+\frac{t}{\alpha}\right)^{1/m}\right) dF(z)dt + \\ &+ \int_{t^*}^\infty \lambda e^{-\lambda t} \int_{t^*}^t e^{-\delta s}\left(c - \frac{1}{m\alpha\left(b^m+\frac{s}{\alpha}\right)^{1-1/m}}\right) ds\,dt. \end{aligned} \tag{3.16}$$

Here t^* is the positive solution of $u + ct = \left(b^m + \frac{t}{\alpha}\right)^{1/m}$ (since $m > 1$, b_t is concave and $u \leq b$, so this number is unique). Note that (3.16) can be interpreted as a conditioning on whether

a claim occurs before the surplus process hits the dividend barrier ($t < t^*$) or after this event (in which case we have an additional term representing the discounted dividends paid until the claim occurs). The solution $W(u,b)$ of (3.9) with its initial condition (3.10) is a fixed point of the integral operator A. For any two bounded functions g_1, g_2

$$|Ag_1(u,b) - Ag_2(u,b)| \leq \|g_1 - g_2\|_\infty \int_0^\infty \lambda e^{-(\lambda+\delta)t} dt \leq \frac{\lambda}{\lambda+\delta} \|g_1 - g_2\|_\infty \tag{3.17}$$

for arbitrary $0 \leq u \leq b < \infty$, where $\|\cdot\|_\infty$ is the supremum norm on $0 \leq u \leq b < \infty$, and thus it follows that A is a contraction and the fixed point is unique by Banach's theorem.

Proceeding in the same way as for $W(u,b)$ above, one can easily show that equation (3.11) together with its initial condition has a unique bounded solution.

In the case of Model B we can proceed in a similar way to obtain a contraction map for the probability of survival as its fixed point: Like in equation (3.16), let t^* be the time when the surplus would reach the dividend barrier given that no claim occurs. Let furthermore $t^{**} = \alpha(b_{max}^m - b^m)$ be the time when the dividend barrier reaches the absorbing barrier, and $\tilde{t} = (b_{max} - u)/c$ the time when the surplus would reach the absorbing barrier in the absence of a dividend barrier and of claims. As the dividend barrier is an increasing function on $\mathbb{R}^+$, t^{**} is uniquely determined, just as is $\tilde{t}$. Combining the two possible scenarios $0 \leq t^{**} \leq \tilde{t} \leq t^*$ and $0 \leq t^* \leq \tilde{t} \leq t^{**}$ (depending on the values of u and b), we define the operator A as

$$A\phi(u,b) = \int_0^T \lambda e^{-\lambda t} \int_0^{z_{min}(u,b,t)} \phi\left(z_{min}(u,b,t) - z, \left(b^m + \frac{t}{\alpha}\right)^{\frac{1}{m}}\right) dF(z)\, dt + e^{-\lambda T}, \tag{3.18}$$

where $T = \max\left(\tilde{t}, t^{**}\right)$ is the time when the surplus process would reach the absorbing upper barrier b_{max}, and

$$z_{min}(u,b,t) = \min\left(u + ct, \left(b^m + \frac{t}{\alpha}\right)^{\frac{1}{m}}\right). \tag{3.19}$$

Let ϕ_1 and ϕ_2 now be two bounded functions on $0 \leq u \leq b \leq b_{max}$, then

$$|A\phi_1(u,b) - A\phi_2(u,b)| \leq \|\phi_1 - \phi_2\|_\infty \int_0^T \lambda e^{-\lambda t} dt = \|\phi_1 - \phi_2\|_\infty \left(1 - e^{-\lambda T}\right).$$

Since $T = T(u,b) < M < \infty$, this operator is a contraction, and Banach's fixed point theorem establishes the existence and uniqueness of the solution. Here, the absorbing barrier and the resulting restriction to the finite area $0 \leq u \leq b \leq b_{max}$ ensures that the solution is unique in contrast to the case without the absorbing barrier.

Correspondingly, the contraction map for the expected sum of dividend payments in Model B is given by

$$\begin{aligned} Ag(u,b) = &\int_0^{t^*} \lambda e^{-(\lambda+\delta)t} \int_0^{u+ct} g\left(u + ct - z, \left(b^m + \frac{t}{\alpha}\right)^{1/m}\right) dF(z) dt + \\ &+ \int_{t^*}^{t^{**}} \lambda e^{-(\lambda+\delta)t} \int_0^{\left(b^m + \frac{t}{\alpha}\right)^{1/m}} g\left(\left(b^m + \frac{t}{\alpha}\right)^{1/m} - z, \left(b^m + \frac{t}{\alpha}\right)^{1/m}\right) dF(z) dt + \\ &+ \int_{t^*}^{t^{**}} e^{-(\lambda+\delta)t} \left(c - \frac{1}{m\alpha\left(b^m + \frac{t}{\alpha}\right)^{1-1/m}}\right) dt, \end{aligned} \tag{3.20}$$

if $t^{**} > t^*$ and $Ag(u,b) = 0$ otherwise, because then the surplus reaches the absorbing barrier before the dividend barrier. The last term in (3.20) represents the dividends that are paid out until t^{**} and is a simplification of the original expression

$$\int_{t^*}^{t^{**}} \lambda e^{-\lambda t} \int_{t^*}^{t} e^{-\delta s} \left(c - \frac{1}{m\alpha \left(b^m + \frac{s}{\alpha}\right)^{1-1/m}} \right) ds\, dt + \int_{t^{**}}^{\infty} \lambda e^{-\lambda t} \int_{t^*}^{t^{**}} e^{-\delta s} \left(c - \frac{1}{m\alpha \left(b^m + \frac{s}{\alpha}\right)^{1-1/m}} \right) ds\, dt.$$

From (3.20) it follows that

$$\|Ag_1(u,b) - Ag_2(u,b)\|_\infty \leq \frac{\lambda}{\lambda+\delta} \left(1 - e^{-(\lambda+\delta)t^{**}}\right) \|g_1 - g_2\|_\infty \ ,$$

for any two bounded functions g_1, g_2 and we again have a contraction in the Banach space of bounded functions equipped with the supremum norm, which implies the existence and uniqueness of the solution.

The following algorithms are now efficient ways of approximating the corresponding fixed point:

3.3.1 Double-recursive algorithm

This procedure will be described for the operator (3.16); it can easily be adapted to the other integral operators introduced above. Moreover we will restrict ourselves to the case of exponentially distributed claim amounts (with parameter γ); the extension of the method to other distributions is straightforward.
The fixed point of (3.16) can be approximated by applying the contracting integral operator A k times to a starting function $h(u,b)$ which we choose to be the inhomogeneous term in the corresponding integral operator (where k is chosen according to the desired accuracy of the solution):

$$g^{(k)}(u,b) = A^k g^{(0)}(u,b),$$
$$g^{(0)}(u,b) = h(u,b) := \int_{t^*}^{\infty} \lambda e^{-\lambda t} \int_{t^*}^{t} e^{-\delta s} \left(c - \frac{1}{m\alpha \left(b^m + \frac{s}{\alpha}\right)^{1-\frac{1}{m}}} \right) ds\, dt.$$

This leads to a $2k$-dimensional integral for $g^{(k)}(u,b)$, which is calculated numerically using Monte Carlo and Quasi-Monte Carlo methods. For that purpose we transform the integration domain of operator (3.16) into the unit cube:

$$\begin{aligned} Ag(u,b) &= h(u,b) + \\ &+ \frac{\lambda}{\lambda+\delta} \Bigg[\left(1 - e^{-(\lambda+\delta)t^*}\right) \int_0^1 \int_0^1 g\left(u + ct_1 - z_1, \left(b^m + \frac{t_1}{\alpha}\right)^{\frac{1}{m}}\right) \left(1 - e^{-\gamma(u+ct_1)}\right) dv_1 dw_1 \\ &+ e^{-(\lambda+\delta)t^*} \int_0^1 \int_0^1 g\left(\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}} - z_2, \left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}} \right) \cdot \left(1 - e^{-\gamma\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}}}\right) dv_2 dw_2 \Bigg] \end{aligned}$$

with

$$t_1 = -\frac{\log\left(1 - w_1\left(1 - e^{-(\lambda+\delta)t^*}\right)\right)}{\lambda+\delta} \qquad z_1 = -\frac{\log\left(1 - v_1\left(1 - e^{-\gamma(u+ct_1)}\right)\right)}{\gamma} \tag{3.21}$$

$$t_2 = t^* - \frac{\log(1-w_2)}{(\lambda+\delta)} \qquad z_2 = -\frac{\log\left(1 - v_2\left(1 - e^{-\gamma\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}}}\right)\right)}{\gamma}. \tag{3.22}$$

The Monte Carlo-estimator of $W(u,b)$ for given values of u and b is

$$W(u,b) \approx \frac{1}{N}\sum_{n=1}^{N} g_n^{(k)}(u,b)\ , \tag{3.23}$$

where the $g_n^{(k)}(u,b)$ are calculated recursively for each n by

$$g_n^{(0)}(u,b) = h(u,b)$$

and

$$\begin{aligned} g_n^{(i)}(u,b) &= h(u,b) + \frac{\lambda}{\lambda+\delta}\cdot \\ &\cdot\left\{\left(1 - e^{-\gamma(u+ct_{1,n}^i)}\right)\left(1 - e^{-(\lambda+\delta)t^*}\right) g_n^{(i-1)}\left(u + ct_{1,n}^i - z_{1,n}^i, \left(b^m + \frac{t_{1,n}^i}{\alpha}\right)^{\frac{1}{m}}\right) + \right. \\ &+\left.\left(1 - e^{-\gamma\left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}}}\right) e^{-(\lambda+\delta)t^*} g_n^{(i-1)}\left(\left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}} - z_{2,n}^i, \left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}}\right)\right\}. \end{aligned}$$

Here $t_{j,n}^i$ and $z_{j,n}^i$ $(j = 1,2)$ are determined according to (3.21) and (3.22) for (quasi-)random deviates v_j, w_j of the uniform distribution in the unit interval $(1 \le i \le k)$.

Since in every recursion step the function g is called twice, the number of evaluations of g doubles in every recursion step. Thus, in order to keep the computations tractable, in what we will call the double-recursive algorithm in the sequel, the double recursion is only used for the first two recursive steps and for the remaining recursion steps the recursive algorithm described in Section 3.3.2 is applied.

3.3.2 Recursive algorithm

Instead of calculating the first two integrals occurring in operator (3.16) separately, one can combine them to one integral. A suitable change of variables then leads to

$$\begin{aligned} Ag(u,b) &= h(u,b) + \\ &\int_0^1\int_0^1 \frac{\lambda}{\lambda+\delta}\left(1 - e^{-\gamma z_{min}(u,b,t)}\right) g\left(z_{min}(u,b,t) - z, \left(b^m + \frac{t}{\alpha}\right)^{\frac{1}{m}}\right) dvdw \end{aligned} \tag{3.24}$$

where t and z are given by

$$\begin{aligned} t &= -\frac{\log(1-w)}{(\lambda+\delta)} \\ z &= -\frac{\log\left(1 - v\left(1 - e^{-\gamma z_{min}(u,b,t)}\right)\right)}{\gamma} \end{aligned} \tag{3.25}$$

and $z_{min}(u,b,t)$ is determined by (3.19). Like in the double recursive case, this integral operator is now applied k times onto $g^{(0)}$, and the resulting multidimensional integral $g^{(k)}(u,b)$ is again approximated by

$$g^{(k)}(u,b) \approx \frac{1}{N}\sum_{n=1}^{N} g_n^{(k)}(u,b), \tag{3.26}$$

where each $g_n^{(k)}(u,b)$ $(n = 1,\ldots,N)$ is based on a pseudo-random (or quasi-random, resp.) point $\mathbf{x}_n \in [0,1]^{2k}$ and calculated by the recursion

$$g_k^{(0)}(u,b) = h(u,b),$$

$$g_n^{(i)}(u,b) = \frac{\lambda}{\lambda+\delta}\left(1 - e^{-\gamma z_{min}(u,b,t_n^i)}\right) g_n^{i-1}\left(z_{min}(u,b,t_n^i) - z_n^i, \left(b^m + \frac{t_n^i}{\alpha}\right)^{\frac{1}{m}}\right) + h(u,b),$$

with $1 \le i \le k$. t_n^i and z_n^i are given by (3.25) with v and w being the value of the $2i$-th and $2i+1$-th, component of $\mathbf{x}_n$, respectively. Note that for this algorithm, the number of integration points needed for a given recursion depth is one fourth of the corresponding number required for the double-recursive case.

3.3.3 Simulation

Since there are no analytical solutions available for the above problems, we need simulation estimates of the ruin probabilities and discounted dividend payments to compare them to the results of the integration methods that were described in the last sections.
We sample N paths of the risk reserve process in the following way: Starting with $t_0 := 0$, $b_0 := b$ and $x_0 := u$, where u is the initial reserve of the insurance company, we first generate an exponentially distributed random variable $\tilde{t}_i$ with parameter λ for the time until the next claim occurs and set $t_{i+1} := t_i + \tilde{t}_i$. The claim amount is sampled from an exponentially distributed random variable z_i (with parameter γ), and the reserve after the claim is $x_{i+1} := \min\{x_i + c\tilde{t}_i, (b_i^m + \tilde{t}_i/\alpha)^{1/m}\} - z_i$. Due to the structure of the dividend barrier, we can reset the origin to t_{i+1} in every step, if we also set $b_{i+1} = \left(b_i^m + \frac{\tilde{t}_i}{\alpha}\right)^{1/m}$. We then have to discount the dividend payments between the i-th and $(i+1)$-th claims by the factor $e^{-\delta t_i}$.

A simulation estimate for the survival probability $\phi(u,b)$ can now be obtained by

$$\phi(u,b) \approx \frac{m}{N},$$

where m is the number of paths for which ruin does not occur (i.e. $x_i > 0\ \forall\ i$). We consider a path as having survived, if for some i the condition $x_i > x_{max}$ is fulfilled, where x_{max} is a sufficiently large threshold. This can be viewed as an absorbing horizontal barrier at x_{max}, and so the process stops with probability 1. Using this stopping criterion, we overestimate the actual probability of survival $\phi(u,b)$; for sufficiently large x_{max}, however, this effect is negligible.

For the simulation of the expected value of the dividend payments, we proceed as described above and whenever the process reaches the dividend barrier, i.e. $x_i + c\tilde{t}_i > (b_i^m + \tilde{t}_i/\alpha)^{\frac{1}{m}}$,

we need to calculate the amount of dividends that are paid until the next claim i occurs:

$$v_i := v_{i-1} + e^{-\delta t_i} \int_{t^*}^{\tilde{t}_i} e^{-\delta s} \left(c - \frac{1}{m\alpha \left(b_i^m + \frac{s}{\alpha}\right)^{1-\frac{1}{m}}} \right) ds, \qquad i \geq 1$$

and $v_0 = 0$, where t^* is the positive solution of $x_i + ct = \left(b_i^m + \frac{t}{\alpha}\right)^{1/m}$, i.e. the time when the process reaches the dividend barrier. The process is stopped, if ruin occurs (i.e. $x_i < 0$ for some i) or at some sufficiently large time t_{max}, after which the expected value of discounted dividends becomes negligible due to the discount factor $e^{-\delta t}$. Let $v(j)$ now be the final value of v_i for path j. The expected value of the dividends is then approximated by

$$E[W(u,b)] \approx \frac{1}{N} \sum_{j=1}^{N} v(j) \; .$$

3.3.4 Quasi-Monte Carlo approach

Following a technique developed in [68], we can now use the Koksma-Hlawka inequality (2.1) to find an upper bound for the error of the recursive algorithm estimate introduced in Section 3.3.2 in terms of the discrepancy of the sequence used:

Theorem 3.1. *If the expected value $W(u,b)$ of the discounted dividends is approximated by $g^{(k)}(u,b)$ as given in (3.23) using a sequence ω of N elements, the error is bounded by*

$$\left\| W(u,b) - g^{(k)}(u,b) \right\|_\infty \leq \frac{\|h(u,b)\|_\infty}{1-q} \left(q^k + q D_N(\omega) \right) \tag{3.27}$$

with $q := \frac{\lambda}{\lambda+\delta}$.

Proof. Since we have $g^{(0)}(u,b) = h(u,b)$, it follows from Banach's fixed point theorem together with the estimate (3.17), that

$$\begin{aligned} \left\| W(u,b) - g^{(k)}(u,b) \right\|_\infty &\leq \left\| W(u,b) - A^k h(u,b) \right\|_\infty + \left\| A^k h(u,b) - g^{(k)}(u,b) \right\|_\infty \\ &\leq \frac{q^k}{1-q} \|h(u,b)\|_\infty + \left\| A^k h(u,b) - g^{(k)}(u,b) \right\|_\infty \end{aligned} \tag{3.28}$$

Iterating the integral equation (3.24) k times leads to

$$A^k h(u_0,b_0) = \sum_{i=1}^{k} \int_{[0,1]^{2i}} \cdots \int \left(\prod_{j=0}^{i-1} C_j q \right) h(u_i,b_i) dv_{i-1} dw_{i-1} \ldots dv_0 dw_0 + h(u_0,b_0) \tag{3.29a}$$

$$= \int_{[0,1]^{2k}} \cdots \int \sum_{i=1}^{k} \left(\prod_{j=0}^{i-1} C_j q \right) h(u_i,b_i) dv_{k-1} dw_{k-1} \ldots dv_0 dw_0 + h(u_0,b_0) \tag{3.29b}$$

where for $0 \leq j \leq k-1$ we have

$$\begin{aligned} t_j &= -\frac{1}{\lambda}\log(1-w_j), \\ z_j &= -\frac{1}{\gamma}\log\left(1 - v_j\left(1 - e^{-\gamma z_{min}(u_j,b_j,t_j)}\right)\right), \\ z_{min}(u_j,b_j,t_j) &= cut_j := \min\left(u_j + ct_j, \left(b_j^m + \frac{t_j}{\alpha}\right)^{\frac{1}{m}}\right), \\ C_j &= \left(1 - e^{-\gamma cut_j}\right), \\ u_{j+1} &= cut_j - z_j, \\ b_{j+1} &= \left(b_j^m + \frac{t_j}{\alpha}\right)^{\frac{1}{m}}. \end{aligned} \tag{3.30}$$

In our recursive algorithm the $2k$-dimensional integral (3.29b) is approximated by quasi-Monte Carlo integration and in order to use Koksma-Hlawka's inequality for bounding the error, we have to determine the total variation of the integrand in (3.29b). For that purpose we investigate each of the summands separately and define F_i to be the integrand of the i-th term in (3.29a):

$$F_i(v_0, w_0, \ldots, v_{i-1}, w_{i-1}) := q^i \prod_{j=0}^{i-1}\left(1 - e^{-\gamma z_{min}(u_j,b_j,t_j)}\right) h(u_i, b_i). \tag{3.31}$$

We now show that this function is increasing in all the variables w_j and decreasing in all the variables v_j $(j = 1, .., i-1)$:
Choose a $j \in \{1, .., i-1\}$ and let v_j be increasing (while all the other variables are fixed), then C_k, u_k, and t_k remain constant for all $k \leq j$. Furthermore z_k remains constant for all $k < j$ and so does b_k for arbitrary k. But then it is easy to see that u_{j+1} and $z_{min}(u_{j+1}, b_{j+1}, t_{j+1})$ are decreasing. By induction and some elementary monotonicity investigations it follows that u_{j+k} and $z_{min}(u_{j+k}, b_{j+k}, t_{j+k})$ are decreasing for all $k \geq 1$. But since $h(u,b)$ is an increasing function of u it follows from (3.31) that F_i is a decreasing function of v_j $(j = 1, .., i-1)$. Similarly it can be shown that F_i is an increasing function of w_j $(j = 1, .., i-1)$.
This monotone behavior now allows to bound the variation of F_i:

$$V([0,1)^{2i}; F_i) = F_i(0, 1, \ldots, 0, 1) - F_i(1, 0 \ldots, 1, 0) \leq \left(\frac{\lambda}{\lambda+\delta}\right)^i \|h\|_\infty$$

By summing up the variations of the F_i we get an upper bound for the total variation of the integrand F of (3.29b)

$$V([0,1]^{2k}; F) \leq \sum_{i=1}^{k}\left(\frac{\lambda}{\lambda+\delta}\right)^i \|h\|_\infty = q\frac{1-q^k}{1-q}\|h\|_\infty \leq \frac{q}{1-q}\|h\|_\infty .$$

If we use this estimate together with Lemma 2.1 we get

$$\left\|A^k h(u,b) - g^{(k)}(u,b)\right\|_\infty \leq \|h\|_\infty \frac{q}{1-q} D_N(\omega)$$

and inserting this into equation (3.28) finally gives

$$\left\|W(u,b) - g^{(k)}(u,b)\right\|_\infty \leq \frac{q^k}{1-q}\|h(u,b)\|_\infty + \|h\|_\infty \frac{q}{1-q} D_N(\omega) = \frac{\|h\|_\infty}{1-q}\left(q^k + qD_N(\omega)\right).$$

□

3.4 Numerical results for the parabolic case

In this section numerical illustrations for a parabolic dividend barrier of the form $b_t = \sqrt{b^2 + t/\alpha}$ and exponentially distributed claim amounts ($F(z) = 1 - e^{-z}$) are presented. Note that in this case

$$t^* = \frac{1}{2\alpha c^2} - \frac{u}{c} + \sqrt{\left(\frac{1}{2\alpha c^2} - \frac{u}{c}\right)^2 + \frac{b^2 - u^2}{c^2}}$$

and the inhomogeneous term $h(u,b)$ in (3.16) can be calculated explicitly to

$$h(u,b) = e^{-t^*(\lambda+\delta)} \left(\frac{c}{\lambda+\delta} - \sqrt{\frac{\pi}{(\lambda+\delta)\alpha}}\, \frac{e^{z^2}}{2}\, \mathrm{erfc}\,(z) \right)$$

with $z = \sqrt{(\lambda+\delta)(\alpha b^2 + t^*)}$ and thus we have $\|h(u,b)\|_\infty \leq \frac{c}{\lambda+\delta}$.

The parameters are set to $c = 1.5$, $\delta = 0.1$, $\alpha = 0.5$, $\lambda = \gamma = 1$ and the absorbing upper barrier in Model B is chosen at $b_{\max} = 4$.

The MC and QMC estimators are obtained using $N = 66\,000$ paths for the recursive case and for the simulation and $N = 33\,000$ for the double-recursive calculations. The corresponding "exact" value, in lack of an analytic solution, is obtained by a MC-simulation over 10 million paths for each choice of u and b.
For the recursive and double recursive calculations we use a recursion depth of $k = 66$, which leads to a 132-dimensional sequence needed for the MC- and QMC-calculations, while for the simulation we take a 400-dimensional sequence so that 200 consecutive claims and interoccurrence times of a risk reserve sample path can use the different dimensions of one element of the sequence and correlations among the claim sizes and claim occurrence times are avoided.

We use so-called hybrid Monte Carlo sequences for all our QMC-calculations, where the initial 50 dimensions are generated by a 50-dimensional low discrepancy sequence and the remaining dimensions are generated by a pseudo-random number generator. Throughout this chapter, we use `ran2` as our pseudo-random number generator, which basically is an improved version of a Minimal Standard generator based on a multiplicative congruential algorithm (for a description we refer to [59]). The use of hybrid Monte Carlo sequences has proven to be a successful modification of the QMC-technique, since for low discrepancy sequences typically the number of points needed to obtain a satisfying degree of uniformness dramatically increases with the number of dimensions.

The different methods and sequences used are compared via the mean square error

$$S = \sqrt{\frac{1}{|P|} \sum_{(u,b)\in P} \Big(g(u,b) - \tilde{g}(u,b)\Big)^2},$$

where $g(u,b)$ and $\tilde{g}(u,b)$ denote the exact and the approximated value, respectively, and the set P is a grid in the triangular region ($b = 0..[0.1]..1, u = 0..[0.1]..b$). In addition, for each method we give the maximal deviation of the approximated value from the corresponding exact value $\|\Delta\|_\infty = \max_{(u,b)\in P} \Big(g(u,b) - \tilde{g}(u,b)\Big)$.

3.4.1 Survival probability

In Model A the survival probability can only be calculated using the simulation approach. Table 3.1 gives the mean-square and the maximal error of the simulation results (together with the corresponding calculation time in seconds) for each of the sequences used (with $N = 66\,000$):

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Simulation	S	0.001307	0.001798	0.001706	0.0009
	$\|\Delta\|_\infty$	0.003741	0.003619	0.003472	0.002451
		(163.16 s)	(149.58 s)	(281.61 s)	(150.09 s)

Table 3.1: *Simulation errors for the survival probability in Model A*

Figure 3.2 shows a log-log-plot of the mean square error S as a function of N. To quantify the effect of using a low discrepancy sequence, we perform a regression analysis by fitting

$$\log_2(S) = a_0 + a_1 \log_2(N) + a_2 \log_2(\log_2(N)) + \varepsilon$$

to the data using a least square fit. Note that Koksma-Hlawka's inequality (2.1) could be interpreted as implying $a_1 = -1$ and $a_2 = s$, where s is the dimension of the sequence used. However, since we use a hybrid sequence and since the effective dimension may differ from the theoretical dimension, the values of a_1 and a_2 deviate from the ones above. Figure 3.3 gives these fitted curves. In the sequel all figures on simulation results will be given in terms of their regression fits.

In Model B approximate solutions for the survival probability can be obtained by the recursive method using the operator (3.18) and by simulation. The numerical errors and the corresponding calculation time are given in Table 3.3 and the fitted curves for the mean square error are depicted in Figure 3.4.

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Simulation	S	0.001796	0.000676	0.001621	0.00062
	$\|\Delta\|_\infty$	0.004066	0.001813	0.002529	0.001217
		(99.71 s)	(86.92 s)	(87.21 s)	(86.91 s)
Recursive	S	0.000934	0.000155	0.000168	0.000128
	$\|\Delta\|_\infty$	0.002504	0.000365	0.000392	0.000317
		(386.44 s)	(374.3 s)	(374.4 s)	(374.21 s)

Table 3.3: *Errors for the survival probability in Model B*

Figure 3.4 shows that while the recursive Monte Carlo method is favorable to the Monte Carlo simulation, for larger values of N the simulation technique using the Halton and the Sobol sequence, respectively, gives even better results. However, the best results in terms of convergence rate of the error are obtained for the recursive method using Quasi-Monte Carlo sequences. To quantify this effect, we introduce the efficiency gain

$$gain_i = \frac{N^*_{\mathrm{MC}}(S)}{N^*_i(S)}$$

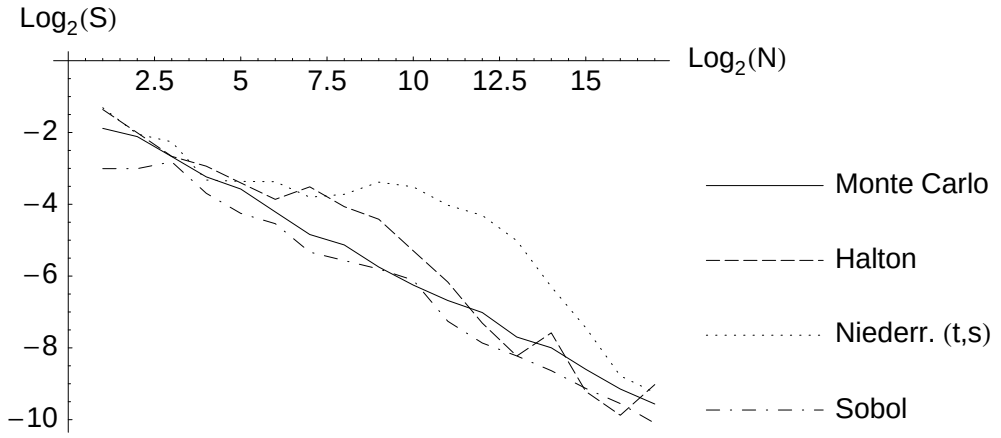

Figure 3.2: *Mean square error of the simulated survival probability in Model A*

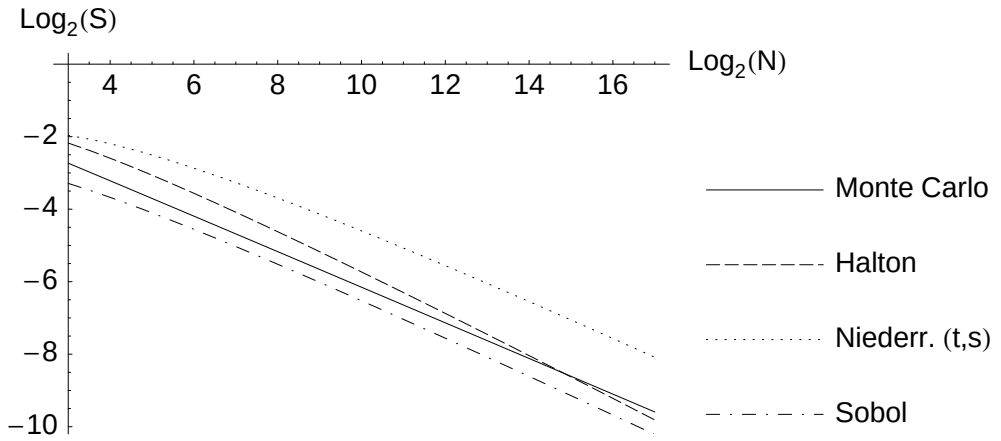

Figure 3.3: *Fits of the simulated survival probability in Model A*

$b\backslash x$	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	10.28										
0.1	10.28	10.75									
0.2	10.32	10.77	11.19								
0.3	10.40	10.86	11.27	11.63							
0.4	10.51	10.99	11.42	11.75	12.04						
0.5	10.62	11.12	11.56	11.94	12.24	12.44					
0.6	10.78	11.28	11.74	12.16	12.48	12.75	12.90				
0.7	10.94	11.48	11.97	12.37	12.74	13.01	13.25	13.35			
0.8	11.13	11.69	12.19	12.65	13.02	13.34	13.60	13.79	13.86		
0.9	11.33	11.91	12.44	12.91	13.31	13.67	13.95	14.22	14.37	14.43	
1.0	11.54	12.14	12.68	13.18	13.60	14.02	14.35	14.62	14.83	14.97	15.01

Table 3.2: *Exact values of the survival probability in % in Model A*

where $N^*_{\mathrm{MC}}(S)$ is the number of paths needed in the Monte Carlo simulation to reach a given error of S, and $N^*_i(S)$ is the corresponding number of paths (the number of summands in approximations (3.23) and (3.26), respectively) using an alternative method. Figure 3.5 shows that except for the $(0, s)$-nets all methods are an improvement in efficiency compared to Monte Carlo simulation, and the gain increases with smaller errors.

3.4.2 Expected value of the dividend payments

The exact values of $W(u, b)$ in Models A and B are given in Tables 3.5 and 3.6, respectively. The numerical results given in Table 3.7 and Figures 3.6 and 3.7 show that the performance of the various solution techniques is similar to the case of survival probabilities. For a moderate choice of N ($N \leq 2^{10}$) the Monte Carlo methods have a smaller mean square error than the QMC simulation techniques; for larger N, however, all Quasi-Monte Carlo methods outperform the Monte Carlo schemes, with the recursive algorithm giving better results than the simulation. This is in particular relevant for practical purposes, since the generation of these QMC-sequences can be done faster than the generation of pseudo-random numbers based on `ran1` or `ran2`.

For the dividend payments in Model B the superiority of the Quasi-Monte Carlo approach is even more pronounced (see Figures 3.8, 3.9 and Table 3.8).

Since for a fixed N the recursive numerical techniques need more calculation time than the simulation approach, it is instructive to investigate the accuracy of the numerical results with respect to calculation time. Figure 3.10 gives a log-log-plot of the mean-square error S as a function of calculation time t for the dividend payments in Model B. It turns out that the Quasi-Monte Carlo techniques clearly outperform the corresponding Monte Carlo techniques. For smaller values of t the Sobol sequence is particularly well-suited for our integrands, whereas for large t the use of the Halton sequence seems preferable.

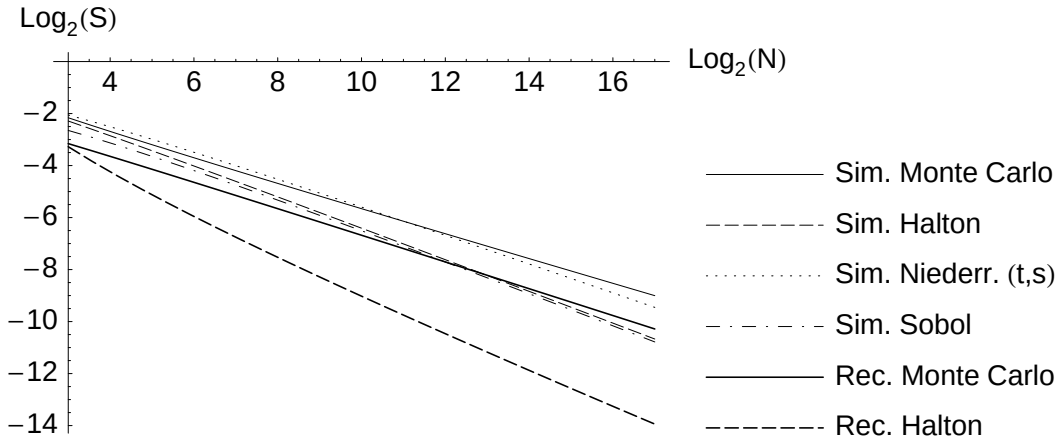

Figure 3.4: *Mean square error of the survival probability in Model B*

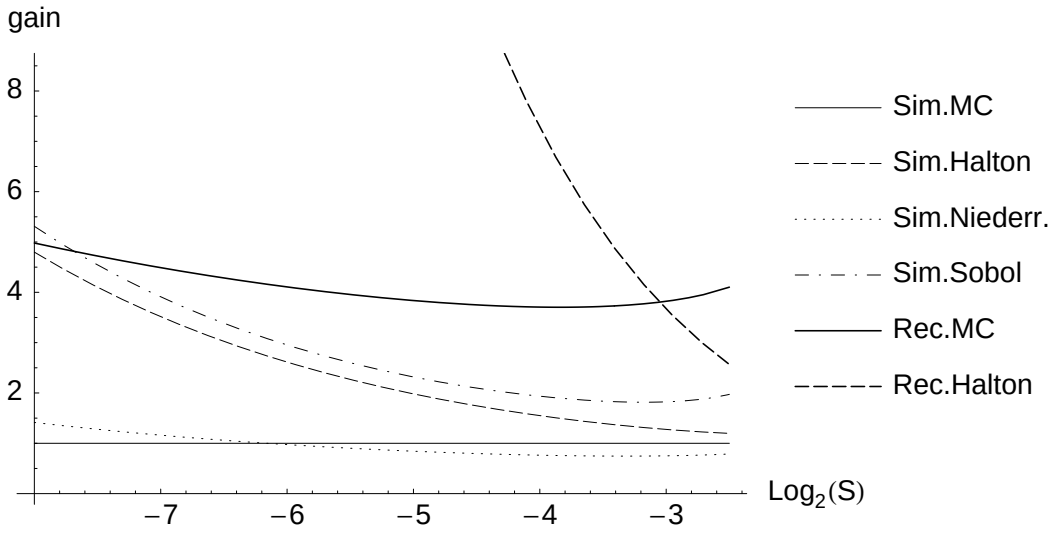

Figure 3.5: *Gain for the survival probability in Model B*

$b\backslash x$	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	23.28										
0.1	23.31	24.32									
0.2	23.40	24.44	25.34								
0.3	23.58	24.62	25.56	26.33							
0.4	23.80	24.87	25.87	26.64	27.27						
0.5	24.09	25.21	26.20	27.07	27.75	28.20					
0.6	24.42	25.56	26.64	27.54	28.29	28.86	29.22				
0.7	24.80	26.02	27.10	28.05	28.90	29.54	30.01	30.28			
0.8	25.22	26.50	27.64	28.65	29.50	30.26	30.84	31.27	31.45		
0.9	25.68	27.00	28.17	29.24	30.20	31.00	31.69	32.20	32.56	32.71	
1.0	26.17	27.50	28.75	29.90	30.88	31.76	32.52	33.14	33.61	33.93	34.07

Table 3.4: *Exact values of the survival probability in % in Model B*

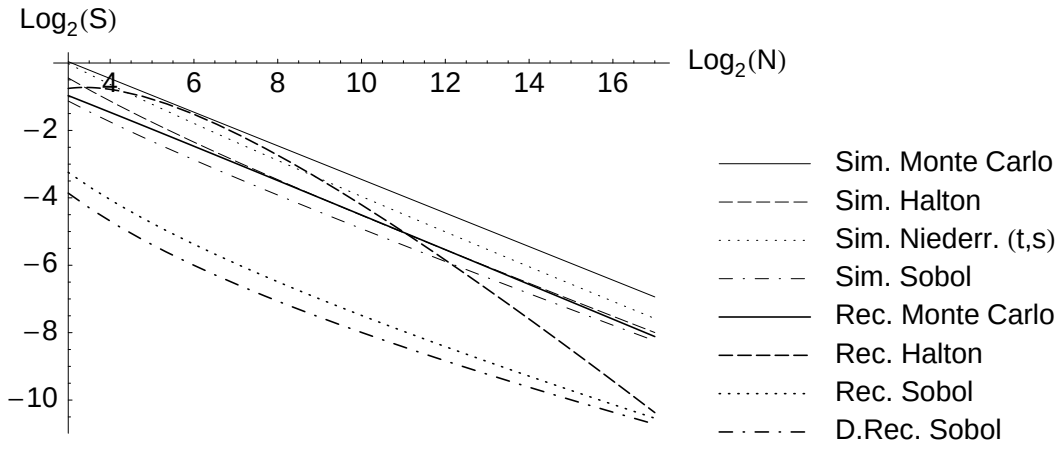

Figure 3.6: *Mean square error of the expected dividend payments in Model A*

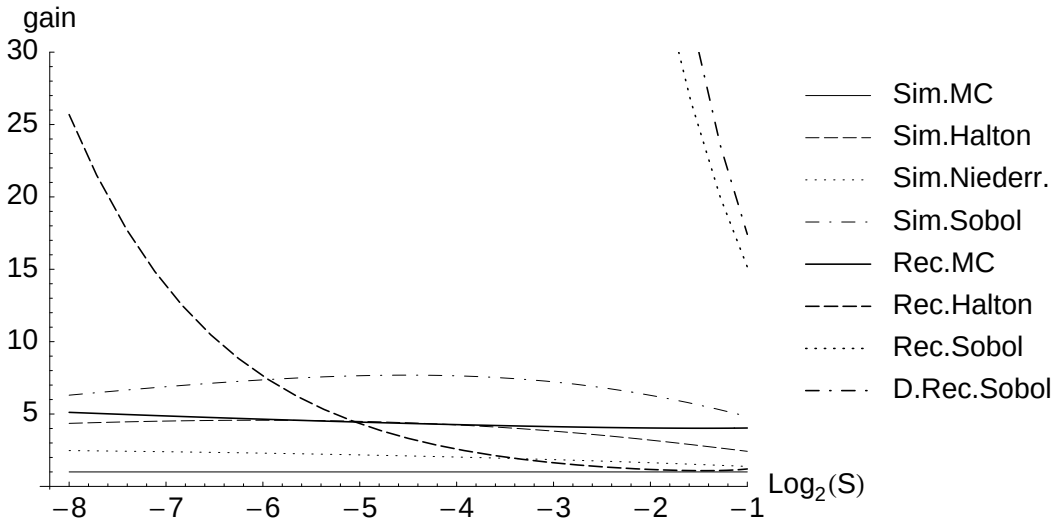

Figure 3.7: *Gain for the expected dividend payments in Model A*

$b\backslash x$	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	1.482										
0.1	1.482	1.592									
0.2	1.481	1.591	1.701								
0.3	1.480	1.590	1.699	1.808							
0.4	1.478	1.588	1.700	1.806	1.913						
0.5	1.476	1.586	1.696	1.805	1.912	2.014					
0.6	1.474	1.583	1.694	1.802	1.908	2.014	2.117				
0.7	1.469	1.582	1.690	1.797	1.904	2.008	2.114	2.215			
0.8	1.466	1.578	1.685	1.793	1.900	2.006	2.110	2.214	2.315		
0.9	1.462	1.572	1.680	1.788	1.894	2.001	2.104	2.208	2.311	2.412	
1.0	1.456	1.565	1.675	1.782	1.886	1.994	2.098	2.201	2.304	2.407	2.506

Table 3.5: *Exact values of the expected dividend payments in Model A*

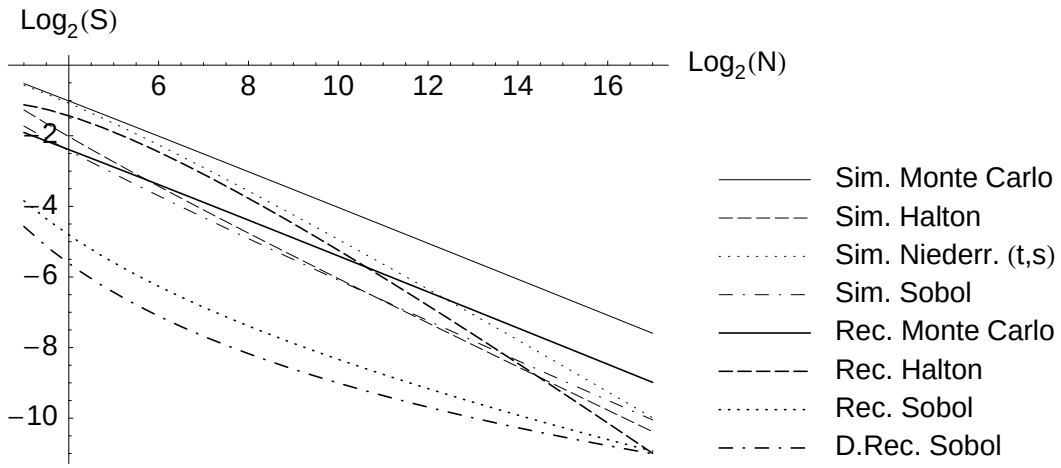

Figure 3.8: *Mean square error of the expected dividend payments in Model B*

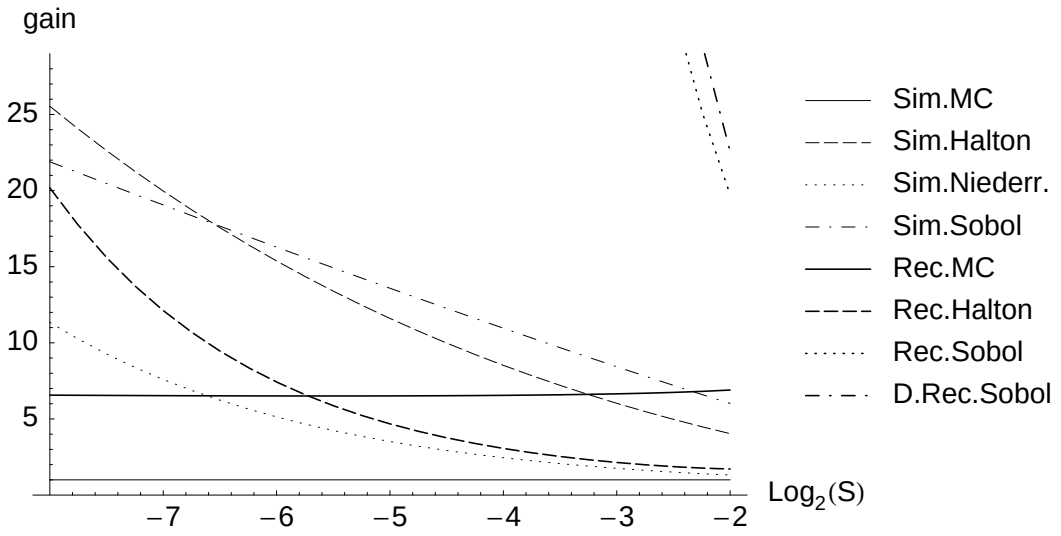

Figure 3.9: *Gain for the expected dividend payments in Model B*

$b\backslash x$	0.0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
0.0	1.045										
0.1	1.045	1.136									
0.2	1.041	1.132	1.225								
0.3	1.036	1.126	1.218	1.312							
0.4	1.028	1.118	1.211	1.302	1.397						
0.5	1.019	1.108	1.198	1.291	1.384	1.479					
0.6	1.007	1.095	1.185	1.276	1.368	1.463	1.559				
0.7	0.993	1.081	1.169	1.258	1.350	1.442	1.536	1.634			
0.8	0.977	1.064	1.151	1.239	1.328	1.420	1.513	1.608	1.706		
0.9	0.960	1.045	1.130	1.217	1.306	1.395	1.486	1.579	1.674	1.773	
1.0	0.940	1.023	1.108	1.193	1.278	1.367	1.457	1.548	1.641	1.737	1.836

Table 3.6: *Exact values of the expected dividend payments in Model B*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Simulation	S	0.007141	0.005095	0.006126	0.004136
	$\|\Delta\|_\infty$	0.018727	0.010935	0.009418	0.006817
		(163.16 s)	(149.58 s)	(281.61 s)	(150.09 s)
Recursive	S	0.004046	0.000755	0.001083	0.000755
	$\|\Delta\|_\infty$	0.012431	0.001758	0.002598	0.001786
		(507.3 s)	(494.72 s)	(494.62 s)	(495.04 s)
Double recursive	S	0.004309	0.00078	0.000871	0.001054
	$\|\Delta\|_\infty$	0.008598	0.001811	0.002389	0.002432
		(3914.71 s)	(1761.22 s)	(1761.44 s)	(3910.15 s)

Table 3.7: *Errors for the expected dividend payments in Model A*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Simulation	S	0.004778	0.000855	0.001262	0.000958
	$\|\Delta\|_\infty$	0.010684	0.002495	0.002914	0.002464
		(99.71 s)	(86.92 s)	(87.21 s)	(86.91 s)
Recursive	S	0.002134	0.000607	0.000479	0.000497
	$\|\Delta\|_\infty$	0.005386	0.001762	0.001526	0.00161
		(149.74 s)	(136.98 s)	(136.8 s)	(136.7 s)
Double recursive	S	0.002207	0.000709	0.000636	0.000721
	$\|\Delta\|_\infty$	0.005466	0.001953	0.001843	0.002008
		(331.1 s)	(325.63 s)	(325.32 s)	(324.69 s)

Table 3.8: *Errors for the expected dividend payments in Model B*

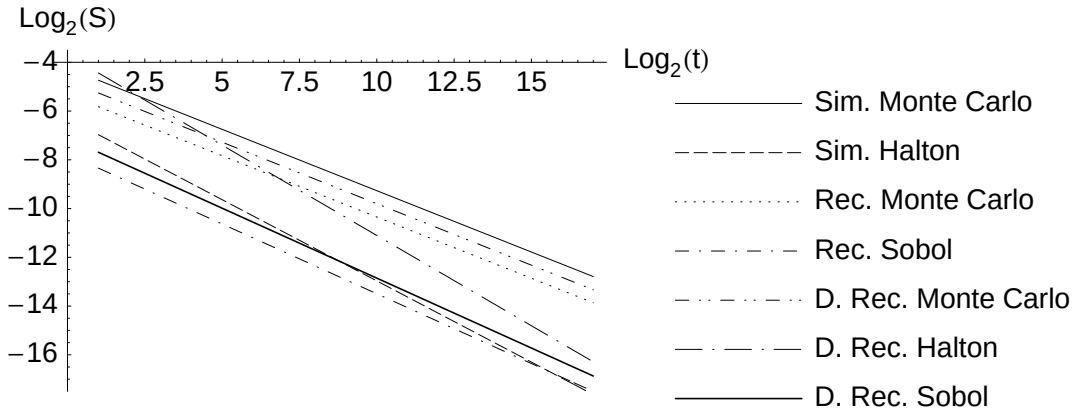

Figure 3.10: *Mean square error of the expected dividend payments in Model B, compared with respect to calculation time*

Chapter 4

Efficient simulation techniques for a generalized ruin model

Contents

In this chapter we consider a generalized version of the classical model for the collective surplus process of an insurance portfolio. In the presence of dividend payments according to a non-linear barrier strategy and interest on the free reserve we derive equations for the probability of ruin and the expected present value of dividend payments which give rise to several numerical number-theoretic solution techniques. For various claim size distributions and a parabolic barrier numerical tests and comparisons of these techniques are performed. In particular, the efficiency gain obtained by implementing low-discrepancy sequences instead of pseudo-random sequences is investigated.
This chapter is an extension to the model presented in the previous chapter, and is based on joint works [3, 4] with H. Albrecher and R. Tichy. The research on this topic was supported in part also by the OeNB Project Nr. 9002.

4.1 Introduction

Let $\{N(t) : t \in \mathbb{R}_+\}$ denote the random process that counts the claims of an insurance portfolio of a company up to time t and assume that $N(t)$ is a homogeneous Poisson process with

intensity λ. Let further $\{X_n : n \in \mathbb{N}\}$ be a sequence of independent identically distributed positive random variables with distribution function $F(y)$ representing the sizes of the successive claims and let $\mu = E(X_i) < \infty$. In a time interval $[t, t+dt]$ the company receives the premium $c\,dt$, where $c > \lambda \int_0^\infty y\,dF(y)$. In addition to the premium income, we assume that the company also receives interest on its reserves with a constant interest force i (for $i = 0$ we have the classical ruin model; for a general background in ruin theory see for instance GERBER [29], THORIN [67] or more recently DEVYLDER [71] and ASMUSSEN [8]). Let $T_n\,(n \in \mathbb{N})$ denote the moment of occurrence of the nth claim. If we introduce the purely discontinuous measure $X_{N_t}\,\mathrm{d}N_t$ which puts a weight equal to X_{N_t} at times $T_n\,(n \in \mathbb{N})$, then the value of the reserve at time t, denoted by R_t, satisfies

$$\mathrm{d}R_t = c\,\mathrm{d}t + R_t \cdot i\,\mathrm{d}t - X_{N_t}\,\mathrm{d}N_t$$

(see for example DELBAEN AND HAEZENDONCK [20]).
We now extend this model by introducing a time-dependent dividend barrier b_t, such that whenever the value of the reserve R_t reaches b_t, dividends are paid out to the shareholders with intensity $(c + R_t \cdot i) - db_t$ and the surplus remains on the barrier, until the next claim occurs. This means that the risk process develops according to

$$dR_t = (c + i\,R_t)\,dt - X_{N_t}\,\mathrm{d}N_t \qquad \text{if } R_t < b_t \tag{4.1}$$

$$dR_t = db_t - X_{N_t}\,\mathrm{d}N_t \qquad \text{if } R_t = b_t. \tag{4.2}$$

Together with the initial capital $R_0 = u,\ 0 \le u < b_0 < \infty$, this determines the risk process $\{R_t,\, t \ge 0\}$ (cf. Figure 4.1).

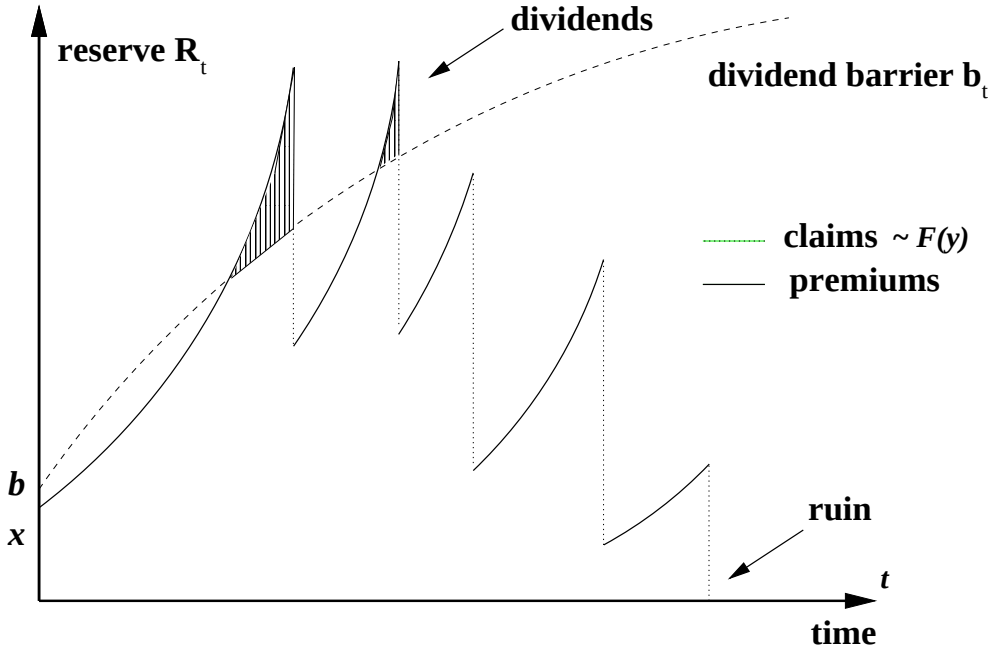

Figure 4.1: *A sample path of* R_t

The following quantities are of particular interest in this context: The survival probability is defined as the probability that the reserve of the portfolio never becomes negative, i.e.

$$\phi(u, b) = Pr\{R_t \ge 0\ \forall\ t \ge 0 \,|\, R_0 = u\},$$

where $u \geq 0$ denotes the initial reserve of the portfolio. Correspondingly the probability of ruin is defined by $\psi(u,b) = 1 - \phi(u,b)$. Another important quantity is the expected sum of discounted dividend payments $W(u,b)$, i.e. the expected present value of all dividends paid until ruin occurs.

Dividend barrier models have a long history in risk theory (see e.g. [15], [29]). GERBER [28] showed that barrier dividends constitute a complete family of Pareto-optimal dividends. In the case of a horizontal dividend barrier $b_t \equiv b_c = \text{const.}$, we have $\phi(u,b) = 0 \; \forall \; 0 \leq u \leq b$. Including a constant interest force on the reserve in the model, PAULSEN AND GJESSING [58] calculated the optimal value of b_c that maximizes the expected value of the discounted dividend payments in this situation. For linear dividend barriers $b_t = b + at$ GERBER [27] derived an upper bound for the probability of ruin by martingale methods and in [30] he obtained exact solutions for the probability of ruin and the expected sum of discounted dividend payments $W(u,b)$ for exponentially distributed claim amounts; this result was generalized by SIEGL AND TICHY [62] to arbitrary Erlang claim amount distributions, see also ALBRECHER AND TICHY [5].

In [2] non-linear dividend barrier models of the type

$$b_t = \left(b^m + \frac{t}{\alpha} \right)^{1/m} \qquad (\alpha, b > 0, m \geq 1). \tag{4.3}$$

were introduced and integro-differential equations for $\phi(u,b)$ and $W(u,b)$ were derived. The existence and uniqueness of the corresponding solutions was discussed and techniques for numerical solutions were developed and tested for the case of an exponential claim size distribution. In [4] this approach was applied to more general claim size distributions and at the same time continuously compounded interest on the free reserve was included in the model.

The article [3] (and thus also this chapter, which is based on the article) is a more general and extended version of [4]. In Section 4.2 we identify $\phi(u,b)$ and $W(u,b)$ as solutions of boundary value problems for integro-differential equations and also as fixed points of contracting integral operators which gives rise to the development of efficient number-theoretic simulation techniques based on Monte Carlo and Quasi-Monte Carlo methods. These are discussed in Section 4.3. In Section 4.4 we give detailed numerical results for a parabolic dividend barrier. The various simulation techniques are compared on a quantitative and qualitative basis. Finally the efficiency gain obtained by implementing various low-discrepancy sequences is investigated and the sensitivity of the simulation results with respect to the model assumptions is discussed.

4.2 Integro-differential equations and integral operators

In the sequel we will consider dividend barriers of type (4.3). Note that $m = 1$ corresponds to the linear barrier case.
The probability of survival $\phi(u,b)$ for the surplus process given by (4.1) and (4.2) can then be expressed as the solution of a boundary value problem in the following way: Conditioning

on the occurrence of the first claim, we get for $u < b$

$$\phi(u,b) = (1-\lambda dt)\phi\left(u + (c + i\,u)\,dt, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) + \\ + \lambda\,dt \int_0^{u+(c+i\,u)\,dt} \phi\left(u + (c+i\,u)\,dt - z, \left(b^m + \frac{dt}{\alpha}\right)^{1/m}\right) dF(z). \quad (4.4)$$

Taylor series expansion of (4.4) and division by dt shows that ϕ satisfies the equation

$$(c + i\,u)\,\frac{\partial \phi}{\partial u} + \frac{1}{\alpha\,m\,b^{m-1}}\,\frac{\partial \phi}{\partial b} - \lambda\,\phi + \lambda \int_0^u \phi(u-z,b)dF(z) = 0, \quad (4.5)$$

which, for reasons of continuity, is valid for $0 \le u \le b$. For $u = b$ the same arguments can be used to show that equation (4.5), with $c + i\,u$ replaced by $\frac{1}{\alpha m\,b^{m-1}}$, also holds. Thus we obtain the boundary condition

$$\frac{\partial \phi}{\partial u}\Big|_{u=b} = 0. \quad (4.6)$$

A further natural requirement is

$$\lim_{b\to\infty} \phi(u,b) = \phi(u), \quad (4.7)$$

where $\phi(u)$ is the probability of survival in absence of the barrier.

Let furthermore $W(u,b)$ denote the expected present value of the future dividend payments, which are discounted according to the risk-less interest rate i, and stop when ruin occurs. Then, in a similar way to (4.4), one can derive the integro-differential equation

$$(c + i\,u)\,\frac{\partial W}{\partial u} + \frac{1}{\alpha\,m\,b^{m-1}}\,\frac{\partial W}{\partial b} - (i+\lambda)\,W + \lambda \int_0^u W(u-z,b)dF(z) = 0, \quad (4.8)$$

with boundary condition

$$\frac{\partial W}{\partial u}\Big|_{u=b} = 1. \quad (4.9)$$

Remark 1: In principle, one could follow this approach for any dividend barrier function $b_t = f(b,t)$ that is monotone increasing in t and satisfies

$$f(b,t) = f\Big(f(b,t_1), t - t_1\Big) \quad \forall\, b > 0 \text{ and } \forall\, t > t_1 > 0. \quad (4.10)$$

The functional equation (4.10) is the well-known translation equation and for functions $f(b,t)$ which are monotone increasing in b and t and continuous in b, the general solution of (4.10) is given by

$$f(b,t) = h\Big(h^{-1}(b) + t\Big),$$

where $h(t) = f(b_0,t)$ is some given initial function (see e.g. Aczél [1]). From $h(t) = (b_0^m + t/\alpha)^{\frac{1}{m}}$ we obtain (4.3) as a special case. Other solutions of (4.10) include for instance $f(b,t) = b + at$ (linear barrier) or $f(b,t) = (\sqrt{b} + t)^2$ (quadratic barrier).

Remark 2: For the special case of exponentially distributed claim sizes it follows from (4.5) and (4.8) that $\phi(u,b)$ and $W(u,b)$ can be expressed as the solutions of boundary value problems for second-order partial differential equations of hyperbolic type. However, due to

the structure of the boundary conditions this does not lead to a simplification of the problem (cf. [2]).

Following a procedure developed by GERBER [30] for the case of linear barriers, we first show that the boundary value problem (4.8) together with (4.9) has a unique bounded solution. For that purpose, we define an operator A by

$$\begin{aligned} Ag(u,b) = &\int_0^{t^*} \lambda e^{-(\lambda+i)t} \int_0^{(c'+u)e^{it}-c'} g\left((c'+u)e^{it}-c'-z, \left(b^m+\frac{t}{\alpha}\right)^{1/m}\right) dF(z)dt + \\ &+ \int_{t^*}^{\infty} \lambda e^{-(\lambda+i)t} \int_0^{\left(b^m+\frac{t}{\alpha}\right)^{1/m}} g\left(\left(b^m+\frac{t}{\alpha}\right)^{1/m}-z, \left(b^m+\frac{t}{\alpha}\right)^{1/m}\right) dF(z)dt + \\ &+ \int_{t^*}^{\infty} \lambda e^{-\lambda t} \int_{t^*}^{t} e^{-is}\left((c+i\,u)e^{is} - \frac{1}{m\alpha\left(b^m+\frac{s}{\alpha}\right)^{1-1/m}}\right) ds\; dt, \end{aligned} \tag{4.11}$$

with $c' = c/i$. Here t^* is the positive solution of $(c'+u)e^{it}-c' = \left(b^m+\frac{t}{\alpha}\right)^{1/m}$ (this number is unique for all $m \geq 1$, since $u < b$). The solution $W(u,b)$ of (4.8) with its initial condition (4.9) is a fixed point of the integral operator A. For any two bounded functions g_1, g_2

$$|Ag_1(u,b) - Ag_2(u,b)| \leq \|g_1 - g_2\| \int_0^{\infty} \lambda e^{-(\lambda+i)t} dt \leq \frac{\lambda}{\lambda+i} \|g_1 - g_2\| \tag{4.12}$$

for arbitrary $0 \leq u \leq b < \infty$, where $\|\cdot\|$ is the supremum norm on $0 \leq u \leq b < \infty$, and thus it follows that A is a contraction and the fixed point is unique by Banach's theorem.
The integral operator (4.11) does not only prove the existence and uniqueness of a solution of (4.8) and (4.9), but also allows for the development of numerical solution algorithms taking advantage of the contraction map (see Section 4.3).

Unfortunately, the same approach does not allow to show the contraction property of the corresponding integral operator for the probability of survival $\phi(u,b)$. We will see in Section 4.3, that by stochastic simulation of the risk reserve process one still can obtain numerical solutions for $\phi(u,b)$ in a satisfying way. But especially for efficiency comparison purposes of the various numerical solution methods it would be nice to have such a contracting integral operator available. Thus we also consider a slight modification of our risk model in that we introduce an absorbing upper barrier $b_{max} \equiv const$, i.e. if the surplus process $R_t \geq b_{max}$ for some $t > 0$, it is absorbed, the dividend payments stop and the company is considered to have survived. From an economic point of view this can be interpreted that the company will then decide to pursue other forms of investment strategies. Mathematically, this model has some nice features (e.g. the process stops in finite time with probability 1). In the sequel we will refer to this modified version as Model B. The boundary value problem for the probability of survival can then be formulated by (4.5), (4.6) and

$$\phi(u, b_{max}) = \frac{\phi(u)}{\phi(b_{max})}, \tag{4.13}$$

where $0 \leq u \leq b \leq b_{max}$ and as before $\phi(u)$ is the probability of survival in absence of the barrier.

In Model B we can now proceed to obtain a contraction map for the probability of survival as its fixed point: Like in equation (4.11), let t^* be the time when the surplus would reach the dividend barrier given that no claim occurs. Let furthermore $t^{**} = \alpha(b_{max}^m - b^m)$ be the time when the dividend barrier reaches the absorbing barrier, and

$$\tilde{t} = \frac{1}{i} \log \left(\frac{c + i\, b_{max}}{c + i\, u} \right)$$

the time when the surplus would reach the absorbing barrier in the absence of a dividend barrier and of claims. As the dividend barrier is an increasing function on $\mathbb{R}^+$, t^{**} is uniquely determined, just as is $\tilde{t}$. Combining the two possible scenarios $0 \le t^{**} \le \tilde{t} \le t^*$ and $0 \le t^* \le \tilde{t} \le t^{**}$ (depending on the values of u and b), we define the operator A as

$$A\phi(u,b) = \int_0^T \lambda e^{-\lambda t} \int_0^{z_{max}(u,b,t)} \phi\left(z_{max}(u,b,t) - z, \left(b^m + \frac{t}{\alpha} \right)^{\frac{1}{m}} \right) dF(z)\, dt + e^{-\lambda T}, \tag{4.14}$$

where $T = \max\left(\tilde{t}, t^{**}\right)$ is the (finite) time when the surplus process reaches the absorbing upper barrier b_{max} in the absence of claims, and

$$z_{max}(u,b,t) = \min\left((c'+u)e^{it} - c', \left(b^m + \frac{t}{\alpha} \right)^{\frac{1}{m}} \right). \tag{4.15}$$

Let ϕ_1 and ϕ_2 now be two bounded functions on $0 \le u \le b \le b_{max}$, then

$$|A\phi_1(u,b) - A\phi_2(u,b)| \le \|\phi_1 - \phi_2\| \int_0^T \lambda e^{-\lambda t} dt = \|\phi_1 - \phi_2\| \left(1 - e^{-\lambda T}\right).$$

Since $T = T(u,b) < M < \infty$, this operator is a contraction, and Banach's fixed point theorem establishes the existence and uniqueness of the solution $\phi(u,b)$ in Model B.

Correspondingly, the contraction map for the expected sum of dividend payments in Model B is given by

$$\begin{aligned} Ag(u,b) = &\int_0^{t^*} \lambda e^{-(\lambda+i)t} \int_0^{(c'+u)e^{it}-c'} g\left((c'+u)e^{it} - c' - z, \left(b^m + \frac{t}{\alpha} \right)^{1/m} \right) dF(z) dt + \\ &+ \int_{t^*}^{t^{**}} \lambda e^{-(\lambda+i)t} \int_0^{\left(b^m + \frac{t}{\alpha}\right)^{1/m}} g\left(\left(b^m + \frac{t}{\alpha} \right)^{1/m} - z, \left(b^m + \frac{t}{\alpha} \right)^{1/m} \right) dF(z) dt + \\ &+ \int_{t^*}^{t^{**}} e^{-(\lambda+i)t} \left((c + i\,u)e^{it} - \frac{1}{m\alpha \left(b^m + \frac{t}{\alpha}\right)^{1-1/m}} \right) dt, \end{aligned} \tag{4.16}$$

if $t^{**} > t^*$ and $Ag(u,b) = 0$ otherwise, because then the surplus reaches the absorbing barrier before the dividend barrier. The last term in (4.16) represents the dividends that are paid out until t^{**} and is a simplification of the original expression

$$\begin{aligned} &\int_{t^*}^{t^{**}} \lambda e^{-\lambda t} \int_{t^*}^{t} e^{-is} \left((c + i\,u)e^{it} - \frac{1}{m\alpha \left(b^m + \frac{s}{\alpha}\right)^{1-1/m}} \right) ds\, dt + \\ &\qquad \int_{t^{**}}^{\infty} \lambda e^{-\lambda t} \int_{t^*}^{t^{**}} e^{-is} \left((c + i\,u)e^{it} - \frac{1}{m\alpha \left(b^m + \frac{s}{\alpha}\right)^{1-1/m}} \right) ds\, dt. \end{aligned}$$

From (4.16) it follows that

$$\|Ag_1(u,b) - Ag_2(u,b)\| \leq \frac{\lambda}{\lambda+i}\left(1 - e^{-(\lambda+i)t^{**}}\right)\|g_1 - g_2\| \ ,$$

for any two bounded functions g_1, g_2 and we again have a contraction in the Banach space of bounded functions equipped with the supremum norm, which implies the existence and uniqueness of the solution.

Since these boundary value problems can not be solved analytically, there is a need for effective algorithms to obtain numerical solutions. Here we focus on the development of number-theoretic solution methods based on the corresponding integral operators and on stochastic simulation, respectively.

4.3 Numerical solution techniques

The following three algorithms are presented in terms of operator (4.11). However, the adaptation to the other integral operators introduced above is straight-forward.

4.3.1 Double-recursive algorithm

Following a technique that was already used in TICHY [68], the fixed point of (4.11) can be approximated by applying the contracting integral operator A k times to a starting function $h(u,b)$ which we choose to be the inhomogeneous term in the corresponding integral operator (where k is chosen according to the desired accuracy of the solution):

$$g^{(k)}(u,b) = A^k g^{(0)}(u,b),$$
$$g^{(0)}(u,b) = h(u,b) := \int_{t^*}^{\infty} \lambda e^{-\lambda t} \int_{t^*}^{t} e^{-is}\left((c + i\,u)e^{is} - \frac{1}{m\alpha\left(b^m + \frac{s}{\alpha}\right)^{1-1/m}}\right) ds\, dt.$$

This leads to a $2k$-dimensional integral for $g^{(k)}(u,b)$, which is calculated numerically using Monte Carlo and Quasi-Monte Carlo methods. For that purpose we transform the integration domain of operator (4.11) into the unit cube:

$$Ag(u,b) = h(u,b) + \frac{\lambda}{\lambda+i}\cdot$$
$$\cdot\left[\left(1 - e^{-(\lambda+i)t^*}\right)\int_0^1\int_0^1 g\left((c'+u)e^{it_1} - c' - z_1, \left(b^m + \frac{t_1}{\alpha}\right)^{\frac{1}{m}}\right)\cdot F\left((c'+u)e^{it_1} - c'\right) dv_1 dw_1\right.$$
$$\left. + e^{-(\lambda+i)t^*}\int_0^1\int_0^1 g\left(\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}} - z_2, \left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}}\right)\cdot F\left(\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}}\right) dv_2 dw_2\right]$$

with

$$t_1 = -\frac{\log\left(1 - w_1\left(1 - e^{-(\lambda+i)t^*}\right)\right)}{\lambda+i} \qquad z_1 = F^{-1}\left(v_1 \cdot F\left((c'+u)e^{it} - c'\right)\right) \tag{4.17}$$

$$t_2 = t^* - \frac{\log(1-w_2)}{(\lambda+i)} \qquad z_2 = F^{-1}\left(v_2 \cdot F\left(\left(b^m + \frac{t_2}{\alpha}\right)^{\frac{1}{m}}\right)\right). \tag{4.18}$$

The Monte Carlo-estimator of $W(u,b)$ for given values of u and b is

$$W(u,b) \approx \frac{1}{N}\sum_{n=1}^{N} g_n^{(k)}(u,b)\ , \tag{4.19}$$

where the $g_n^{(k)}(u,b)$ are calculated recursively for each n by

$$g_n^{(0)}(u,b) = h(u,b)$$

and

$$\begin{aligned} g_n^{(i)}(u,b) &= h(u,b) + \frac{\lambda}{\lambda+i}\cdot \\ \cdot \Bigg\{ & F\left((c'+u)e^{it_{1,n}^i} - c'\right)\left(1-e^{-(\lambda+i)t^*}\right) g_n^{(i-1)}\left((c'+u)e^{it_{1,n}^i} - c' - z_{1,n}^i, \left(b^m + \frac{t_{1,n}^i}{\alpha}\right)^{\frac{1}{m}}\right) + \\ & + F\left(\left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}}\right) e^{-(\lambda+i)t^*} g_n^{(i-1)}\left(\left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}} - z_{2,n}^i, \left(b^m + \frac{t_{2,n}^i}{\alpha}\right)^{\frac{1}{m}}\right)\Bigg\}. \end{aligned}$$

Here $t_{j,n}^i$ and $z_{j,n}^i$ $(j=1,2)$ are determined according to (4.17) and (4.18) for (quasi-)random deviates v_j, w_j of the uniform distribution in the unit interval $(1 \le i \le k)$.

Since in every recursion step the function g is called twice, the number of evaluations of g doubles in every recursion step. Thus, in order to keep the computations tractable, in what we will call the double-recursive algorithm in the sequel, the double recursion is only used for the first three recursive steps and for the remaining recursion steps the recursive algorithm described in Section 4.3.2 is applied.

4.3.2 Recursive agorithm

Instead of calculating the first two integrals occurring in operator (4.11) separately, one can combine them to one integral. A suitable change of variables then leads to

$$\begin{aligned} Ag(u,b) &= h(u,b) + \\ &\int_0^1\int_0^1 \frac{\lambda}{\lambda+i} F\left(z_{max}(u,b,t)\right) g\left(z_{max}(u,b,t) - z, \left(b^m + \frac{t}{\alpha}\right)^{\frac{1}{m}}\right) dvdw \end{aligned} \tag{4.20}$$

where t and z are given by

$$\begin{aligned} t &= -\frac{\log(1-w)}{(\lambda+i)} \\ z &= -F^{-1}\left(v \cdot F(z_{max}(u,b,t))\right) \end{aligned} \tag{4.21}$$

and $z_{max}(u,b,t)$ is determined by (4.15). The integral operator (4.20) is now applied k times onto $g^{(0)}$, and the resulting multidimensional integral $g^{(k)}(u,b)$ is again approximated by

$$g^{(k)}(u,b) \approx \frac{1}{N}\sum_{n=1}^{N} g_n^{(k)}(u,b), \tag{4.22}$$

where each $g_n^{(k)}(u,b)$ $(n = 1,\ldots,N)$ is based on a pseudo-random (or quasi-random, resp.) point $\mathbf{x}_n \in [0,1]^{2k}$ and calculated by the recursion

$$g_k^{(0)}(u,b) = h(u,b),$$
$$g_n^{(i)}(u,b) = \frac{\lambda}{\lambda+i} F\left(z_{max}(u,b,t_n^i)\right) g_n^{(i-1)}\left(z_{max}(u,b,t_n^i) - z_n^i, \left(b^m + \frac{t_n^i}{\alpha}\right)^{\frac{1}{m}}\right) + h(u,b),$$

with $1 \le i \le k$. t_n^i and z_n^i are given by (4.21) with v and w being the value of the $(2i)$-th and $(2i+1)$-th, component of $\mathbf{x}_n$, respectively. Note that for this algorithm, the number of integration points needed for a given recursion depth is one eighth of the corresponding number required for the double-recursive case.

4.3.3 Iterative algorithm

Another solution technique based on the integral operator (4.20) is to discretize the domain of u and b by a grid (u_j, b_k), $0 \le j \le j_{max}$, $0 \le k \le k_{max}$. After assigning a suitable initial value to each discretization point (u_j, b_k), the operator is applied sequentially to each point (u_j, b_k) of the grid. The resulting approximative solution $\hat{g}_{j,k}^{(i)}$ at point (u_j, b_k) and iteration depth i is calculated from the values of $\hat{g}^{(i-1)}$ at depth $(i-1)$ by a two-dimensional integral, which is evaluated by Monte Carlo and Quasi-Monte Carlo methods. Since $\hat{g}^{(i-1)}$ is only defined for the discretization points (u_j, b_k), the function $g(z_{max}(u,b,t) - z, (b^m + t/\alpha)^{1/m})$ in operator (4.20) is replaced by an interpolation function $I(\hat{g}^{(i-1)}, z_{max}(u,b,t) - z, (b^m + t/\alpha)^{1/m})$ defined in (4.24). Thus we have

$$\begin{aligned}
\hat{g}_{j,k}^{(0)} =& h(u_j, b_k) \\
\hat{g}_{j,k}^{(i)} =& h(u_j, b_k) + \frac{\lambda}{\lambda+i} \cdot \\
& \frac{1}{N}\sum_{n=1}^{N} F\left(z_{max}(u_j, b_k, t_n^i)\right) I\left(\hat{g}^{(i-1)}, z_{max}(u_j, b_k, t_n^i) - z_n^i, \left(b_k^m + \frac{t_n^i}{\alpha}\right)^{\frac{1}{m}}\right)
\end{aligned} \tag{4.23}$$

for $1 \le i \le i_{max}$. The variables (t_n^i, z_n^i) are obtained from the elements of a 2-dimensional sequence according to (4.21), and the iteration depth i_{max} does not have to be fixed in advance but may be chosen according to the desired accuracy of the approximation.

In our calculations we use a linear interpolation function for approximating the value of $g^{(i)}$ at point (u,b) with $u_j \le u \le u_{j+1}$ and $b_k \le b \le b_{k+1}$:

$$\begin{aligned}
I(\hat{g}^{(i)}, u, b) = & \left(1 - \frac{u-u_j}{u_{j+1}-u_j}\right)\left(1 - \frac{b-b_k}{b_{k+1}-b_k}\right)\hat{g}_{j,k}^{(i)} + \frac{u-u_j}{u_{j+1}-u_j}\frac{b-b_k}{b_{k+1}-b_k}\hat{g}_{j+1,k+1}^{(i)} + \\
& \left(1 - \frac{u-u_j}{u_{j+1}-u_j}\right)\frac{b-b_k}{b_{k+1}-b_k}\hat{g}_{j,k+1}^{(i)} + \frac{u-u_j}{u_{j+1}-u_j}\left(1 - \frac{b-b_k}{b_{k+1}-b_k}\right)\hat{g}_{j+1,k}^{(i)} .
\end{aligned} \tag{4.24}$$

If $u_{j+1} > b_k$ (i.e. the left upper corner of the surrounding rectangle lies above $u = b$ and thus outside the valid domain of g), then $g_{j+1,k}^{(i)}$ needs to be replaced by

$$g_{j,k}^{(i)} + g_{j+1,k+1}^{(i)} - g_{j,k+1}^{(i)}$$

in (4.24) meaning that the plane defined by the remaining three corners is then used for the linear interpolation. If (u,b) lies outside the area that is covered by the grid, the nearest rectangle $((u_j, b_k), (u_{j+1}, b_{k+1}))$ covered by the grid is used for extrapolation by (4.24).

4.3.4 Simulation

Another way to obtain numerical solutions is stochastic simulation of the surplus process. For that purpose we sample N paths of the risk reserve process in the following way: Starting with $t_0 = 0$, $b_0 = b$ and $x_0 = u$, where u is the initial reserve of the insurance portfolio, we successively generate exponentially distributed random variables $\tilde{t}_j$ with parameter λ for the time until the next claim occurs and set $t_{j+1} := t_j + \tilde{t}_j \, (j \in \mathbb{N})$. The claim amount z_j is then sampled from the corresponding claim size distribution by the inversion method, and the reserve after the claim is $x_{j+1} = \min\{(c' + x_j)e^{i\tilde{t}_j} - c', (b_j^m + \tilde{t}_j/\alpha)^{1/m}\} - z_j$. Due to the structure of the dividend barrier, we can reset the origin to t_{j+1} in every step, if we also set $b_{j+1} = \left(b_j^m + \frac{\tilde{t}_j}{\alpha}\right)^{1/m}$. We then have to discount the dividend payments between the j-th and $(j+1)$-th claims by the factor e^{-it_j}.

A simulation estimate for the survival probability $\phi(u, b)$ can now be obtained by

$$\phi(u, b) \approx \frac{m}{N},$$

where m is the number of paths for which ruin does not occur (i.e. $x_j > 0 \; \forall \, j$). We consider a path as having survived, if for some j the condition $x_j > x_{max}$ is fulfilled, where x_{max} is a sufficiently large threshold. This can be viewed as an absorbing horizontal barrier at x_{max}, and so the process stops with probability 1. Using this stopping criterion, we overestimate the actual probability of survival $\phi(u, b)$; for sufficiently large x_{max}, however, this effect is negligible.

For the simulation of the expected value of the dividend payments, we proceed as described above and whenever the process reaches the dividend barrier, i.e. $(c' + x_j)e^{i\tilde{t}_j} - c' > (b_j^m + \tilde{t}_j/\alpha)^{\frac{1}{m}}$, we need to calculate the amount of dividends that are paid until the next claim j occurs:

$$v_j = v_{j-1} + e^{-it_j} \int_{t^*}^{\tilde{t}_j} e^{-is} \left((c + ix_j)e^{is} - \frac{1}{m\alpha \left(b_j^m + \frac{s}{\alpha}\right)^{1-\frac{1}{m}}} \right) ds, \qquad j \geq 1$$

and $v_0 = 0$, where t^* is the positive solution of $(c' + x_j)e^{i\,t} - c' = \left(b_j^m + \frac{t}{\alpha}\right)^{1/m}$, i.e. the time when the process reaches the dividend barrier. The process is stopped, if ruin occurs (i.e. $x_j < 0$ for some j) or at some sufficiently large time t_{max}, after which the expected value of discounted dividends becomes negligible due to the discount factor e^{-it}. Let $v(k)$ now be the final value of v_j for path k. The expected value of the dividends is then approximated by

$$W(u, b) \approx \frac{1}{N} \sum_{k=1}^{N} v(k) \; .$$

4.4 Numerical results for the parabolic case

In this section we present numerical results for a parabolic dividend barrier of the form $b_t = \sqrt{b^2 + t/\alpha}$ and various claim amount distributions. Note that in this case, t^* is the

solution of the equation

$$(c' + u)e^{it^*} - c' = \left(b^2 + \frac{t^*}{\alpha}\right)^{1/2},$$

which needs to be calculated numerically, and the inhomogeneous term $h(u, b)$ in (4.11) can be calculated to

$$h(u,b) = e^{-\lambda t^*}\left(\frac{c+iu}{\lambda} - \frac{e^{-it^*}e^{z^2}}{2}\sqrt{\frac{\pi}{(\lambda+i)\alpha}}\, erfc(z)\right)$$

with $z = \sqrt{(\lambda + i)(\alpha b^2 + t^*)}$.

4.4.1 General remarks

Table 4.1 gives the choice of the parameter values and the densities of the claim size distributions used in our calculations. The corresponding distribution parameters are chosen such that the mean and variance of the distributions coincide and are equal to 1 and 0.5, respectively. Three of the four distributions are heavy-tailed.

	Parameter	Values
b	height of dividend barrier at time $t = 0$	0..[0.1]..1
u	initial capital	0..[0.1]..b
c	premium density	1.5
λ	intensity of claim number process	1
i	constant interest force	0.1
α	dividend barrier parameter	0.5
t_{max}	stopping criterion for simulations	100
b_{max}	absorbing upper barrier in Model B	4

Claim size distributions	density
Gamma $(2, 2)$	$f(x) = 4x\, e^{-2x}$
Lognormal $(-0.203, 0.637)$	$f(x) = \frac{1}{\sqrt{2\pi}\cdot 0.637\cdot x}\, e^{-\frac{(\log x + 0.203)^2}{2(0.637)^2}}$
Pareto $(2.732, 0.634)$	$f(x) = 0.786\frac{1}{x^{3.732}}$
Weibull $(1.44, 1.10)$	$f(x) = 1.31\left(\frac{x}{1.1}\right)^{0.44} e^{-\left(\frac{x}{1.1}\right)^{1.44}}$

Table 4.1: *Parameter values and claim size distributions*

The MC and QMC estimates are obtained using $N = 66\,000$ paths for the recursive case and for the simulation and $N = 33\,000$ for the double-recursive and iterative calculations. Since exact values are not available, we estimated them by a MC-simulation over 10 million paths for every choice of u and b.

For the recursive and double recursive calculations we use a recursion depth of $k = 66$, which leads to a 132-dimensional sequence needed for the MC- and QMC-calculations, while for the simulation it turned out to be sufficient to take a 400-dimensional sequence so that 200

consecutive claims and interoccurrence times of a risk reserve sample path can be simulated from one element of the sequence and correlations among the claim sizes and claim occurrence times are avoided.

For the iterative calculations we use a depth of $i_{max} = 66$ and enlarge the grid to $b = 7$ so that the extrapolation from the array in $0 \leq u \leq b \leq 1$ is sufficiently accurate.

All our QMC-calculations are actually hybrid Monte Carlo estimates, i.e. the initial 50 dimensions are generated by a 50-dimensional QMC sequence and the remaining dimensions are generated by a pseudo-random number generator. The use of hybrid Monte Carlo sequences has proven to be a successful modification of the QMC-technique, since for low discrepancy sequences typically the number of points needed to obtain a satisfying degree of uniformness dramatically increases with the number of dimensions. Moreover, due to the nature of our risk reserve process, the initial dimensions of the sequence have a higher impact on the solution than higher dimensions.
Throughout this chapter, we use `ran2` as our pseudo-random number generator as described in [59], which basically is an improved version of a Minimal Standard generator based on a multiplicative congruential algorithm.

The different methods and sequences used are compared via the mean square error (MSE)

$$S = \sqrt{\frac{1}{|P|} \sum_{(u,b)\in P} \Big(g(u,b) - \tilde{g}(u,b)\Big)^2},$$

where $g(u,b)$ and $\tilde{g}(u,b)$ denote the exact and the approximated value, respectively, and the set P is a grid in the triangular region $(b = 0..[0.1]..1, u = 0..[0.1]..b)$. In addition, for each method we give the maximal deviation of the approximated value from the corresponding exact value $\|\Delta\|_\infty = \max_{(u,b)\in P} \Big(g(u,b) - \tilde{g}(u,b)\Big)$.

The simulations showed that the implementation of Faure and Niederreiter $(0,s)$-sequences cannot compete with the performance of other low-discrepancy sequences for the integrands of our problems. Therefore the simulation results of these two sequences have not been included in the following considerations.

4.4.2 Error analysis

Survival probability

In Model A the survival probability can only be calculated using the simulation approach. Figure 4.2 shows the errors for the Weibull distribution and Table 4.2 gives the mean-square and the maximal error of the simulation results for each of the sequences and claim size distributions used ($N = 66\,000$).
Here, only the Sobol sequence leads to an improvement compared to the Monte Carlo simulation.

In Model B, the integral operator (4.14) can be used to calculate the survival probability, and the errors of the different methods are given in Table 4.3.

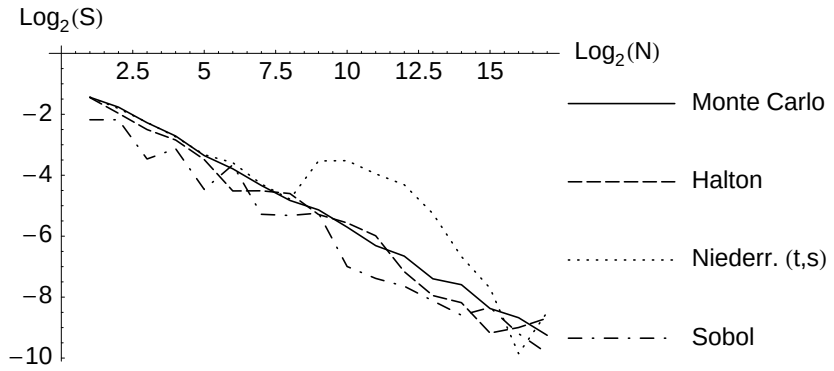

Figure 4.2: *MSE of the simulation of $\phi(u,b)$ as a function of N (Model A, Weibull distribution)*

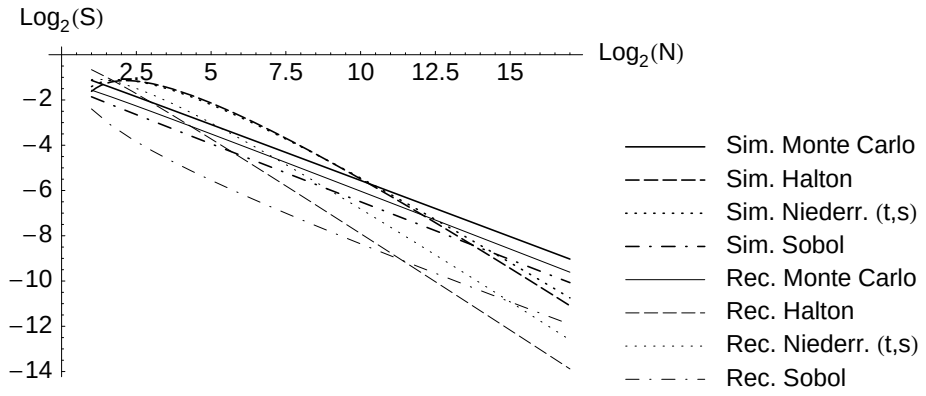

Figure 4.3: *MSE of $\phi(u,b)$ estimates as a function of N (Pareto distribution, Model B), fitted*

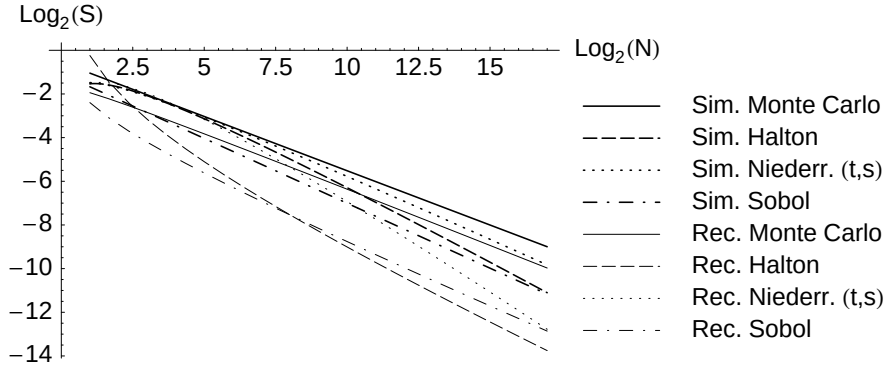

Figure 4.4: *Fitted MSE of $\phi(u,b)$ estimates (Gamma distribution, Model B)*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Gamma	S	0.001744	0.002247	0.002585	0.001111
	$\|\Delta\|_\infty$	0.004195	0.004773	0.004595	0.002393
Lognormal	S	0.001648	0.001678	0.001813	0.001437
	$\|\Delta\|_\infty$	0.003738	0.004164	0.003402	0.002687
Pareto	S	0.001888	0.002119	0.002043	0.002695
	$\|\Delta\|_\infty$	0.004477	0.004285	0.005239	0.00431
Weibull	S	0.001632	0.002423	0.002656	0.001141
	$\|\Delta\|_\infty$	0.00463	0.004685	0.004179	0.002338

Table 4.2: *Simulation errors for the survival probability in Model A*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Gamma					
Simulation	S	0.00186705	0.000698195	0.00180109	0.000303748
	$\|\Delta\|_\infty$	0.004275	0.001265	0.003022	0.000612
Recursive	S	0.00115668	0.000148828	0.000216361	0.000157103
	$\|\Delta\|_\infty$	0.003236	0.000331	0.000486	0.00036
Iterative	S	0.010934	0.0107691	0.0108299	0.0108279
	$\|\Delta\|_\infty$	0.015	0.013732	0.013807	0.013803
Lognormal					
Simulation	S	0.0018042	0.000351439	0.00195453	0.000282303
	$\|\Delta\|_\infty$	0.004136	0.001045	0.003431	0.000923
Recursive	S	0.00119801	0.000146449	0.000221486	0.000187532
	$\|\Delta\|_\infty$	0.0033	0.000332	0.000523	0.000424
Iterative	S	0.0115233	0.0113429	0.01141	0.011408
	$\|\Delta\|_\infty$	0.015954	0.014688	0.014772	0.014767
Pareto					
Simulation	S	0.00187995	0.000494491	0.00159844	0.000652944
	$\|\Delta\|_\infty$	0.004005	0.001343	0.003035	0.001666
Recursive	S	0.00144045	0.000177989	0.000210099	0.000208204
	$\|\Delta\|_\infty$	0.003893	0.000513	0.000539	0.00044
Iterative	S	0.0131512	0.0129077	0.0129771	0.012975
	$\|\Delta\|_\infty$	0.018795	0.01741	0.017498	0.017498
Weibull					
Simulation	S	0.0017325	0.000530047	0.00132312	0.000342577
	$\|\Delta\|_\infty$	0.004521	0.001225	0.002562	0.000812
Recursive	S	0.00114726	0.000155255	0.000215812	0.000147719
	$\|\Delta\|_\infty$	0.003259	0.000377	0.00047	0.00036
Iterative	S	0.0106633	0.0105005	0.0105598	0.0105578
	$\|\Delta\|_\infty$	0.014519	0.013309	0.013382	0.013378

Table 4.3: *Simulation errors for the survival probability in Model B*

To quantify the effect of using a low discrepancy sequence, we perform a regression analysis by fitting

$$\log_2(S) = a_0 + a_1 \log_2(N) + a_2 \log_2(\log_2(N)) + \varepsilon$$

to the data using a least square fit. Note that Koksma-Hlawka's inequality (2.1) could be interpreted as implying $a_1 = -1$ and $a_2 = s$, where s is the dimension of the sequence used. However, since we use a hybrid sequence and since the effective dimension may differ from the theoretical dimension, the values of a_1 and a_2 deviate from the ones above. Figures 4.3 and 4.4 show these regression fits for the Pareto and Gamma distributions, the Weibull and Lognormal distributions show a similar behaviour. In the sequel, all figures will be given in terms of their regression fits.

Expected value of the dividend payments

The simulation results for the expected value of the dividends in Model A show a clear advantage of QMC methods over MC integration. As an illustration, Figure 4.5 depicts the MSE of the simulation results as a function of N for the Weibull distribution.
While for small N, the Sobol sequence outperforms the other sequences by a factor of about 4 in terms of the MSE, for large N the Halton sequence used in the recursive algorithm is to be preferred. To quantify this effect, we introduce the efficiency gain

$$gain_i = \frac{N^*_{\mathrm{MC}}(S)}{N^*_i(S)}$$

where $N^*_{\mathrm{MC}}(S)$ is the number of paths needed in the Monte Carlo simulation to reach a given error of S, and $N^*_i(S)$ is the corresponding N using an alternative method. Figure 4.6 shows that except for the $(0, s)$-nets (which are not plotted) all methods are an improvement in efficiency compared to Monte Carlo simulation and the gain increases at smaller errors.
The above comparisons are performed with respect to N, the number of summands in the MC and QMC approximations. However, it might be preferable to compare the accuracy of the various numerical solution techniques with respect to calculation time, see Table 4.4 and Figure 4.7. It turns out that the performance of the double recursive algorithm is still competitive when measured with respect to calculation time; however, the recursive method using Sobol's sequence seems preferable. One also has to notice that the Niederreiter sequence in base 2 gives results worse than the Monte Carlo methods, which is due to the fact that in Model A, all 50 QMC dimensions of the elements are relevant. However, the quality of the different sequences also depends on the claim distribution used, as a comparison with Figure 4.8 shows. Moreover, it is clearly visible that although the simulation is the fastest method using N elements, it is actually the worst when the error is considered.
Figure 4.8 furthermore shows an attempt to compare the results of the iterative method with the other methods. Here some care is needed, since the recursion depth k of the recursive algorithms is fixed in advance (and thus also the error caused by choosing k) and the number N of sequence elements is increased with time. On the other hand, for the iterative method N is fixed in advance and the iteration depth is increased with time.

The calculations show that for all iterative methods and the above choice of initial values, it takes 40 to 50 iterations until the approximations are sufficiently close to the exact values. Thus one might try to improve the efficiency of the algorithm by first simulating the process

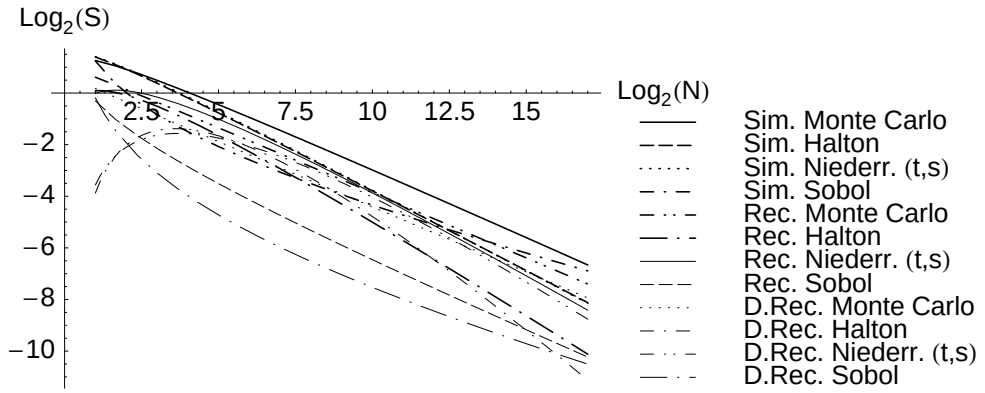

Figure 4.5: *MSE of expected dividend payments (Weibull distribution, Model A)*

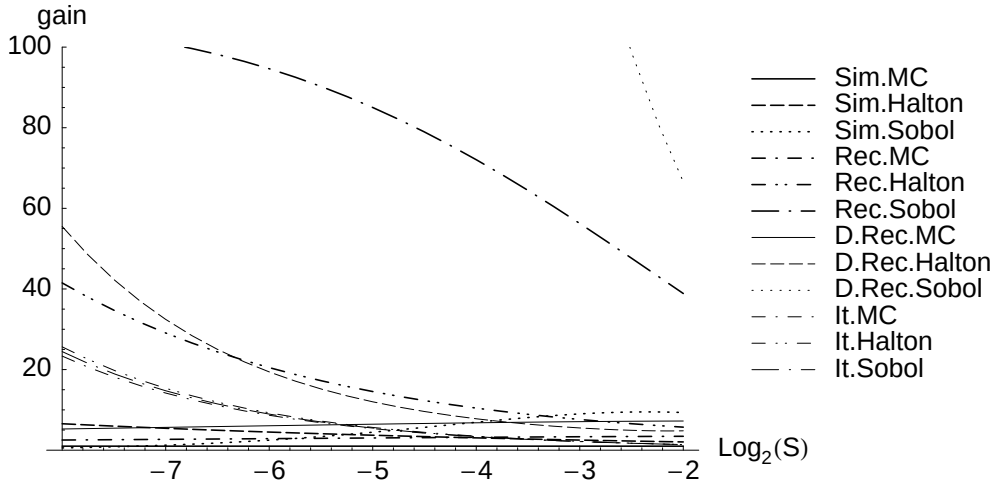

Figure 4.6: *Efficiency gain for calculation of expected dividends (Model A, Weibull distribution)*

	Simulation	Iterative	Recursive	Double Rec.
Monte Carlo	209.551	413.685	746.824	3062.61
Halton	186.033	407.416	802.648	3162.44
Niederr. (t,s)	250.527	388.903	732.736	3036.35
Sobol	186.813	174.247	947.162	2811.14
Faure	393.982	174.3	732.161	3028.67
(0,s) net	179.47	389.017	732.048	3028.34

Table 4.4: *Calculation times in seconds (expected dividends, Lognormal distribution, Model A)*

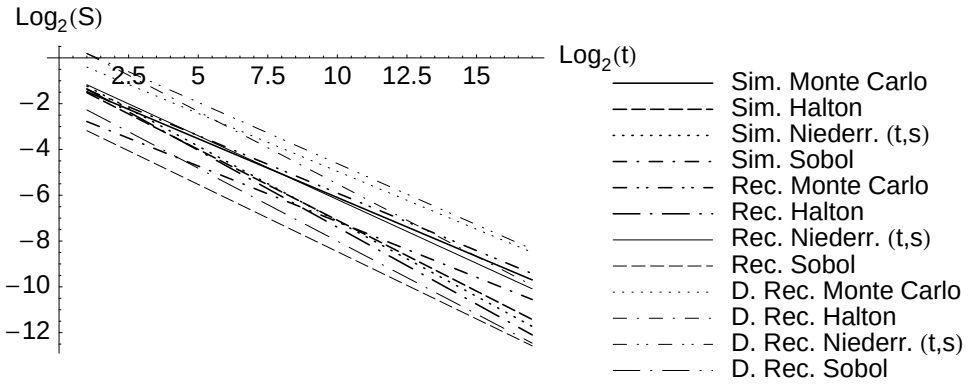

Figure 4.7: *MSE of the expected dividend payments as a function of calculation time (Lognormal distribution, Model A)*

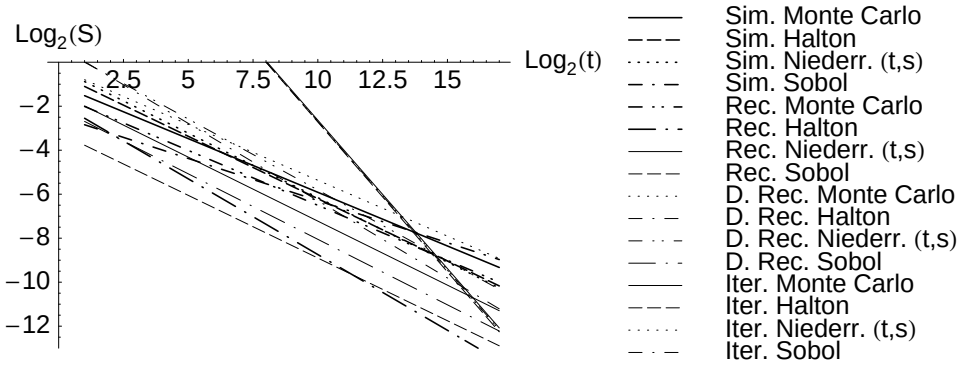

Figure 4.8: *Calculation times in seconds (expected dividend payments, Pareto distribution, Model A)*

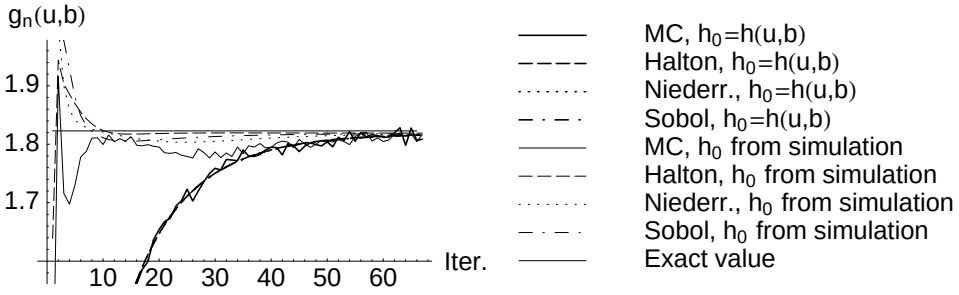

Figure 4.9: *Convergence of the dividend payments estimate for* $(u, b) = (0, 1)$ *(iterative method using* $h(u, b)$ *and 10 simulation paths, resp., as initial values for the grid, Lognormal Distribution, Model A)*

(with a small number N_s of simulation paths) and then use these simulation estimates as initial values for the iterative procedure. Naturally, this choice of initial values for the grid has a much lower initial error than using $h(u_j, b_k)$, but is much less smooth than h which lowers the convergence rate of the algorithm. However, Figure 4.9 shows that even for $N_s = 10$, this approach substantially improves the plain iterative method.

In contrast to Model A, in Model B the effective dimension of the simulation as well as the dimension of the iterated integral operator becomes very small. It turns out that for the Weibull, Gamma and Lognormal distribution on average only about 8 claims or iterations of the operator are needed until b becomes large enough so that practically no dividends will be paid out any more, and for the Pareto distribution only 3 to 6 claims are needed. So while Model A is a high-dimensional problem (with 50 QMC dimensions in our case), Model B is low-dimensional with an effective dimension of less than 20 (every claim or recursive step needs two dimensions), and one can expect the usual good properties for low-discrepancy sequences in low-dimensional environments. Figure 4.10 shows the MSE as a function of N for the Gamma distribution. The behavior is similar to Model A, except that now the (t, s) nets exhibit a good convergence which is even better than Halton's sequence (other distributions also show a similar behaviour). Here again, Sobol's sequence is by far the best for a low number N of points, but loses quality for larger N (cf. Figure 4.11). Figure 4.12 depicts the MSE as a function of calculation time.

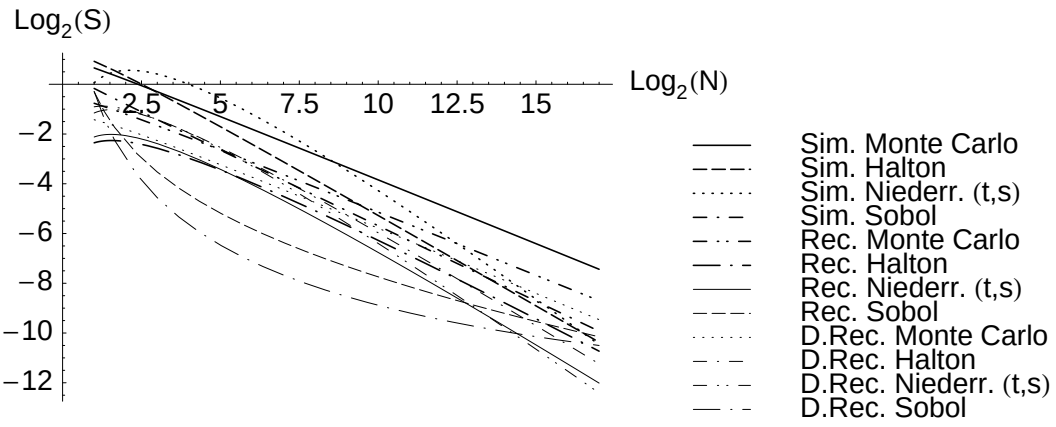

Figure 4.10: *MSE of expected dividend payments (Gamma distribution, Model B)*

As pointed out by Bratley, Fox and Niederreiter [14] and numerically investigated by Radovic, Sobol' and Tichy [60], the distribution behavior of the initial elements of (t, s) nets is not satisfying due the so-called leading-zeros phenomenon. Thus, it has been suggested to start the sequence at $n = p^k$ (called the "step"), with k being at least the maximum degree of the polynomials used to generate the sequence and p denoting the base of the construction. As Figure 4.13 shows, introducing the step indeed considerably improves the performance of (t, s)-sequences for moderate sizes of N.

The iterative method in Model B is very sensitive to the choice of the mesh-size of the grid, so that one has to select a very small mesh-size to get good results. It turns out that for a

mesh-size of 0.5 and 0.2 for values of $b > 1$ up to $b = 4$ (when no dividends will be paid out any more due to the absorbing barrier), the obtained values typically still are about 0.02 and 0.003, respectively, greater than the exact value.

Tables 4.5 and 4.6 give a complete list of all the errors of the various methods for all distributions in Model A and B, respectively.

4.4.3 Model analysis

In this final section we want to use our simulation results to investigate the sensitivity of the probability of survival and the expected dividend payments to the claim size distribution and to the consideration of interest rates. For that purpose we fix a value of b (the initial height of the dividend barrier) and plot $\phi(u, b)$ and $W(u, b)$ against u. Figures 4.14 and 4.15 depict $\phi(u, b)$ for the choice $b = 1$ in Model A and B, respectively. They also include the corresponding plot for $i = 0$. Note that all the distributions are normalized to have equal mean and variance. Since heavy tail distributions exhibit their characteristic behavior for larger values of u, we also give a plot of $\phi(u, b)$ against u for $b = 30$ in Model A (Figure 4.16). A double logarithmic plot of the ruin probability (Figure 4.17) against u (for fixed $b = 30$) displays a similar behaviour of the heavy-tail distributions in our dividend barrier model as it has been obtained by simulation of surplus processes without a barrier (see e.g. ASMUSSEN AND BINSWANGER [9]). Note that the Pareto distribution implies a qualitatively different behavior of $\phi(u, b)$ for large u. Related simulations have shown that for larger values of the variance, this is also the case for Log-normal distributions (cf. HEERSINK [39]).

Figures 4.18 and 4.19 show the dependence of the expected dividend payments on u for fixed values of $b = 1$ and $b = 30$, respectively. Here it turns out that the consideration of gaining interest on the free reserve has a large effect on the values of $W(u, b)$, whereas the choice of the claim size distribution is more or less negligible.
As an illustration Tables 4.7 to 4.10 give the exact values of $\phi(u, b)$ and $W(u, b)$ for the Gamma distribution in Model A and B, respectively.

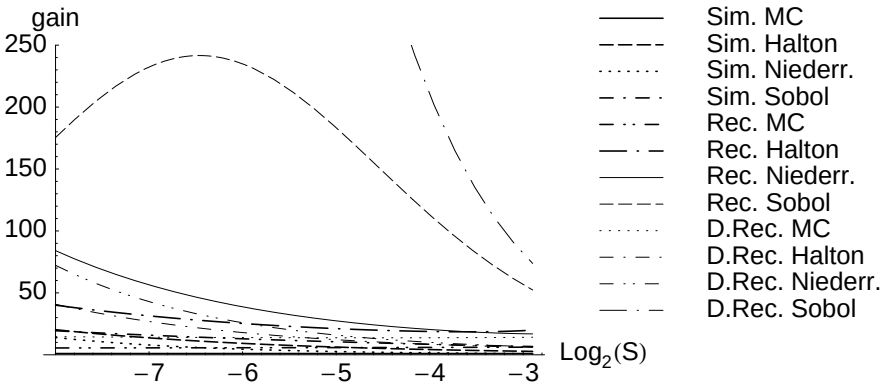

Figure 4.11: *Efficiency gain for calculation of expected dividends (Model B, Gamma distribution)*

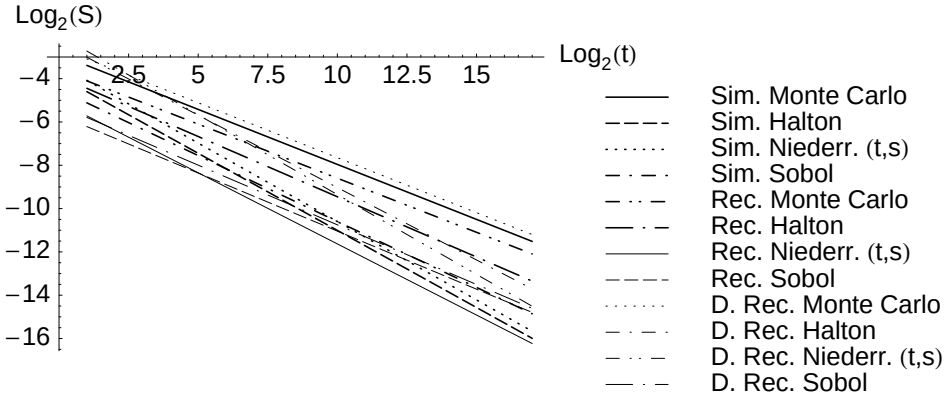

Figure 4.12: *MSE of expected dividend payments (Gamma distribution, Model B) as a function of calculation time*

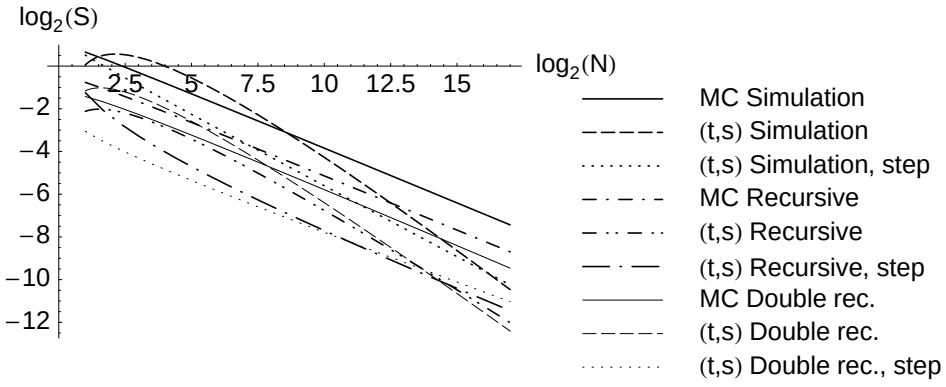

Figure 4.13: *MSE of expected dividends, step=53^3 (Gamma distribution, Model B)*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Gamma					
Simulation	S	0.0102387	0.00589075	0.00644128	0.0122924
	$\|\Delta\|_\infty$	0.02573	0.01542	0.01382	0.01653
Recursive	S	0.00690747	0.00186699	0.00504432	0.00138267
	$\|\Delta\|_\infty$	0.02303	0.00331	0.00699	0.00274
Double Rec.	S	0.00699453	0.00185775	0.00333911	0.00110242
	$\|\Delta\|_\infty$	0.01626	0.00362	0.00535	0.00249
Iterative	S	0.016958	0.0153584	0.0160474	0.0160834
	$\|\Delta\|_\infty$	0.02798	0.02057	0.02143	0.02146
Lognormal					
Simulation	S	0.0104705	0.00609589	0.00585217	0.0139076
	$\|\Delta\|_\infty$	0.02215	0.01477	0.01377	0.01843
Recursive	S	0.00719265	0.00164129	0.00622522	0.0016576
	$\|\Delta\|_\infty$	0.02355	0.00323	0.00863	0.00296
Double Rec.	S	0.00708872	0.00148873	0.00484965	0.00100742
	$\|\Delta\|_\infty$	0.01592	0.00319	0.00739	0.00248
Iterative	S	0.0170572	0.0154885	0.0162279	0.0162694
	$\|\Delta\|_\infty$	0.02873	0.02154	0.0225	0.02254
Pareto					
Simulation	S	0.0111856	0.0130283	0.0127553	0.0221408
	$\|\Delta\|_\infty$	0.0297	0.0237	0.02513	0.0275
Recursive	S	0.00828872	0.00167252	0.00790734	0.00251632
	$\|\Delta\|_\infty$	0.02591	0.00407	0.0114	0.00455
Double Rec.	S	0.00767502	0.00131978	0.00807013	0.00119566
	$\|\Delta\|_\infty$	0.01775	0.00371	0.0114	0.00322
Iterative	S	0.0216169	0.0206179	0.0213459	0.0214136
	$\|\Delta\|_\infty$	0.03623	0.02909	0.03007	0.03013
Weibull					
Simulation	S	0.00985157	0.005163	0.00685562	0.012519
	$\|\Delta\|_\infty$	0.02465	0.01205	0.01505	0.01741
Recursive	S	0.0068602	0.00201594	0.00457433	0.00133397
	$\|\Delta\|_\infty$	0.02297	0.00354	0.00627	0.00272
Double Rec.	S	0.00699997	0.00211651	0.002768	0.00110075
	$\|\Delta\|_\infty$	0.01634	0.00384	0.00449	0.00236
Iterative	S	0.016865	0.0152578	0.0159104	0.0159438
	$\|\Delta\|_\infty$	0.02783	0.02037	0.02119	0.02122

Table 4.5: *Errors of the various methods for expected dividend payments in Model A*

		Monte Carlo	Halton	Niederr. (t,s)	Sobol
Gamma					
Simulation	S	0.00548782	0.00101735	0.000896726	0.00102161
	$\|\Delta\|_\infty$	0.01285	0.00238	0.00201	0.00246
Recursive	S	0.00258046	0.000573281	0.00048327	0.000696459
	$\|\Delta\|_\infty$	0.00652	0.00171	0.00151	0.00201
Double Rec.	S	0.00237654	0.00062463	0.000638122	0.000951905
	$\|\Delta\|_\infty$	0.00559	0.00185	0.00188	0.00239
Iterative	S	0.0176776	0.018468	0.0183926	0.018383
	$\|\Delta\|_\infty$	0.02558	0.02328	0.02317	0.02316
Lognormal					
Simulation	S	0.00496058	0.000959204	0.00115588	0.000858514
	$\|\Delta\|_\infty$	0.01278	0.00206	0.00274	0.00289
Recursive	S	0.00263774	0.000567729	0.000465821	0.000696362
	$\|\Delta\|_\infty$	0.00689	0.00182	0.00161	0.00215
Double Rec.	S	0.00229559	0.000588638	0.000605718	0.000989495
	$\|\Delta\|_\infty$	0.00526	0.00193	0.00197	0.00256
Iterative	S	0.0216594	0.0225257	0.0224476	0.0224349
	$\|\Delta\|_\infty$	0.03055	0.02876	0.02864	0.02864
Pareto					
Simulation	S	0.005052	0.000733847	0.00111069	0.000917839
	$\|\Delta\|_\infty$	0.01517	0.00156	0.00292	0.00219
Recursive	S	0.0028135	0.000453588	0.000424179	0.000635163
	$\|\Delta\|_\infty$	0.00802	0.00166	0.00156	0.00204
Double Rec.	S	0.00197331	0.000496007	0.000579632	0.000935201
	$\|\Delta\|_\infty$	0.003926	0.00178	0.00203	0.00245
Iterative	S	0.0301029	0.0310616	0.031024	0.0310033
	$\|\Delta\|_\infty$	0.04231	0.04179	0.04173	0.04171
Weibull					
Simulation	S	0.00533089	0.000849476	0.000803501	0.000875673
	$\|\Delta\|_\infty$	0.0136	0.00242	0.00216	0.00203
Recursive	S	0.00257658	0.000570681	0.000486129	0.000688766
	$\|\Delta\|_\infty$	0.00652	0.00178	0.00158	0.00207
Double Rec.	S	0.00241979	0.000619308	0.000643352	0.000921779
	$\|\Delta\|_\infty$	0.00572	0.00191	0.00196	0.00243
Iterative	S	0.016305	0.0170657	0.0169917	0.016983
	$\|\Delta\|_\infty$	0.02382	0.02149	0.02138	0.02138

Table 4.6: *Errors of the various methods for expected dividend payments in Model B*

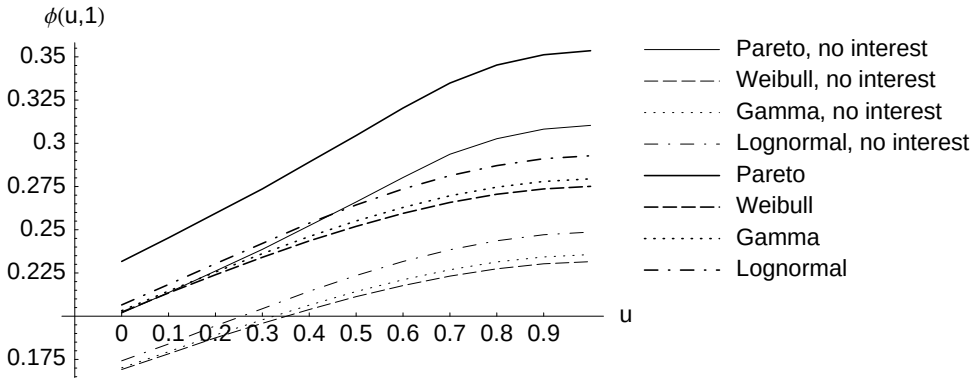

Figure 4.14: *Survival probability* $\phi(u,1)$ *in Model A*

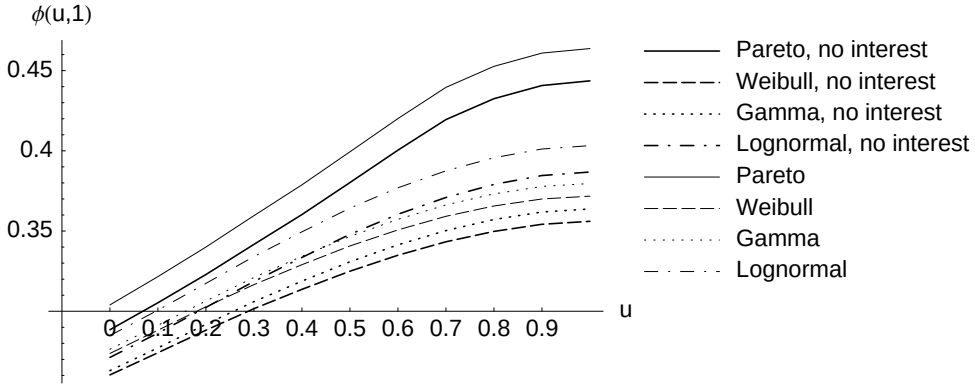

Figure 4.15: *Survival probability* $\phi(u,1)$ *in Model B*

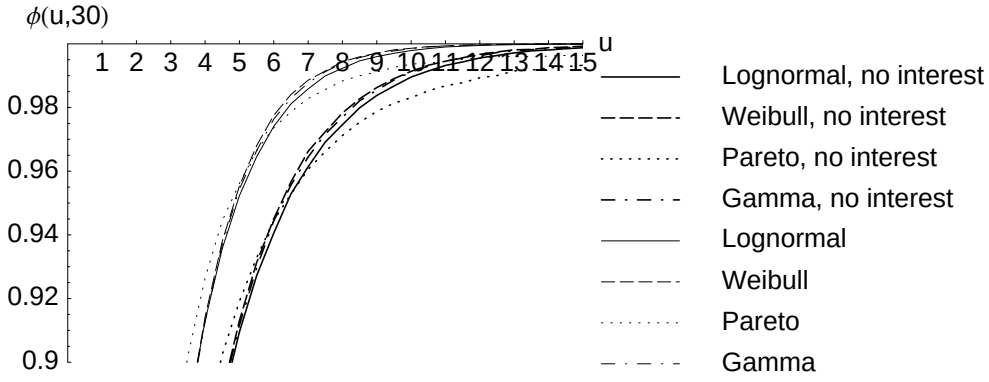

Figure 4.16: *Survival probability* $\phi(u,30)$ *in Model A*

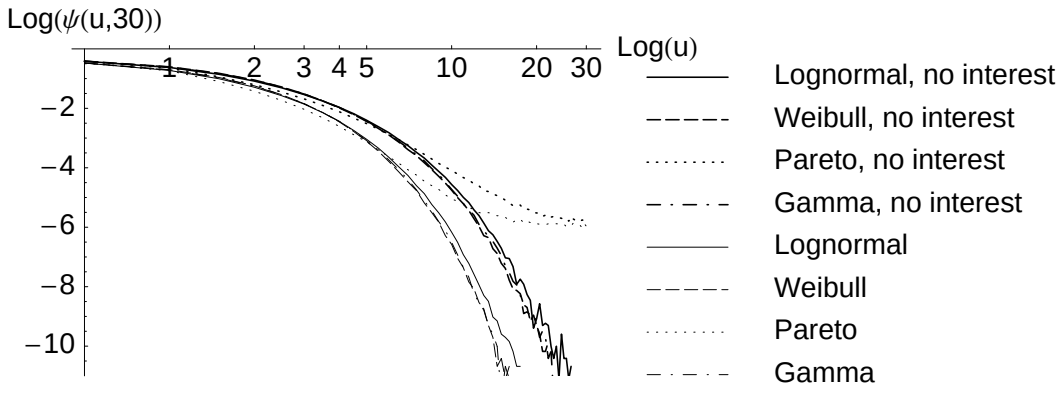

Figure 4.17: *Log-log plot of the ruin probability $\psi(u, 30)$ against u (Model A)*

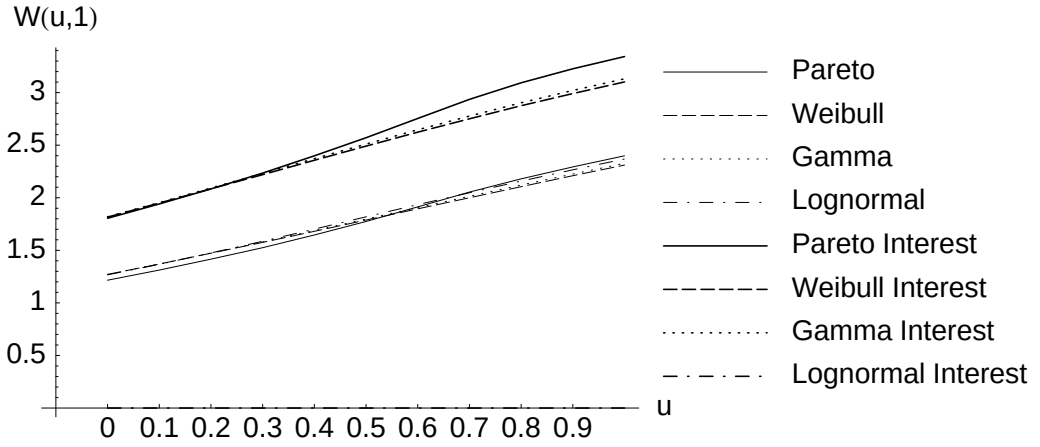

Figure 4.18: *Expected dividend payments $W(u, 1)$ in Model A*

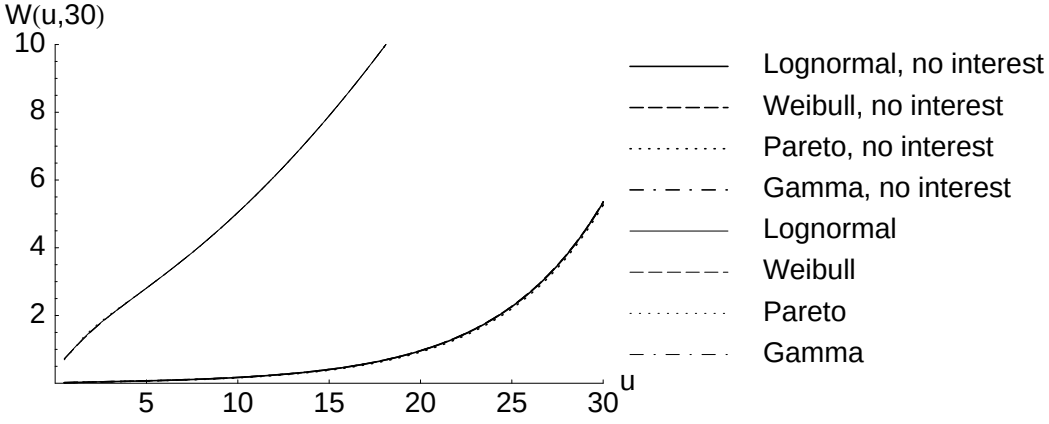

Figure 4.19: *Expected dividend payments $W(u, 30)$ in Model A*

$b\backslash x$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.
0	0.18										
0.1	0.18	0.188									
0.2	0.181	0.189	0.197								
0.3	0.182	0.191	0.199	0.206							
0.4	0.184	0.193	0.201	0.209	0.214						
0.5	0.187	0.196	0.204	0.212	0.219	0.223					
0.6	0.189	0.199	0.208	0.217	0.224	0.23	0.233				
0.7	0.193	0.202	0.212	0.221	0.229	0.235	0.241	0.243			
0.8	0.196	0.206	0.216	0.226	0.235	0.242	0.248	0.252	0.254		
0.9	0.2	0.21	0.221	0.231	0.24	0.249	0.256	0.261	0.265	0.266	
1.	0.203	0.215	0.226	0.236	0.246	0.255	0.263	0.27	0.275	0.278	0.279

Table 4.7: *Exact values for the survival probability of the Gamma distribution, Model A*

$b\backslash x$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.
0	1.78										
0.1	1.782	1.904									
0.2	1.783	1.906	2.036								
0.3	1.785	1.911	2.04	2.17							
0.4	1.789	1.915	2.048	2.177	2.303						
0.5	1.794	1.922	2.053	2.188	2.315	2.436					
0.6	1.799	1.927	2.062	2.196	2.327	2.456	2.572				
0.7	1.805	1.935	2.069	2.205	2.339	2.468	2.594	2.709			
0.8	1.811	1.942	2.076	2.215	2.352	2.485	2.614	2.736	2.85		
0.9	1.816	1.948	2.084	2.224	2.362	2.498	2.632	2.757	2.88	2.99	
1.	1.818	1.954	2.093	2.231	2.372	2.51	2.646	2.776	2.903	3.02	3.131

Table 4.8: *Exact values for the dividends of the Gamma distribution, Model A*

$b\backslash x$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.
0	0.245										
0.1	0.245	0.256									
0.2	0.246	0.257	0.267								
0.3	0.248	0.259	0.27	0.279							
0.4	0.251	0.262	0.274	0.283	0.291						
0.5	0.254	0.266	0.278	0.289	0.297	0.303					
0.6	0.258	0.27	0.283	0.294	0.304	0.312	0.316				
0.7	0.262	0.275	0.288	0.3	0.311	0.32	0.327	0.33			
0.8	0.266	0.28	0.294	0.307	0.319	0.329	0.337	0.343	0.346		
0.9	0.271	0.286	0.3	0.314	0.327	0.338	0.348	0.355	0.36	0.362	
1.	0.276	0.291	0.306	0.321	0.334	0.347	0.357	0.366	0.373	0.378	0.38

Table 4.9: *Exact values for the survival probability of the Gamma distribution, Model B*

$b\backslash x$	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.
0	1.096										
0.1	1.096	1.189									
0.2	1.092	1.186	1.286								
0.3	1.088	1.182	1.281	1.384							
0.4	1.082	1.176	1.276	1.377	1.481						
0.5	1.075	1.168	1.266	1.369	1.472	1.577					
0.6	1.064	1.157	1.256	1.357	1.461	1.566	1.671				
0.7	1.053	1.146	1.243	1.343	1.447	1.551	1.655	1.762			
0.8	1.039	1.132	1.228	1.327	1.43	1.535	1.64	1.745	1.85		
0.9	1.024	1.115	1.21	1.309	1.411	1.514	1.619	1.724	1.829	1.935	
1.	1.006	1.095	1.19	1.289	1.388	1.49	1.594	1.699	1.804	1.909	2.014

Table 4.10: *Exact values for the dividends of the Gamma distribution, Model B*

Part II

Runge-Kutta QMC Methods for Delay Differential Equations

Chapter 5

QMC methods for the solution of delay differential equations

Contents

In this chapter Quasi-Monte Carlo methods for Runge Kutta solution techniques of differential equations, which were developed by Stengle, Lécot, Coulibaly and Koudiraty, are extended to delay differential equations of the form $y'(t) = f(t, y(t), y(t - \tau(t)))$. The retarded argument is approximated by interpolation, after which the conventional (Quasi-)Monte Carlo Runge Kutta methods can be applied. We give a proof of the convergence of this method and its order in a general form, which does not depend on a specific Quasi-Monte Carlo Runge Kutta method. Finally, a numerical investigation shows that similar to ordinary differential equations this quasi-randomized method leads to an improvement for heavily oscillating delay differential equations compared even to high order Runge Kutta schemes.
The content of this chapter is being published in [46].

5.1 Introduction

In physics and other engineering subjects, the rate of change of a process $y(t)$ often does not only depend on the value of $y(t)$ at time t, but also on the values of the process in the past. The "delay differential equation" for the process then can be written in the general form:

$$y'(t) = f(t, y(t), y(t - \tau_1(t)), ...y(t - \tau_k(t))) \quad \text{for } t > t_0, \tag{5.1}$$

$$y(t) = y_0(t) \quad \text{for } t \leq t_0 \, . \tag{5.2}$$

The most noticeable difference to ordinary initial value problems is that the initial value must be given as a function on an interval $[\inf_{t\geq t_0, 0\leq j\leq k} t - \tau_j(t), t_0]$ instead of only the value at the starting point t_0.
In [65] and [66] Stengle proposed a (randomized) Monte Carlo algorithm for the solution of the initial value problem

$$y'(t) = f(t, y(t)), \quad 0 < t < T, \qquad y(0) = y_0 \ , \tag{5.3}$$

where f is smooth in y, but only bounded and Borel measurable in t. The family of solution methods he proposed is akin to the Runge-Kutta family, and thus called the Runge-Kutta Monte Carlo (RKMC) family. In his derivation he does not discretize the time, but only the spatial dimensions y, and solves the remaining integral equation using Monte Carlo integration. Lécot [52], Coulibaly and Lécot [17] and Lécot and Koudiraty [50] generalized this method for orders up to three, and additionally used Quasi-Monte Carlo methods to calculate the resulting integral (thus the name RKQMC methods for them).
In their articles, Stengle, Coulibaly, Koudiraty and Lécot apply the Runge-Kutta (Quasi) Monte Carlo methods only to ordinary differential equation with initial conditions as given in equation (5.3). We will first give a brief outline of their methods here, before applying them to delay differential equations. Under the assumptions mentioned above (f does not even have to be differentiable in t, let alone smooth), $f(y, y(t))$ is Taylor-expanded only with respect to $y(t)$ and the differential equation is recursively substituted into itself. This leads to an equation of the form (see [50]):

$$f(t_0 + h) = y(t_0) + \sum_{i=1}^{s} \int_{t_0}^{u_0} \cdots \int_{t_0}^{u_{i-1}} F_i\left(u_1, \ldots, u_i; y(t_0)\right) du_i \ldots du_1 + \mathcal{O}(h^{s+1}) \ , \tag{5.4}$$

where $u_0 = t_0 + h$ and the F_k, $1 \leq k \leq s$, are defined recursively by $F_0(y) = y$, $F_i := D_y F_{i-1} f(u_i, y)$. The sum is then combined into one s-dimensional integral over a new function G_s with $\bar{u} = \left(u_{\pi(1)}, \ldots, u_{\pi(s)}\right)$ and a permutation π such that $u_{\pi(1)} \leq u_{\pi(2)} \leq \cdots \leq u_{\pi(s)}$:

$$f(t_0 + h) = y(t_0) + \frac{1}{s! h_n^{s-1}} \int_{(t_0, t_0+h)^s} G_s\left(\bar{u}; y(t_0)\right) du + \mathcal{O}(h^{s+1}) \ , \tag{5.5}$$

The function G_s is rewritten using a suitable identity (and according order conditions on the coefficients) to get rid of the derivatives of F_i with respect to $y(t)$. Finally, the remaining integral is approximated by Monte Carlo integration. The s-order RKQMC method generates a sequence $(y_n)_{0\leq n}$ defined by (see e.g. [52] or [50]):

$$y_{n+1} = y_n + \frac{h_n}{s! N} \sum_{0\leq j<N} G_s\left(t_{n,j}, y(t_n)\right) \ , \tag{5.6}$$

where $t_{n,j} = t_n + h_n \bar{x}_j$ and $\bar{x}_j^{(1)} \leq \bar{x}_j^{(2)} \leq \cdots \leq \bar{x}_j^{(s)}$ are the elements of the s-dimensional uniformly distributed sequence $(x_j)_{0\leq j\leq N}$ with their dimensions sorted in ascending order. The explicit forms of the functions G_1, G_2 and G_3 as given by Lécot [52] and Koudiraty [50] are shown in equations (5.14) to (5.16), the corresponding order conditions can be found in the cited articles (esp. [50, equations (4.12) to (4.15)]).
We will use this method and extend it to delay differential equations in the rest of the chapter. Our discussion will not depend on one specific RKQMC method, however, we will also give specific proofs for the convergence and its order of the RKQMC1 [52], RKQMC2 [52] and RKQMC3 [50] methods applied to delay differential equations.

5.2 Description of the problem

We consider initial value problems for delay (also called retarded) differential equations (DDE) with one retarded argument having the form

$$\begin{aligned} y'(t) &= f\left(t, y(t), y(t-\tau(t))\right), \quad \text{for } t \geq t_0, \\ y(t) &= \phi(t), \text{ for } t \leq t_0 \, , \end{aligned} \tag{5.7}$$

where $y(t)$ is a d-dimensional real-valued function (which is in general not smooth), $\tau(t)$ is the continuous delay function, which we assume to be bounded from below ($0 < \tau_0 = \inf_{t \geq t_0} \tau(t)$). Furthermore, $\phi(t)$ is the initial function, which is assumed to be piecewise continuous at least on the interval $[\inf_{t \geq t_0}(t - \tau(t)), t_0]$. We will not discuss the simplest case of a constant delay function $\tau(t) = \tau_0$, but instead the more general case of a delay function $\tau(t)$ satisfying

$$t_1 - \tau(t_1) \leq t_2 - \tau(t_2) \quad \text{for } t_1 \leq t_2 \, , \tag{5.8}$$

i.e. $T(t) := t - \tau(t)$ is an increasing function of t.

Solutions of the differential equation (5.7) are continuous for all $t > t_0$ and piecewise differentiable. Provided that $f(t, y, z)$ is (locally) Lipschitz w.r.t. y and z, the existence and uniqueness theorems for ordinary differential equations carry over verbatim. However, even if f and ϕ are smooth, the solution $y(t)$ is only smooth if $\phi(t)$ solves the differential equation (5.7). Otherwise, the solution $y(t)$ will have discontinuous derivatives $y^{(j)}(t_k)$ for $j \geq k$ at times t_k, which are recursively defined by $t_k = T^{-1}(t_{k-1})$ (see [55]). This means that with each additional time interval of length $h_k = t_k - t_{k-1}$, the discontinuities are smoothed out, and all the derivatives up to the k-th are continuous. For a constant delay function, we have $T^{-1}(t_{k-1}) = t_{k-1} + \tau_0$, the solution $y(t)$ has k continuous derivatives at $t_k = t_0 + kau_0$, and $y^{(k+1)}(t)$ in general has a jump discontinuity at t_k. If the initial function $\phi(t)$ or its derivatives have discontinuities, a similar statement holds with t_0 replaced by the discontinuities of the initial function (see [55]).

5.3 Runge Kutta QMC methods applied to delay differential equations

We will first discuss how the delayed argument $y(t-\tau(t))$ can be treated, and then deduce the Runge Kutta Quasi-Monte Carlo scheme for delay differential equations, which is a straight generalization of Stengle's, Lécot's and Koudiraty's schemes applied to this type of differential equations. However, due to the special nature of delay differential equations, some things need to be handled more carefully.

5.3.1 Treatment of the retarded value

The main obstacle when solving delay differential equations is the way to treat the additional argument $y(t-\tau(t))$ which depends on the past values of the solution. Several attempts have been made (e.g. by Bellman [11] et al.) to calculate the retarded values by recursively using the DDE itself, however, with this approach the number of calculations increase drastically with T. The other approach to the retarded argument is interpolation (see e.g. [56]), which

we will use. Since for the numerical solution of the differential equation the time is split into discrete time steps, labeled with index n, we have the approximated values of the solution at the discrete times t_j, $0 \leq j < n$, available when calculating $y_n = y(t_n)$ for $n \geq 0$. It seems natural to use an interpolating function $P_q(t; (y_i)_{i \leq n})$ to approximate the values $y(t - \tau(t))$ from the values $y(t_j)$, $0 \leq j < n$, for $t - \tau(t) \geq t_0$. If $t - \tau(t) < t_0$, the initial function $\phi(t)$ can be used for the exact values of the solution.
Since the solutions of the differential equation are not smooth, the interpolation must be done using only past values with $\bar{\tau}_1 \leq t_k \leq \cdots \leq t - \tau(t) \leq \cdots \leq t_{k+m} \leq \bar{\tau}_2$ where $(\bar{\tau}_1, \bar{\tau}_2)$ is the largest interval of smoothness containing $t - \tau(t)$, i.e. $\bar{\tau}_1$ and $\bar{\tau}_2$ are the closest points where the derivatives of the solution have discontinuities as discussed in the last section. As the value of the derivative of $y(t)$ can be easily calculated for each time t_j by simply inserting into the differential equation, Hermite interpolation (as investigated in [55]) is used here to reduce the number of required support points to half the number required by ordinary Lagrange interpolation. Although the interpolating function then also depends on the approximated values of the derivative of $y(t)$, we will suppress this in our notation.

Note that the requirement of the interpolation points lying inside an interval of smoothness also poses a practical restriction on the value of the delay function:

$$\tau(t) \geq ph \qquad \text{or} \qquad h \leq \frac{\tau(t)}{p}\,, \tag{5.9}$$

where p is the number of points needed for interpolation, and h is the time step, if the time is discretized into fixed time intervals. Note also, that this restriction is sufficient, but not the best possible, and of course it can also be seen as an upper bound for the time step h.
If step-size control methods are applied to the Runge-Kutta method (i.e. the time step h_j of the j-th step is not fixed but chosen such that the error does not exceed a prescribed threshold), the restriction reads $\tau(t) \geq \sum_{j=0}^{p-1} h_{\nu-j}$ where ν is the largest number for which $t_\nu = t_0 + \sum_{j=1}^{\nu} h_j < t$, assuming a smooth initial function. For a non-smooth initial function the additional discontinuities of the derivatives force an even tighter bound.
If these restrictions are not fulfilled, one can still try to obtain a RKQMC scheme by a Taylor-Expansion in the retarded argument as well, although this would complicate the algorithm and its derivation considerably.

These restrictions are the reason why mostly low-order methods for delay differential equations have been considered. In the sequel, we will also not develop high order methods, but instead look at the effect of the application of Quasi-Monte Carlo methods to Runge Kutta algorithms for delay differential equations.

By approximating the retarded value by the past values of the solution, we can directly insert this interpolation function

$$z(t) = z_{(y_i)_{i \leq n}}(t) = \begin{cases} \phi(t - \tau(t)), & \text{if } t - \tau(t) \leq t_0 \\ P_q\left(t - \tau(t); (y_i), (y_i')\right) & \text{otherwise} \end{cases} \tag{5.10}$$

into the differential equation and arrive at an ordinary differential equation

$$y'(t) = f(t, y(t), y(t - \tau(t))) \approx f(t, y(t), z(t)) = g(t, y(t))\,. \tag{5.11}$$

Applying the RKQMC methods for ordinary differential equations (equation (5.6) or [17, 52, 50]), we can approximate a solution of the delay differential equation.

5.3.2 The RKQMC schemes for DDE

In the sequel, we will use a notation which is independent of the Runge Kutta scheme and only where necessary use scheme-specific results. We furthermore assume $h_n < 1$ for all n. Using the notation of [50], the exact value $y(t_{n+1})$ is approximated by the s-order Runge-Kutta (Q)MC scheme as:

$$y(t_{n+1}) = y(t_n) + \frac{1}{s! h_n^{s-1}} \int_{(t_n, t_{n+1})^s} G_s\left(\bar{u}; y(t_n); y(t \le t_n)\right) du + h_n \varepsilon_n \tag{5.12}$$

where G_s is the differential increment function of the scheme, and ε_n is the local truncation error. This integral is then approximated by Quasi-Monte Carlo integration and interpolation of the retarded argument as

$$y_{n+1} = y_n + \frac{h_n}{s! N} \sum_{0 \le j < N} \tilde{G}_s\left(\bar{t}_{j,n}; y_n, (y_i)_{i \le n}\right), \tag{5.13}$$

where $\tilde{G}_s(t; y; (y_i)_{i \le n}) := G_s\left(t; y; z_{(y_i)_{i \le n}}(\cdot)\right)$ denotes the function G_s with the retarded values interpolated from the points $(y_i)_{i \le n}$.
For the first-, second- and third-order RKQMC schemes of Lécot and Koudiraty, the corresponding functions G_s are:

$$G_1\left(u; y; z(\cdot)\right) = f\left(u, y, z(u - \tau(u))\right) \tag{5.14}$$

$$\begin{aligned} G_2\left(\bar{u}; y; z(\cdot)\right) = {} & f\left(\bar{u}_1, y, z(\bar{u}_1 - \tau(\bar{u}_1))\right) + \frac{1}{\beta} f\left(\bar{u}_2, y, z(\bar{u}_2 - \tau(\bar{u}_2))\right) + \\ & + \frac{1}{\alpha} f\left(\bar{u}_2, y + \alpha h_n f\left(\bar{u}_1, y, \bar{u}_1 - \tau(\bar{u}_1)\right), z(\bar{u}_2 - \tau(\bar{u}_2))\right) \end{aligned} \tag{5.15}$$

$$\begin{aligned} G_3\left(\bar{u}; y; z(\cdot)\right) = {} & a_1 f(\bar{u}_1, y, z(\bar{u}_1 - \tau(\bar{u}_1))) + \\ & + \sum_{l=1}^{L_2} a_{2,l} f\left(\bar{u}_2, y + b_{2,l} h_n f(\bar{u}_1, y, z(\bar{u}_1 - \tau(\bar{u}_1))), z(\bar{u}_1 - \tau(\bar{u}_1))\right) + \\ & + \sum_{l=1}^{L_3} a_{3,l} \Big(\bar{u}_3, y + b_{3,l}^{(1)} h_n f\left(\bar{u}_1, y, z(\bar{u}_1 - \tau(\bar{u}_1))\right) + \\ & \quad + b_{3,l}^{(2)} h_n f\left(\bar{u}_2, y_n + c_{3,l} h_n(\bar{u}_1, y_n), z(\bar{u}_2 - \tau(\bar{u}_2))\right), z(\bar{u}_1 - \tau(\bar{u}_1))\Big) \end{aligned} \tag{5.16}$$

The condition on α and β is $\frac{1}{\alpha} + \frac{1}{\beta} = 1$, and the conditions for the coefficients in G_3 can be found in [50].

5.3.3 Convergence proof

In the convergence proof we will use the following definitions for the Runge Kutta error δ_n, the QMC approximation error d_n, the interpolation error η_n and the local truncation error ε_n:

$$\delta_n := \frac{1}{s!h_n^s} \int_{(t_n,t_{n+1})^s} \left\{ \tilde{G}_s\left(\bar{u}; y_n, (y_i)_{i\le n}\right) - \tilde{G}_s\left(\bar{u}; y(t_n), (y_i)_{i\le n}\right) \right\} du \tag{5.17}$$

$$d_n := \frac{1}{s!N} \sum_{0\le j<N} \tilde{G}_s\left(\bar{t}_{j,n}; y_n, (y_i)_{i\le n}\right) - \frac{1}{s!h_n^s} \int_{(t_n,t_{n+1})^s} \tilde{G}_s\left(\bar{u}; y_n, (y_i)_{i\le n}\right) du \tag{5.18}$$

$$\eta_n := \frac{1}{s!h_n^s} \int_{(t_n,t_{n+1})^s} \left\{ \tilde{G}_s\left(\bar{u}; y(t_n), (y_i)_{i\le n}\right) - G_s\left(\bar{u}; y(t_n), y(t\le t_n)\right) \right\} du \tag{5.19}$$

$$\varepsilon_n := \frac{y(t_{n+1}) - y(t_n)}{h_n} - \frac{1}{s!h_n^s} \int_{(t_n,t_{n+1})^s} G_s\left(\bar{u}; y(t_n); y(t\le t_n)\right) du \tag{5.20}$$

Since the interpolation does not explicitly depend on the RKQMC scheme used, we can give a general bound for the interpolation error η_n:

Proposition 5.1. *If G_s is Lipschitz continuous in its second and third arguments (with Lipschitz constant $\mathcal{L}_2$), the interpolation is chosen such that the interpolation error is of order p, and the interpolation is Lipschitz continuous in the sense that $\tilde{G}_s$ fulfills*

$$\left\| \tilde{G}_s\left(t, y(t); (v_i, v_i')_{i\le n}\right) - \tilde{G}_s\left(t, y(t); (w_i, w_i')_{i\le n}\right) \right\| \le \tilde{\mathcal{L}}_2 \max_{i\le n} \|v_i - w_i\| \tag{5.21}$$

for $v_i, w_i \in B(y(t_i), \rho)$, $\rho > 0$ and $t \le t_i + h_i$ for all $i \le n$, then the interpolation error η_n is bounded by

$$\|\eta_n\| \le \frac{\mathcal{L}_2'}{s!} \left(\max_{j\le n} \|e_j\| + K h_n^p \right) \tag{5.22}$$

with constants $\mathcal{L}_2' = \max(\mathcal{L}_2, \tilde{\mathcal{L}}_2)$ and $K > 0$.

Remark 5.1. Without loss of generality, we do not need to include the values of the derivatives at the right hand side of assumption (5.21), since they are calculated by the delay differential equation itself. The Lipschitz-continuity of G_s in general demands the Lipschitz-continuity of $f(t, y(t), y(t-\tau(t)))$, so that the difference in the derivatives can be bounded by the difference of the function values using this Lipschitz-continuity.

Proof. Adding and subtracting the term $G_s(\bar{u}, y(t_n); z_{(y(t_i))_{i\le n}}(\cdot))$ to the integrand of the definition of η_n and using the Lipschitz conditions on G_s and $\tilde{G}_s$ yields the result

$$\begin{aligned} \|\eta_n\| \le \frac{1}{s!h_n^s} \Bigg(& \tilde{\mathcal{L}}_2 \max_{i\le n} \|y_i - y(t_i)\| \, h_n^s + \\ & + \mathcal{L}_2 \int_{(t_n,t_{n+1})^s} \max_{1\le j\le s} \left\| z_{(y(t_i))_{i\le n}}(u_j - \tau(u_j)) - y(u_j - \tau(u_j)) \right\| du \Bigg) \le \\ & \le \frac{\mathcal{L}_2'}{s!} \left(\max_{j\le n} \|e_n\| + K h^p \right) . \end{aligned}$$

□

Theorem 5.2 (Convergence of the RKQMC methods for DDE). *Let G_s be Lipschitz continuous in its second and third arguments with Lipschitz constant $\mathcal{L}_2$ and of bounded variation in the sense of Hardy and Krause. Let furthermore the RKQMC method be chosen such*

that for an order $p \geq 1$ there exist $c_1(h) = \mathcal{O}(1)$, $c_2(h) = \mathcal{O}(1)$ and $c_3(h) = \mathcal{O}(1)$ with

$$\|\varepsilon_n\| \leq c_1(h_n) h_n^p \tag{5.23}$$

$$\|\delta_n\| \leq c_2(h_n) \|e_n\| \tag{5.24}$$

$$\|d_n\| \leq c_3(h_n) D_N^*(S) \ , \tag{5.25}$$

where $S = (\mathbf{x}_1, \ldots, \mathbf{x}_N)$ is the point set used for the QMC integration. If the retarded values are interpolated by a q-order method, and the assumptions of Proposition 5.1 are fulfilled, then the error $\|e_{n+1}\|$ of the corresponding RKQMC method for delay differential equations is bounded by

$$\|e_n\| \leq \|e_0\| e^{t_n\left(c_2 + \frac{\mathcal{L}_2}{s!}\right)} + \frac{e^{t_n\left(c_2+\frac{\mathcal{L}_2}{s!}\right)} - 1}{c_2 + \frac{\mathcal{L}_2}{s!}} \left\{ c_3 D_N^*(X) + \frac{\mathcal{L}_2}{s!} M H^q + c_1 H^p \right\} \ , \tag{5.26}$$

where $H = \max_{0 \leq i \leq n-1} h_i$ and $c_i = \max_{0 \leq j \leq n-1} c_i(h_j) = c_i(H)$.

Remark 5.2. If the RKQMC method used is convergent for ordinary differential equations, the existence of $c_1(h_n)$, $c_2(h_n)$ and $c_3(h_n)$ is ensured since this is the most important part of the convergence proof for ODE. For this reason, we will not try and give the assumptions of theorem 5.2 in a more fundamental way. Inequality (5.23) states that the RK method is at least of order p and inequality (5.25) is basically a consequence of f and thus G_s being of bounded variation in the sense of Hardy and Krause, so that the Quasi-Monte Carlo integration error can be estimated by the famous Koksma-Hlawka inequality.

Proof of theorem 5.2: Combining the methods of [50] and [55], the error $e_n = y_n - y(t_n)$ of the scheme can be written as

$$\begin{aligned}
e_{n+1} = y_{n+1} - y(t_{n+1}) &= \{y_n - y(t_n)\} + \\
&+ \frac{h_n}{s!} \left\{ \frac{1}{N} \sum_{0 \leq j < N} \tilde{G}_s \left(\bar{t}_{j,n}; y_n, (y_i)_{i \leq n} \right) - \frac{1}{h_n^s} \int_{(t_n, t_{n+1})^s} \tilde{G}_s \left(\bar{u}; y_n, (y_i)_{i \leq n} \right) du \right\} + \\
&+ \frac{1}{s! h_n^{s-1}} \int_{(t_n, t_{n+1})^s} \left\{ \tilde{G}_s \left(\bar{u}; y_n, (y_i)_{i \leq n} \right) - \tilde{G}_s \left(\bar{u}; y(t_n), (y_i)_{i \leq n} \right) \right\} du + \\
&+ \frac{1}{s! h_n^{s-1}} \int_{(t_n, t_{n+1})^s} \left\{ \tilde{G}_s \left(\bar{u}; y(t_n), (y_i)_{i \leq n} \right) - G_s \left(\bar{u}; y(t_n), y(t \leq t_n) \right) \right\} du - \\
&- h_n \varepsilon_n \ .
\end{aligned}$$

And so

$$\begin{aligned}
\|e_{n+1}\| \leq \|e_{n+1}\|_{\text{bnd}} &:= \|e_n\| + h_n \|d_n\| + h_n \|\delta_n\| + h_n \|\eta_n\| + h_n \|\varepsilon_n\| \\
&\leq \|e_n\| + h_n c_3(h_n) D_N^*(X) + h_n c_2(h_n) \|e_n\| + c_1(h_n) h_n^{p+1} + \\
&\quad + h_n \frac{L_2}{s!} \left\{ M h_n^q + \max_{j \leq n} \|e_n\| \right\} \\
&\leq \left(1 + h_n \left(c_2 + \frac{\mathcal{L}_2}{s!} \right) \right) \|e_n\|_{\text{bnd}} + h_n \left(c_3 D_N^*(X) + \frac{\mathcal{L}_2}{s!} M H^q + c_1 H^p \right)
\end{aligned}$$

Recursively inserting this inequality for $\|e\|_{\text{bnd}}$ into itself, using the inequality $1 + h_n a \leq e^{h_n a}$ and a telescopic sum finally yield the result. □

Remark 5.3. One should note that in the proof given in [55], it is assumed that the approximation formula $z(t)$ to the retarded values is smooth, and so the set of support points for the interpolation can only be changed between two time steps. Since the QMC integration needs the calculation of a lot of points from a large interval, a significant amount of extrapolation would be involved. However, the assumptions of the RK(Q)MC methods by Stengle, Lécot, Coulibaly and Koudiraty, and the method presented here do not require the function $g(t, y(t))$ to be smooth in t any more, but demand only boundedness of g and its derivatives w.r.t. y for all t. This is still fulfilled if we choose the support points different for different values of t (but always choose the same support points in a finite neighborhood of t), thus avoiding unnecessary extrapolation.

5.3.4 Convergence of the RKQMC1, RKQMC2 and RKQMC3 methods

For the first- to third-order RKQMC methods, Lécot and Koudiraty proved the following lemmas under the assumptions that

1. there exist $t > 0, \rho > 0$ such that $D_y^m f$ is measurable for $0 \leq m \leq s$ on the set $E := \cup_{0 \leq t \leq T} [t, \min(t+\tau, T)] \times B(y(t), \rho)$ and, for fixed t, continuous in y on the open ball $B(y(t), \rho)$

2. and that for every $t \in [0, T]$ and every $y \in B(y(t), \rho)$ the m-th derivative $D_y^m f(u, y)$ is defined for $u \in [t, \min(t+\tau, t)]$ and bounded by $\left\| D_y^m f \right\|_E$ for $0 \leq m \leq s$, and its variation is bounded by $V_E(D_y^m f)$ for $0 \leq m \leq s-1$.

Lemma 5.3 (Lécot, [52]). *For the RKQMC1 method, if $h_n \leq \tau$ and $\|e_n\| + h_n \|f\|_{\infty,\varepsilon} < \rho$, then*

$$\|\varepsilon_n\| \leq \frac{h_n}{2} \|f\|_{\infty,\varepsilon} \left\| D_y^1 f \right\|_{\infty,\varepsilon} \tag{5.27}$$

$$\|d_n\| \leq \left\| D_y^1 f \right\|_{\infty,\varepsilon} \|e_n\| \tag{5.28}$$

$$\|\delta_n\| \leq V_\varepsilon(f) D_N^*(X) \tag{5.29}$$

Lemma 5.4 (Lécot, [52]). *If for the RKQMC2 method the additional conditions $h_n \leq \tau$ and $\|e_n\| + (1+\alpha) h_n \|f\|_{\infty,\varepsilon} < \rho$ hold, then with $D_i = \left\| D_y^i f \right\|_{\infty,\varepsilon}$, $i = 1, 2$, the following inequalities hold:*

$$\|\varepsilon_n\| \leq \frac{h_n^2}{12} \|f\|_{\infty,\varepsilon} \left((3\alpha^2 + 2) \|f\|_{\infty,\varepsilon} D_2 + 2 (D_1)^2 \right) = c_1^{(2)}(f) h_n^2 \tag{5.30}$$

$$\|d_n\| \leq \frac{1}{2} \left(1 + \frac{1}{|\alpha|} + \frac{1}{|\beta|} + h_n D_1 \right) D_1 \|e_n\| = c_2^{(2)}(f, h_n) \|e_n\| \tag{5.31}$$

$$\|\delta_n\| \leq \frac{1}{2} \left(1 + \frac{1}{|\alpha|} + \frac{1}{|\beta|} + h_n \left(D_1 + 2 V_\varepsilon(D_y^1 f) \right) \right) V_\varepsilon(f) D_N^*(X) = \tag{5.32}$$

$$= c_3^{(2)}(f, h_n) D_N^*(X) \tag{5.33}$$

Lemma 5.5 (Koudiraty, [50]). *If $h_n \leq \tau$ and $\|e_n\| + h_n c^* \|f\|_E < \rho$ with*

$$c^* := 1 + \max \left(\max_{1 \leq l \leq L_1} |b_{2,l}|, \max_{1 \leq l \leq L_3} \left| b_{3,l}^{(1)} \right| + \max_{1 \leq l \leq L_3} \left| b_{3,l}^{(2)} \right|, \max_{1 \leq l \leq L_3} |c_{3,l}| \right)$$

hold for the RKQMC3 method, then there exist $c_1(h_n) = \mathcal{O}(1)$, $c_2(h_n) = \mathcal{O}(1)$ *and* $c_3(h_n) = \mathcal{O}(1)$ *such that*

$$\|\varepsilon_n\| \leq c_1(h_n) h_n^3 \tag{5.34}$$

$$\|\delta_n\| \leq c_2(h_n) \|e_n\| \tag{5.35}$$

$$\|d_n\| \leq c_3(h_n) D_N^*(X) \tag{5.36}$$

Inserting these lemmas in our result from theorem 5.2, one obtains the convergence proofs for the RKQMC1, RKQMC2 and RKQMC3 methods for delay differential equations:

Corollary 5.6. *For the first order RKQMC1 method applied to delay differential equations, if the past values are interpolated in a way that the assumptions of Proposition 5.1 are fulfilled, and* $H \leq \tau$ *as well as*

$$e^{\left\|D_y^1 f\right\|_{\infty,\varepsilon} t_n} \|e_0\| + \frac{e^{Dt_n} - 1}{D} \left(\frac{1}{2} \|f\|_{\infty,\varepsilon} DH + V_\varepsilon(f) D_N^*(X) \right) + H \|f\|_{\infty,\varepsilon} < \rho$$

hold with $D = \left\|D_y^1 f\right\|_{\infty,\varepsilon}$, *then the method is convergent for* $h_n \to 0$, $D_N^*(X) \to 0$ *and* $\|e_0\| \to 0$, *and the error is bounded by*

$$\|e_n\| \leq \|e_0\| e^{(D+\mathcal{L}_2)t_n} + \frac{e^{(D+\mathcal{L}_2)t_n} - 1}{D + \mathcal{L}_2} \left(V_\varepsilon(f) D_N^*(X) + \mathcal{L}_2 M H^q + \frac{1}{2} \|f\|_{\infty,\varepsilon} DH \right) \tag{5.37}$$

Corollary 5.7. *For the second order RKQMC2 method for delay differential equations, the error under the assumptions of Lemma 5.4 and proposition 5.1 is bounded by* (5.26) *with* $c_1 := c_1^{(2)}(f)$, $c_2 := \max_{0 \leq j < n} c_2^{(2)}(f, h_n)$ *and* $c_3 := \max_{0 \leq j < n} c_3^{(2)}(f, h_n)$ *with the definitions from equations* (5.30) *to* (5.32), *and thus the method is convergent. Its order is* $\min(2, q)$, *if a low-discrepancy point set is used with* $N = \mathcal{O}\left(H^{-\min(2,q)}\right)$ *points.*

Corollary 5.8. *For the third order RKQMC3 method for delay differential equations, the error under the assumptions of Lemma 5.5 and proposition 5.1 is bounded by* (5.26) *with* $c_1^{(3)}(h_n)$, $c_2^{(3)}(h_n)$ *and* $c_3^{(3)}(h_n)$ *as defined in [50]. So the method is convergent with order* $\min(3, q)$, *if a low-discrepancy point set is used with* $N = \mathcal{O}\left(H^{-\min(3,q)}\right)$ *points.*

5.4 Computational experiments

We now investigate the effects of the use of Runge Kutta QMC methods compared to conventional Runge Kutta methods at the example of the delay differential equation

$$y'(t) = 3y(t-1)sin(\lambda t), \qquad \text{for } t \geq 0 \tag{5.38}$$

$$y(t) = 1, \qquad \text{for } t \leq 0 \tag{5.39}$$

with $\lambda = 2^\nu$ and $1 \leq \nu \leq 16$. The exact solutions for different values of ν is shown in figure 5.1. As one can see, the solutions oscillate heavily, although their amplitudes get smaller as ν grows. One also has to notice the kink at $t = 1$ for all solutions, which is the first discontinuity of $y'(t)$ due to the initial function not being a solution of the DDE.
In the sequel, if not mentioned explicitly, all calculations were done with a time step of $h_n = h = 0.001$ for values up to $T = 5$, and the past values are interpolated by a fourth order

Hermite interpolation. To compare the RKQMC methods with conventional Runge Kutta methods, we use the RKQMC methods presented in the previous chapter, as well as some low order Runge Kutta Methods (the 4-stage Runge method of order 3 and the 3-stage method of Heun of order 3). Additionally, we computed the results with Butcher's 6-th order method as an example of a high order Runge Kutta scheme. These schemes are discussed in [34] at great detail, so we will not even give the Butcher tableaus here.

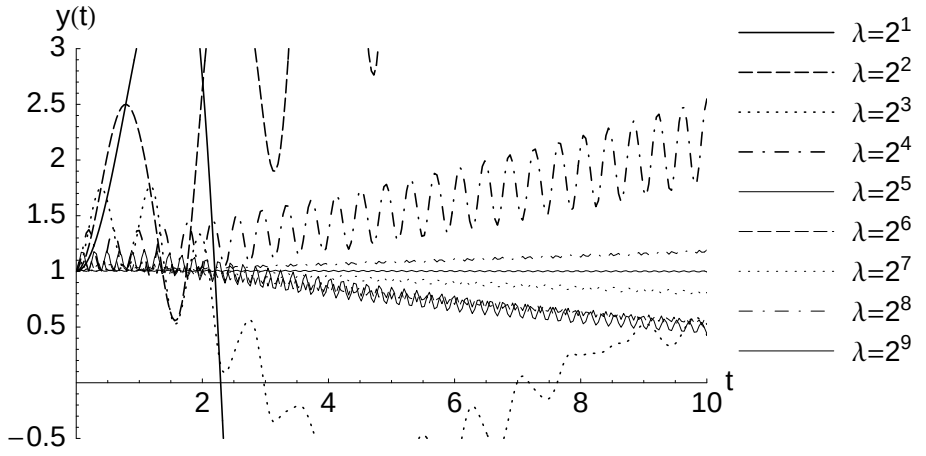

Figure 5.1: *The exact solution of* (5.38) *for some values of* λ.

To compare the various methods, we compute the average error

$$err = \frac{1}{K}\sum_{i=1}^{K} |y(t_{im}) - y_{im}|$$

of a method where K and m are chosen such that $t_{im} - t_{(i-1)m} = 0.1$ and $t_{Km} = T$.
We are interested in the convergence order of the RKQMC methods applied to delay differential equations. In theorem 5.2 and its corollaries 5.6, 5.7 and 5.8, the error bound depends on the discrepancy of the point set used for the simulation, as well as on the method itself, and on h_n. If we take $N = \mathcal{O}(h_n^i)$, $i = 1, 2, 3$, according to the corollaries we would expect a different convergence order for the RKQMC1, RKQMC2 and RKQMC3 method. For $\lambda = 2^5$, however, we do not observe any difference with this specific DDE, neither for different orders of N, nor for the different RKQMC schemes, as figure 5.2 shows (other delay differential equations show the expected behavior).
A very important question when dealing with QMC methods is how many points to use for the integration. Taking too many points does not hurt accuracy, but unnecessarily wastes computing time. On the other hand, if one takes N too small, the Quasi-Monte Carlo integration error will not become negligible and thus introduce a bias to the result. If we take a fixed $h_n = 0.001$, $\lambda = 2^5$ and just increase the number of points, figure 5.3 shows that values of N larger than about $2^9 = 512$, cannot noticeably improve the error any more, because then the Runge Kutta and the truncation error outweigh the Quasi-Monte Carlo error. One can also notice that with low-discrepancy sequences like Sobol's sequence, which we used for all our calculations, the curve is much smoother than with pseudo-random numbers as used in Monte Carlo methods. This is presumably caused by the good distribution properties of these sequences, and can improve the error considerably below the Monte Carlo error.

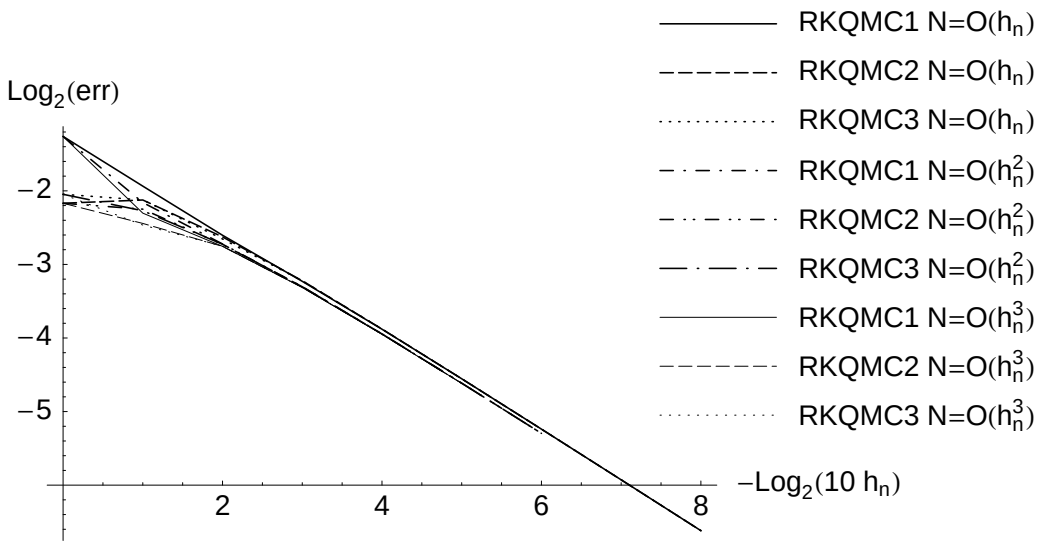

Figure 5.2: *Error for $N = \mathcal{O}(h_n)$, $N = \mathcal{O}(h_n^2)$ and $N = \mathcal{O}(h_n^3)$. ($\lambda = 32$)*

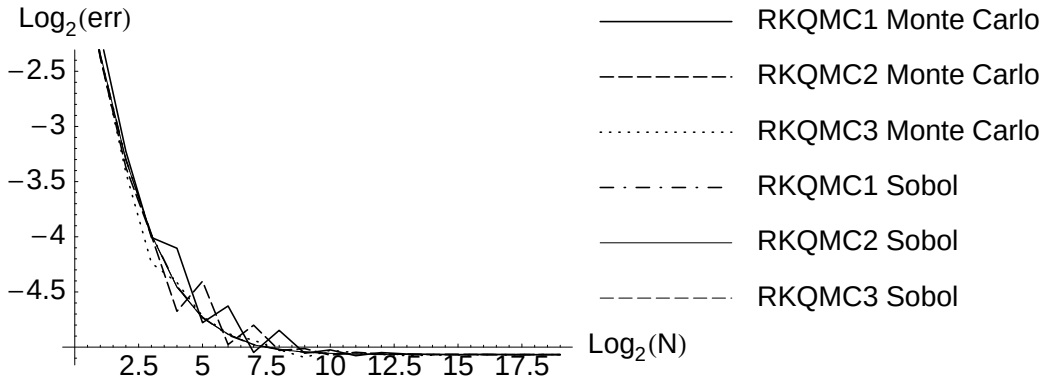

Figure 5.3: *With fixed h_n and increasing N, there is no visible difference in the convergence order ($\lambda = 32$).*

As Stengle [66] pointed out, the RK(Q)MC methods need a lot more evaluations of f, and thus can only claim advantage if these calculations can be carried out in parallel. Additionally, the RKQMC methods might even outperform conventional high order Runge-Kutta schemes, or calculations with a much smaller h_n. The results of Stengle, Lécot and Koudiraty show that their RK(Q)MC methods only gain advantage for differential equations where f varies significantly faster in t than in space. Figure 5.4 and table 5.1 show the error for the various methods for different values of ν in $\lambda = 2^\nu$. The time steps are $h_n = 0.001$ for the Butcher, Runge and Heun method, and $h_n = 0.01$ for the RKQMC methods. While for small values of λ the conventional Runge Kutta methods clearly give better results than the randomized Runge Kutta schemes, for high values of λ the situation changes, and the RKQMC schemes (which are low order methods) at least reach the same or a better average error than even the high order Butcher method.

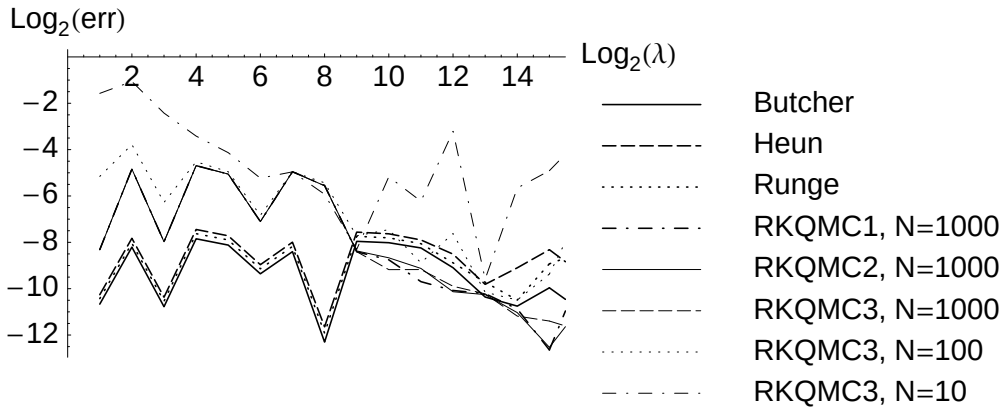

Figure 5.4: *For rapidly varying DDE ($\lambda \geq 2^{10}$), RKQMC methods outperform conventional high order Runge Kutta schemes.*

λ	Butcher	Runge	RKQMC1 $N = 1000$	RKQMC2 $N = 1000$	RKQMC3 $N = 1000$	RKQMC3 $N = 10$
2^1	−10.6663	−10.4277	−8.31395	−8.31394	−8.31389	−1.57259
2^2	−8.20716	−8.04143	−4.83919	−4.83909	−4.83908	−1.0597
2^3	−10.7767	−10.5397	−7.97728	−7.97651	−7.97628	−2.42371
2^4	−7.84711	−7.62639	−4.69098	−4.69087	−4.69086	−3.41549
2^5	−8.11869	−7.89766	−5.05594	−5.05595	−5.05624	−4.13132
2^6	−9.35768	−9.14174	−7.10261	−7.10411	−7.10611	−5.23092
2^7	−8.40319	−8.17748	−4.9544	−4.95436	−4.95707	−4.95889
2^8	−12.303	−11.9188	−5.54122	−5.55782	−5.55972	−5.92666
2^9	−7.95694	−7.73868	−8.4076	−8.37118	−8.40286	−8.30661
2^{10}	−8.0165	−7.80652	−8.75515	−8.66781	−9.17772	−5.1714
2^{11}	−8.24265	−8.05737	−9.69635	−9.12556	−9.17449	−6.22673
2^{12}	−9.09544	−8.88462	−10.0774	−10.1313	−9.89475	−3.20923
2^{13}	−10.3594	−9.72112	−10.2139	−10.2555	−10.2228	−9.58793
2^{14}	−10.7523	−10.4912	−10.8732	−11.0333	−11.1829	−5.65672
2^{15}	−9.96253	−8.92665	−12.6424	−12.5296	−11.396	−4.90893
2^{16}	−10.956	−8.4328	−9.28126	−10.7504	−11.8211	−3.4884
2^{17}	−8.85172	−9.77356	−9.94702	−10.4492	−9.48664	−5.50948
2^{18}	−8.80146	−8.58722	−11.3812	−10.3786	−10.1079	−6.2911

Table 5.1: *Error for increasing parameter values λ ($h_n = 0.001$ for Butcher, Runge and Heun, and $h_n = 0.01$ for RKQMC)*

5.5 Conclusion

In this chapter we showed that the randomized Runge Kutta schemes as proposed by Stengle, Lécot, Coulibaly and Koudiraty can be successfully extended to delay differential equations, and for rapidly varying differential equations even the low order RKQMC methods lead to an improvement over conventional (high- and low-order) Runge Kutta schemes. Although the RKQMC methods need several times more function evaluations than its Runge Kutta counterparts of the same order, the possibility of parallel computing and the fact that the low order RKQMC methods even outperform high order Runge Kutta schemes for certain types of delay differential equations, make the RKQMC methods for delay differential equations a viable solution method.

Chapter 6

Multiply delayed differential equations

Contents

Starting from the results of the previous chapter, we will now extend the RKQMC methods for DDE with one retarded argument to delay differential equations with an arbitrary number of such delayed arguments. The restriction to a finite number exists only for practical reasons (to ensure an implementation on a computer is possible). The convergence proof will use an arbitrary Volterra functional equation, only with certain conditions to ensure the convergence. Finally, an extensive numerical investigation is carried out.
The results presented in this chapter are a joint work with R. Tichy and were published in [47] and [48].

6.1 Introduction

When dealing with real-world problems, the change of a process $y(t)$ and thus the derivative $y'(t)$ in its mathematical representation often not only depend on the value of the process at present, but also on the past values. The differential equations describing such processes are usually called delay or retarded differential equation, since they also involve terms of the form $y(t-\tau(t))$, where $\tau(t)$ is a function with positive function values, the simplest being a constant retardation $y(t-\tau)$ with $\tau > 0$. In the previous chapter we looked at a class of delay differential equations with one retarded argument of the form

$$y'(t) = f\left(t, y(t), y(t-\tau(t))\right), \quad \text{for } t \geq t_0, \tag{6.1}$$

where the solution $y(t)$ is a d-dimensional real-valued function, $\tau(t)$ is the continuous delay function, which was assumed to be bounded from below by $\tau_0 > 0$). Furthermore, $\phi(t)$ is the

initial function, which is piecewise continuous at least on the interval $(\inf_{t_0 \le t}(t - \tau(t)), t_0)$. The analysis there was restricted to the case when $\tau(t)$ fulfills the condition $t_1 - \tau(t_1) \le t_2 - \tau(t_2)$ for $t_1 \le t_2$.

6.2 Description of the problem

In this chapter we will consider Quasi-Monte Carlo Runge Kutta methods for a generalization of the delay differential equation (6.1) to an arbitrary (but finite) number of retarded arguments:

$$\begin{aligned} y'(t) &= f\left(t, y(t), y(t-\tau_1(t)), \ldots, y(t-\tau_k(t))\right), \quad \text{for } t \ge t_0,\ k \ge 1, \\ y(t) &= \phi(t), \ \text{for } t \le t_0\ , \end{aligned} \tag{6.2}$$

where the solution $y(t)$ is a d-dimensional real-valued function, $\tau_1(t), \ldots, \tau_k(t)$ are continuous delay functions, which we assume to be bounded from below by $\tau_0 > 0$. Furthermore, $\phi(t)$ is the initial function, which shall be continuous at least on the interval $[\inf_{t_0 \le t}(t - \tau(t)), t_0]$. Like in [46], we will sequentially calculate the approximated function values y_n at times $t_n = t_0 + \sum_{j=0}^{n} h_j$ for time steps h_n. To obtain the values of the retarded arguments $y(t-\tau_j(t))$ from the sequence (y_n) we will use Hermite interpolation, because the numerical value of the derivative $y'(t_j)$ at time t_j is known from the differential equation. The main advantage of Hermite interpolation over interpolation methods using just the function values is that it only needs half the points to obtain a given interpolation order.
For f continuous in t and Lipschitz in the other variables, Driver [23] gave a local existence and uniqueness theorem for a very general Volterra functional delay differential equation, which contains (6.2) as a special case. In this case, the existence theorem reads:

Lemma 6.1 (local existence and uniqueness, Driver [23]). *Let f be (i) continuous in t and (ii) locally Lipschitz in the other arguments $y(t)$ and $y(t-\tau_k(t))$, and the initial function $\phi(t)$ continuous on $[\alpha, t_0]$, where $\alpha = \inf_{\substack{t \ge t_0 \\ 1 \le j \le k}} (t - \tau_j(t))$. Then for sufficiently small $h > 0$ there exists a unique solution $y(t; t_0, \phi)$ to the differential equation (6.2) for $\alpha \le t < t_0 + h$.*

Remark 6.1. One should notice that while the RKQMC methods might still be applicable, if f is not continuous but only bounded and measurable in t, this theorem cannot be applied. However, under hypotheses similar to the ones Stengle assumes in [66, Hypothesis 2.1], Picard iteration can be applied to assure local existence and uniqueness if f is only bounded and measurable in t.
If f is not globally continuous, however, Hermite interpolation and thus RKQMC methods for DDE can only be applied, if the solution y is at least piecewise continuous on intervals which contain $p/2$ or more support points $t_j, \ldots, t_{j+p/2}$ so that the interpolation error can be bounded using the interpolation order.

However, even smoothness of f and ϕ does not guarantee smoothness of the solution $y(t)$, because similar to delay differential equations with one retarded argument (see [55]) the solution in general will have discontinuous first derivatives at times $t_j^{(1)}$ which are defined as the solutions of the equations $t_j^{(1)} - \tau_j(t_j^{(1)}) = 0$ for $1 \le j \le k$, discontinuous second derivatives at times $t_{j,l}^{(2)}$ with $t_{j,l}^{(2)} - \tau_l(t_{j,l}^{(2)}) = t_j^{(1)}$ for $1 \le j, l \le k$, and so on. On the intervals between them, a calculation of the exact solution is possible in theory by inserting the already

calculated solution from the previous intervals to get rid of the retarded arguments. Thus the calculation of an exact solution means sequentially solving ordinary differential equations on each of the intervals, which quickly becomes an analytically intractable problem.

Here, however, we will not be concerned with exact solutions, but with its numerical approximation by means of (quasi-)randomized Runge Kutta methods. We will first present our RKQMC method, which combines the RKQMC methods by Stengle, Lécot, Koudiraty and Coulibaly with the Hermite interpolation method for delay differential equations. We will give a short convergence proof for arbitrary RKQMC schemes applied to delay differential equations of multiple retarded arguments under certain technical conditions on the delay differential equation, the interpolation and the specific RKQMC scheme taken from ordinary differential equations. A convergence proof for delay differential equations with only one retarded argument was already given by the authors in [46]. The biggest part of the chapter will be dedicated to an extensive numerical investigation of the RKQMC methods for delay differential equations.

6.3 The RKQMC method for differential equations with multiple retarded arguments

Following the path of Stengle [66] as well as Lécot, Coulibaly and Koudiraty ([52, 17, 50], we will now deduce Quasi-Monte Carlo schemes for delay differential equations with multiple retarded arguments. These schemes are akin to the family of Runge-Kutta schemes and thus they are often called Runge Kutta (Quasi-) Monte Carlo schemes (RKQMC in short).
In [50] Lécot and Koudiraty treated the case of ordinary differential equations and derived a quasi-randomized Runge Kutta scheme. If we assume that we already know the exact or approximated solution $\hat{y}$ to the equation up to time t (or at least up to $\max_{1\leq i\leq k}(t-\tau_k(t))$), this solution can be inserted into the right hand side of equation (6.2), and the equation simplifies to an ordinary differential equation at time t with $g(t,y(t)) := f(t,y(t),y(t-\tau_1(t)),\dots,y(t-\tau_k(t))$. The value of the solution at this time can now simply be calculated by such a Runge Kutta QMC scheme. For this reason, in the following paragraphs we will repeat Koudiraty's arguments which lead to RKQMC methods for ordinary differential equations. Throughout the whole argumentation, the existence and uniqueness of the solution $y(t)$ with $y' \in L^1(0,T)$ needs to be assumed, which in our case is guaranteed by the theorem and the remarks of the previous section.
Starting from the differential equation $y'(t) = g(t,y(t),y(t-\tau_1(t)),\dots)$ or its equivalent integral representation

$$y(t_0+h) = y(t_0) + \int_{t_0}^{t_0+h} g(u,y(u),y(t-\tau_1(t)),\dots)du$$

the retarded values $y(t-\tau_1(t)),\dots,y(t-\tau_n(t))$ are approximated by ordinary Hermite interpolation from the already calculated values at the discrete times $t_n \leq t$. This is done using an interpolation function

$$\tilde{z}(t) = z_{(y_i)_{i\leq n}}(t) = \begin{cases} \phi(t), & \text{if } t \leq t_0 \\ P_q\left(t;(y_i),(y_i')\right) & \text{otherwise} \end{cases} \tag{6.3}$$

which uses the initial function $\phi(t)$ for all values prior to the starting value. For later times the function $z(t)$ is constructed piecewise by Hermite-interpolation where appropriate grid points t_n are used as support points and the value of $y'(t_n)$ is approximated by $f(t_n, y(t_n), y(t_n - \tau_1(t_n)), \ldots, y(t_n - \tau_n(t_n)))$. The support points to construct the Hermite polynomial for a certain value of t are chosen such that extrapolation is avoided, and all support points lie inside the same interval of smoothness as the time t. Naturally, it is of advantage to use adjacent grid points with the value of t in between, so that the interpolation error is kept to a minimum. Apart from this incentive, the method does not put any further restrictions on the choice, except that the resulting function $g\,(t, y(t); z(t - \tau_1(t)), \ldots, z(t - \tau_n(t)))$ needs to be of bounded variation in t to be able to apply the RKQMC methods at all. In contrast to previous works (e.g. Oberle and Pesch [55]), smoothness in the interpolation function is thus not required, which would mean that only one interpolation polynomial has to be used for all interpolation values in one time step. Since our method involves integration over a whole interval, this would involve an extensive amount of extrapolation, so the possibility to chose the interpolation polynomial freely according to only the value of t is of vital importance here. Inserting the interpolation function $z(t)$ into the delay differential equation transforms it into an ordinary differential equation, or an ordinary integral equation, respectively:

$$y(t_0 + h) = y(t_0) + \int_{t_0}^{t_0+h} g(u, y(u), z(u - \tau_1(u)), \ldots)du \tag{6.4}$$

Starting from equation (6.4), the function g is Taylor-expanded up to order s with respect to $y(t)$ and recursively inserted into itself. Since this requires g to be s-times differentiable in y, the function f needs to be s-times differentiable in y. According to the discussion above for smooth f and ϕ the solution $y(t)$ is also smooth on the interval $[t_0, \min_{i\leq k} \tau_i)$. Note that g does not have to be Taylor-expanded in the retarded arguments $y(t - \tau_1(t), y(t - \tau_2(t)), \ldots,$ because their values are already calculated via Hermite interpolation.
Using the general identity

$$\left(\int_{t_0}^{u_i} f(u)du\right)^n = n! \int_{t_0}^{u_i} \cdots \int_{t_0}^{u_{i+n}} f(u_{i+n-1}) \cdot \cdots \cdot f(u_{i+n})du_{i+n}du_{i+1}$$

this gives [50]

$$\begin{aligned} y(t_0 + h) = y(t_0) &+ \int_{t_0}^{t_0+h} F_1(u_1; y)du_1 + \int_{t_0}^{t_0+h} \int_{t_0}^{u_1} F_2(u_1, u_2; y)du_2du_1 + \ldots \\ &+ \int_{t_0}^{t_0+h} \int_{t_0}^{u_1} \int_{t_0}^{u_{s-1}} F_s(u_1, \ldots, u_s; y)du_s \ldots du_1 \\ &=: \frac{1}{s!h^{s-1}} \int_{t_0}^{t_0+h} \int_{t_0}^{t_0+h} G_s(\bar{u}_1, \ldots \bar{u}_s; y)du_s \ldots du_1 \end{aligned}$$

with

$$F_i(u_1, \ldots, u_i; y) := D_y^1 F_{i-1}(u_1, \ldots, u_{i-1}; y) f(u_i; y(u_i), y(u_i - \tau_1(u_i)), \ldots)$$

and $F_0(y) := y$. Here $\bar{u}$ denotes the vector u with the components sorted in ascending order. The function G_s still contains derivatives of $f(t; y, y(t - \tau_1(t)), ...)$ with respect to $y(t)$. Using a specific identity like the ones proposed by Stengle [66] or Lécot and Koudiraty [50] to

approximate linear combinations of derivatives of f by Runge-Kutta-like schemes of order s, one can obtain particular increment functions

$$\tilde{G}_s(\bar{u}_1, \dots \bar{u}_s; y(t), y(t-\tau_1(t)), \dots, y(t-\tau_n(t)))$$

which approximate the exact increment function to order h^{s+1}:

$$\tilde{G}_s = G_s(\bar{u}_1, \dots \bar{u}_s; y(t), y(t-\tau_1(t)), \dots, y(t-\tau_n(t))) + \mathcal{O}(h^{s+1}) \; .$$

As a final step to obtain the method, the remaining integral over $[t_0, t_0+h)^s$ is calculated by Quasi-Monte Carlo integration

$$\int_{t_0}^{t_0+h} \int_{t_0}^{t_0+h} G_s(\bar{u}_1, \dots \bar{u}_s; y) \approx \frac{1}{N} \sum_{i=1}^{N} G_s\left(t^{(\bar{n})}; y\right) \tag{6.5}$$

Stengle [65, 66] proposed a scheme of second order which uses pseudo-random numbers, while the schemes proposed by Lécot (first and second order, [52]), Coulibaly and Lécot (second order, [17]) and Lécot and Koudiraty (third order, [50]) use quasi-random numbers and thus absolute upper error bound were given for these schemes.
In the calculations presented in this chapter, we will use these existing first, second and third order RKQMC schemes for the increment function $G_s(t, y(t), y(t-\tau_1(t)), \dots, y(t-\tau_n(t))$ and not derive our own schemes.

6.4 Convergence of the method

Before we can prove convergence of the method, we have to define it in a more rigorous manner. For brevity, we will suppress the argument t of the function y whenever it is clear what the argument is.
We first rewrite the delay differential equation (6.2) in a more Volterra functional equation-like form

$$\begin{aligned} y'(t) &= f(t, y(t), z(t)) && t \geq t_0 \\ z(t) &= (Fy)(t) \\ y(s) &= g(s) && s \leq t_0 \end{aligned} \tag{6.6}$$

where all dependence on retarded values is moved into $(Fy)(t)$. We will furthermore assume that both F and f are Lipschitz continuous in all arguments except t.
Our method, which will generate a sequence (y_n) of values that approximate the exact solution $y(t)$ at the grid points $t_0 \leq t_1 \leq t_2 \leq \dots \leq t_m$, then reads

$$\begin{aligned} y_{n+1} &= y_n + h_n \sum_{i=1}^{N} G_s\left(t_{n,i}; y_n, \tilde{z}(t)\right) \\ \tilde{z}(t) &= (\tilde{F} y_j)(t) \end{aligned} \tag{6.7}$$

for $n \geq 0$, and $y_j = \phi(t_j)$ for $j < 0$. The function $(\tilde{F} y_j)$ uses the interpolation function (6.3) for the retarded values of y_j.
Our goal is now to establish upper bounds for the error $\|e_{j+1}\| = \|y_{j+1} - y(t_{j+1})\|$. We will henceforth only use RKQMC methods which converge with order $\mathcal{O}(h^p)$ for ordinary differential equations. Following the idea of Oppelstrup [56] for the solution of DDE using Hermite interpolation, the following theorem proves convergence of our method for DDE:

Theorem 6.2. *Let $y(t)$ be the solution of the DDE (6.6), and $(y_j)_{0\le j\le n}$ the approximate solution obtained by (6.7) on the grid $t_0 \le t_1 \le \cdots \le t_n$. Let furthermore $\left\|E_G^{ODE}\right\|_j$ be the error in the j-th step of the RKQMC method applied to the ODE which uses the exact solution for the retarded values, and $\left\|E_G^{ODE}\right\| := \max_j \left\|E_G^{ODE}\right\|_j$. The conventional Hermite interpolation error $\left\|E_r^{interpol}\right\|_j$ is defined as $\max_{t_j\le u\le t_{j+1}}(Fy(t))(u) - (\tilde{F}y(t))(u)$, and $\left\|E_r^{interpol}\right\| := \max_{0\le j\le n}\left\|E_r^{interpol}\right\|_j$. If*

1. *the RKQMC method with increment function $G_s(t_n, y_n)$ converges with order p for ODE as $h_j \to 0$,*
2. *G_s as defined in (6.7) is Lipschitz in $\tilde{z}$ with constant $\tilde{\mathcal{L}} > 0$, and F is Lipschitz with constant $\mathcal{L}_2 > 0$,*
3. *$\tilde{F}$ fulfills a Lipschitz criterion $\left\|(\tilde{F}u_j)(t) - (\tilde{F}v_j)(t)\right\| \le \tilde{\mathcal{L}} \max\|u_k - v_k\|$ with Lipschitz constant $\tilde{\mathcal{L}} > 0$,*
4. *the Hermite interpolation has order r,*
5. *and the initial error $\|e_0\|$ vanishes,*

then the method converges, and the error is bounded by

$$\|e_{j+1}\| \le \|e_0\| e^{\mathcal{L}t_j} + \frac{\left(e^{t_j\mathcal{L}} - 1\right)}{\mathcal{L}} \left(\left\|E_G^{ODE}\right\| + \mathcal{L}_1\mathcal{L}_2 \left\|E_r^{interpol}\right\|\right)$$

Remark 6.2. Like in the case of one retarded argument [46], this theorem shows that the RKQMC1, RKQMC2 and RKQMC3 methods by Lécot, Coulibaly and Koudiraty ([52, 17, 50]) can be successfully applied to DDE. We refrain from giving explicit upper bounds for the error, since these are usually very generous and several orders of magnitude (up to 10 orders) above the numerically experienced errors. For this reason, error bounds as obtained above are useful to prove convergence, but not to give a good error estimate.

Remark 6.3. Condition 4 puts an effective upper bound on the step size, namely that $\frac{r}{2}$ Points $t_k, ..., t_{k+\frac{r}{2}}$ need to be inside an interval of smoothness as discussed in chapter 6.2.

Proof of theorem 6.2. Using the definition of our method, we get for the error $\|e_{j+1}\|$:

$$\|e_{j+1}\| = \left\| e_j + h_h \left(\hat{G}(t_j, y(\cdot)) - \sum_{i=1}^{N} G_s(t_{j,i}; y_j, \tilde{z}(t_{j,i})) \right) \right\| ,$$

where $\hat{G}(t_j, y(\cdot)) = \frac{1}{h_j}\int_{t_j}^{t_j+h_j} g\left(u, y(u), y(u-\tau_1(u)), \ldots y(u-\tau_n(u))\right) du$ is the exact increment function using the solution $y(\cdot)$ for the retarded values. This leads to the further estimate

$$\begin{aligned}\|e_{j+1}\| &\le \|e_j\| + h_j \left\| \hat{G}(t_j, y(\cdot)) - \frac{1}{N}\sum_{i=1}^{N} G_s(t_{j,i}; y(t_j), z(\cdot)) \right\| + \\ &+ h_j \left\| \frac{1}{N}\sum_{i=1}^{N} G_s(t_{j,i}; y(t_j), z(\cdot)) - \frac{1}{N}\sum_{i=1}^{N} G_s(t_{j,i}; y_j, z(\cdot)) \right\| + \\ &+ h_j \left\| \frac{1}{N}\sum_{i=1}^{N} G_s(t_{j,i}; y_j, z(\cdot)) - \frac{1}{N}\sum_{i=1}^{N} G_s(t_{j,i}; y_j, \tilde{z}(\cdot)) \right\| \\ &\le \|e_j\| + h_j \left\| E_G^{\mathrm{ODE}} \right\|_j + h_j \frac{1}{N}\sum_{i=1}^{N} \|G_s(t_{j,i}, y_j, y(\cdot)) - G_s(t_{j,i}, y_j, \tilde{z}(\cdot))\|\end{aligned}$$

where $\left\|E_G^{\mathrm{ODE}}\right\|_j$ denotes the error in the j-th step of the RKQMC method applied to the ODE using the exact solution for the retarded arguments. The error due to the interpolation can furthermore be estimated by

$$\begin{aligned}&\|G_s(t_{j,i}, y_j, z(\cdot)) - G_s(t_{j,i}, y_j, \tilde{z}(t_j))\| \le \\ &\le \left\|G_s(t_{j,i}; y_j, z(\cdot)) - G_s(t_{j,i}; y_j, \tilde{z}_{(y(t_i))_{i\le j}}(\cdot))\right\| + \left\|G_s(t_{j,i}; y_j, \tilde{z}_{(y(t_i))_{i\le j}}(\cdot)) - G_s(t_{j,i}, y_j, \tilde{z}(\cdot))\right\| \\ &\le \mathcal{L}_1\mathcal{L}_2 \left(\left\| y(\cdot) - \tilde{z}_{(y(t_i))_{i\le j}}(\cdot) \right\| + \left\| \tilde{z}_{(y(t_i))_{i\le j}}(\cdot) - \tilde{z}(\cdot) \right\| \right) \\ &\le \mathcal{L}_1\mathcal{L}_2 \left(\left\| E_r^{\mathrm{interpol}} \right\|_j + \tilde{\mathcal{L}} \max_{0\le k\le j} \|e_k\| \right)\end{aligned}$$

using the Lipschitz conditions on the third argument of G_s with constant $\mathcal{L}_1$ and on F with constant $\mathcal{L}_2$.
Since the $\|e_i\|$ are monotone increasing, $\max_{0\le k\le j}\|e_k\| = \|e_j\|$. If we now put everything together, we get

$$\begin{aligned}\|e_{j+1}\| &\le \|e_j\| + h_j \left\| E_G^{\mathrm{ODE}} \right\|_j + h_j\mathcal{L}_1\mathcal{L}_2 \left(\left\| E_r^{\mathrm{interpol}} \right\|_j + \tilde{\mathcal{L}} \|e_j\| \right) \\ &\le \|e_j\| \left(1 + \mathcal{L}_1\mathcal{L}_2\tilde{\mathcal{L}}h_j\right) + h_j \left(\left\| E_G^{\mathrm{ODE}} \right\| + \mathcal{L}_1\mathcal{L}_2 \left\| E_r^{\mathrm{interpol}} \right\| \right) \\ &=: \|e_j\| (1 + \mathcal{L}h_j) + h_j\mathcal{C} \\ &\le \|e_0\| \prod_{k=0}^{j} (1 + h_k\mathcal{L}) + \mathcal{C}\sum_{n=0}^{j} h_n \prod_{k=n+1}^{j} (1 + h_k\mathcal{L}) \qquad (6.8) \\ &\le \|e_0\| \, e^{\mathcal{L}\sum_{k=0}^{j} h_k} + \mathcal{C}\sum_{n=0}^{j} \left(\frac{1 + h_n\mathcal{L}}{\mathcal{L}} \prod_{k=n+1}^{j} (1 + h_n\mathcal{L}) - \frac{1}{\mathcal{L}} \prod_{k=n+1}^{j} (1 + h_n\mathcal{L}) \right) = \\ &= \|e_0\| \, e^{\mathcal{L}t_j} + \frac{\mathcal{C}}{\mathcal{L}} \left(\prod_{k=0}^{j} (1 + h_n\mathcal{L}) - 1 \right) \le \|e_0\| \, e^{\mathcal{L}t_j} + \frac{\mathcal{C}}{\mathcal{L}} \left(e^{t_j\mathcal{L}} - 1 \right) \\ &= \|e_0\| \, e^{\mathcal{L}_1\mathcal{L}_2\tilde{\mathcal{L}}t_j} + \frac{\left(e^{t_j\mathcal{L}_1\mathcal{L}_2\tilde{\mathcal{L}}} - 1 \right)}{\mathcal{L}_1\mathcal{L}_2\tilde{\mathcal{L}}} \left(\left\| E_G^{\mathrm{ODE}} \right\| + \mathcal{L}_1\mathcal{L}_2 \left\| E_r^{\mathrm{interpol}} \right\| \right) \qquad (6.9)\end{aligned}$$

where in (6.8) the inequality was recursively inserted into itself, and an empty product should be understood as 1.
From the last inequality (6.9) the convergence of the method is obvious, and it follows that for a vanishing initial error $\|e_0\|$ the convergence of the RKQMC method for DDE is $\min\{p, r\}$,

$$
\begin{array}{c|ccc}
0 & & & \\
\frac{1}{3} & \frac{1}{3} & & \\
\frac{2}{3} & 0 & \frac{2}{3} & \\
\hline
 & \frac{1}{4} & 0 & \frac{3}{4}
\end{array}
\qquad
\begin{array}{c|cccc}
0 & & & & \\
\frac{1}{2} & \frac{1}{2} & & & \\
1 & 0 & 1 & & \\
1 & 0 & 0 & 1 & \\
\hline
 & \frac{1}{6} & \frac{2}{3} & 0 & \frac{1}{6}
\end{array}
$$

$$
\begin{array}{c|ccccccc}
0 & & & & & & & \\
\frac{1}{2} & \frac{1}{2} & & & & & & \\
\frac{2}{3} & \frac{2}{9} & \frac{4}{9} & & & & & \\
\frac{1}{3} & \frac{7}{36} & \frac{2}{9} & -\frac{1}{12} & & & & \\
\frac{5}{6} & -\frac{35}{144} & -\frac{55}{36} & \frac{35}{48} & \frac{15}{8} & & & \\
\frac{1}{6} & -\frac{1}{360} & -\frac{11}{36} & -\frac{1}{8} & \frac{1}{2} & \frac{1}{10} & & \\
1 & -\frac{41}{260} & \frac{22}{13} & \frac{43}{156} & -\frac{118}{39} & \frac{32}{195} & \frac{80}{39} & \\
\hline
 & \frac{13}{200} & 0 & \frac{11}{40} & \frac{11}{40} & \frac{4}{25} & \frac{4}{25} & \frac{13}{200}
\end{array}
$$

Table 6.1: *Butcher tableaus (taken from [34]) for Heun's (third order, 3-step) and Runge's (third order, 4-step) classical Runge-Kutta schemes, as well as for Butcher's high-order scheme (6-th order, 7 step)*

where p is the convergence order of the RKQMC method for ODE, and r is the order of the interpolation error. □

6.5 Numerical experiments

The idea of using Quasi-Monte Carlo integration for the integration over t stems from the attempt to minimize the error for heavily oscillating differential equations (or delay differential equations with heavily oscillating solutions, which results in a heavily oscillating ODE after inserting the solution $y(t)$ for the retarded values). Instead of accepting the (possibly large) error at just one retarded value, the dependence on t is averaged out, and the error thus minimized. Since RKQMC methods do not exhibit any advantage over conventional methods for non-oscillating equations, we will only investigate several heavily oscillating differential equations.

In our examples we will compare the first, second and third order RKQMC methods with some conventional Runge Kutta schemes with Hermite interpolation as proposed for example by Oppelstrup [56] or Oberle and Pesch [55]. In particular, as low-order Runge-Kutta schemes we applied the well-known 4-stage Runge scheme of order 3 and the 3-stage method of Heun of order 3, while as a high-order Runge-Kutta scheme we use Butcher's 6-th order, 7-stage method as described in [34]. The Butcher tableaus of these classical Runge Kutta schemes are given in Figure 6.1 for reference.

As a measure of quality of the obtained numerical solutions we will use the sum of deviations from the exact solution at the times $t = 0 \ldots 0.1 \ldots 20$, i.e.

$$S^{\lambda,(\mathrm{meth})} := \sum_{i=1}^{200} \left| y\left(\frac{i}{10}\right) - y_{\frac{i}{10}}^{(\mathrm{meth}),\lambda} \right| , \tag{6.10}$$

and all graphical comparisons on the dependence on λ will be given in terms of least-squares fits of the functions $\{1, x, x^2, \log(x)\}$ to the Functions $\log\left(S^{(\mathrm{meth}),\lambda}\right)$. As we mentioned above,

the RKQMC methods use a sum of N function values of f in every time step to approximate the integral in (6.5), so a lot more calculation time is needed for a single RKQMC step than for a conventional Runge-Kutta step. To include this effect into our evaluation, we will also compare the methods using the least-squares fits to the logarithm of the timed error S_T, which we define as

$$S_T^{\lambda,(\text{meth})} := T^{\lambda,(\text{meth})} S^{\lambda,(\text{meth})} , \tag{6.11}$$

with $T^{\lambda,(\text{meth})}$ being the calculation time for the given method and value of λ, so that two methods, where one needs only half the time, but twice the error of the other, are treated alike.

Unless we mention it explicitly, all values are obtained with a constant time step of $h_n = h = 0.001$ for classical Runge Kutta schemes, and a step of $h = 0.01$ for RKQMC schemes. The simulation is done up to time $T = 20$, and a forth order Hermite interpolation is used for the retarded values $y(t - \tau_k(t))$. As low-discrepancy sequence for the numerical integration we use Sobol's sequence, where the N numbers for each time step are taken sequentially, i.e. we use the first Nn elements of the sequence to obtain the solution value at time t_n.

As a first example of a delay differential equation with two retarded arguments, we applied our RKQMC methods to the delay differential equation

$$\begin{aligned} y'(t) &= 3y(t-1)\sin(\lambda t) + 2y(t-1.5)\cos(\lambda t), \quad t \geq 0 \\ y(t) &= 1, \quad t \leq 0 , \end{aligned} \tag{6.12}$$

which is very similar in its structure to the DDE we investigated in [46]. Its "exact" solutions, which we obtained by a run of Butcher's high order method using a step size of $h_n = 0.00001$, are shown in figure 6.1.

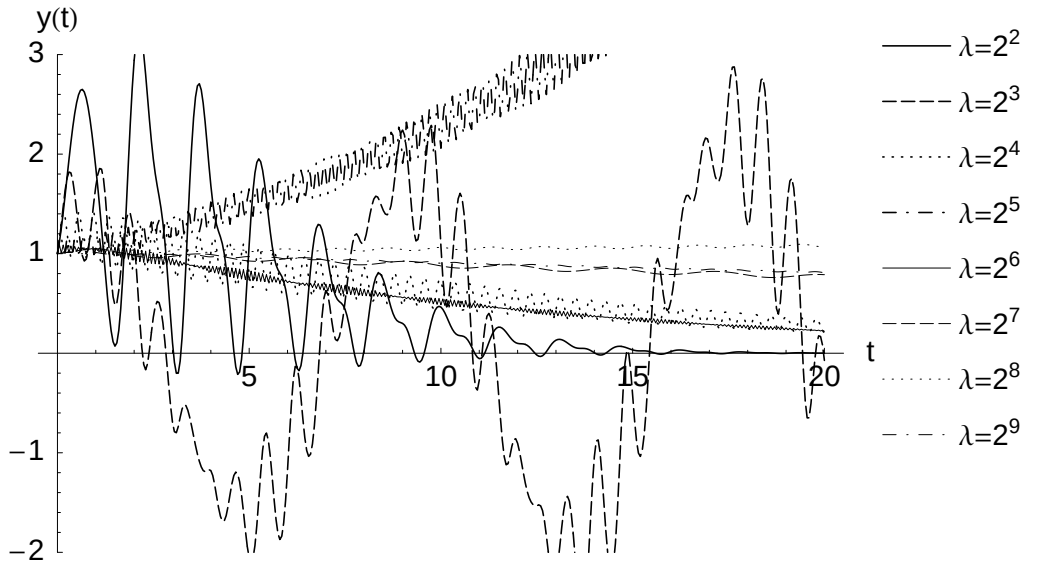

Figure 6.1: *Exact solution of example* (6.12) *for some values of* λ *obtained by a run with very small step size.*

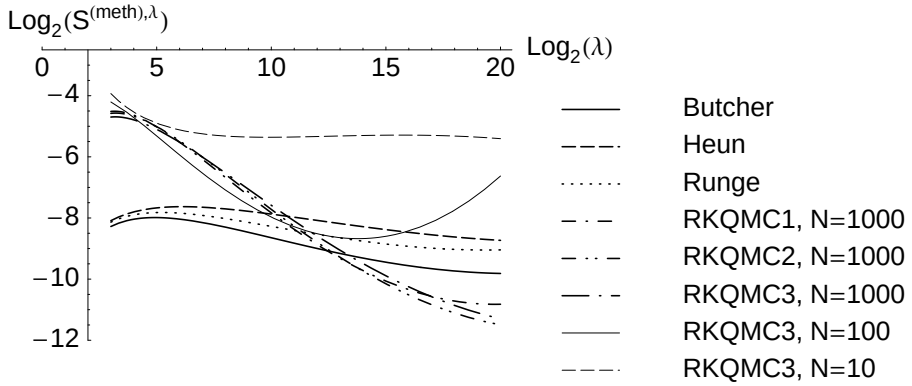

Figure 6.2: *RKQMC vs. conventional Runge-Kutta schemes for equation* (6.12).

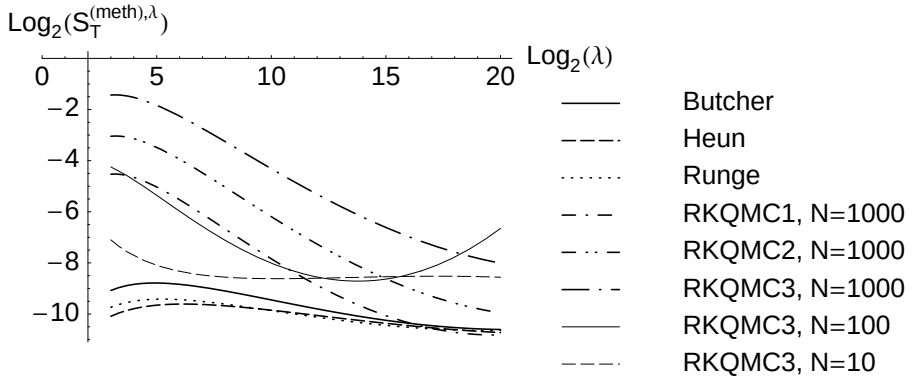

Figure 6.3: *Time-corrected comparison for equation* (6.12).

λ	Butcher	Runge	RKQMC1 $N=1000$	RKQMC2 $N=1000$	RKQMC3 $N=1000$	RKQMC3 $N=10$
2^5	−7.39198	−7.17466	−4.21003	−4.21009	−4.21026	−5.55528
2^6	−9.12876	−8.91401	−6.63217	−6.63697	−6.63852	−5.05383
2^7	−9.39176	−9.16272	−5.71178	−5.7132	−5.71862	−5.64201
2^8	−8.32904	−8.11228	−6.85581	−6.85166	−6.8277	−4.75116
2^9	−7.28245	−7.05248	−5.54155	−5.53954	−5.53957	−5.63427
2^{10}	−8.25863	−8.00724	−6.65457	−6.61329	−6.76324	−4.56086
2^{11}	−7.65107	−7.45674	−8.69076	−8.28356	−8.43429	−6.29302
2^{12}	−8.31722	−8.08087	−9.05153	−8.95747	−8.85853	−2.83075
2^{13}	−9.70187	−9.80577	−9.45962	−9.49673	−9.4834	−8.92099
2^{14}	−10.5467	−8.91969	−11.3958	−11.477	−11.7516	−5.78229
2^{15}	−9.90117	−9.0376	−12.5316	−11.4426	−10.1873	−5.38535
2^{16}	−10.6722	−8.91812	−9.03928	−10.7756	−10.0035	−2.928
2^{17}	−9.77875	−9.56794	−9.40501	−9.2114	−8.78821	−4.24964
2^{18}	−7.38222	−7.32184	−10.7567	−11.135	−10.9163	−5.74555
2^{19}	−9.57111	−9.9538	−9.1594	−9.72612	−9.10414	−5.56226
2^{20}	−10.7389	−9.03267	−12.0962	−12.7167	−13.2895	−6.09049
time	(0.5745s)	(0.331s)	(0.995s)	(2.8885s)	(9.657s)	(0.1075s)

Table 6.2: *Error for increasing values of λ in equation* (6.12). *The last row shows the average time needed for the method.*

Figure 6.2 and table 6.2 show the results of our calculations for increasing $\lambda = 2^k$, $0 \leq k \leq 20$. As one might expect from the the results in [46], for small values of λ, the conventional Runge-Kutta schemes clearly give better results than the quasi-randomized methods presented here. However, for values of λ above 2^{11}, the RKQMC methods become competitive, and give better results for $\lambda > 12$. One has to notice that all three RKQMC methods we investigate show roughly the same behaviour, although in theory they are of different order. However, the number N of sample points is of great importance to the numerical result, especially, taking N too small leads to a large bias in the results. It turned out that about $N \approx 1000$ evaluations for each time step are enough to achieve a sufficient accuracy of the Quasi-Monte Carlo integration involved, so we only present results with $N = 1000$. The comparison above does not yet include the time needed to obtain the solution, which varies greatly with the method. If we compare the methods using the time-corrected error S_T as defined in (6.11), figure 6.3 shows that – although RKQMC methods need a lot more function evaluations – especially the RKQMC1 method still can compete with the conventional Runge Kutta schemes for large values of λ.

As a second example, we chose the delay differential equation

$$\begin{aligned} y'(t) &= -\frac{1}{2}\left(3y\left(t-\frac{1}{2}\right) + 4\cos\left(y\left(t-\frac{\pi}{10}\right)\right) - 5\right) + 3\log(\lambda)^2 t\cos(\lambda t), \quad t \geq 0 \\ y(t) &= 1, \quad t < 0\,, \end{aligned} \tag{6.13}$$

which has an oscillating solution with increasing amplitude. Here again, for a slowly oscillating differential equation conventional Runge Kutta schemes are favorable, while RKQMC methods gain considerably for heavy oscillations, as figures 6.4 and 6.5 and table 6.3 show.

As a final example, we applied the RKQMC methods to the following delay differential equation

$$\begin{aligned} y'(t) &= \pi\frac{\lambda}{2}\left(y\left(t-2-\frac{3}{2\lambda}\right) - y\left(t-2-\frac{1}{2\lambda}\right)\right), \quad t \geq 0 \\ y(t) &= sin(\lambda t\pi), \quad t < 0\,, \end{aligned} \tag{6.14}$$

with the exact solution $y(t) = sin(\lambda t\pi)$, so the problem does not have discontinuities in any derivative, since the initial condition is already the solution of the differential equation. Because the k-th derivative of $y(t)$ can only be bounded by λ^k, for large λ the interpolation error can also only be bounded by a power of λ. One thus has to expect an exploding error, and the solution method will be very unstable. An interesting question in this case is if RKQMC methods – although still obtaining an exploding error – behave at least a little better than conventional Runge Kutta. Or in other words, can the averaging over the whole interval $[t_k, t_{k+1})$ delay the explosion of the error? Figures 6.6 and 6.7 show that while for conventional Runge Kutta schemes the error increases rapidly already for $\lambda > 2^{10}$ (or even $\lambda > 2^7$), the RKQMC methods stay more or less stable until $\lambda > 2^{13}$, and even then the error is several orders of magnitude smaller than with conventional methods.

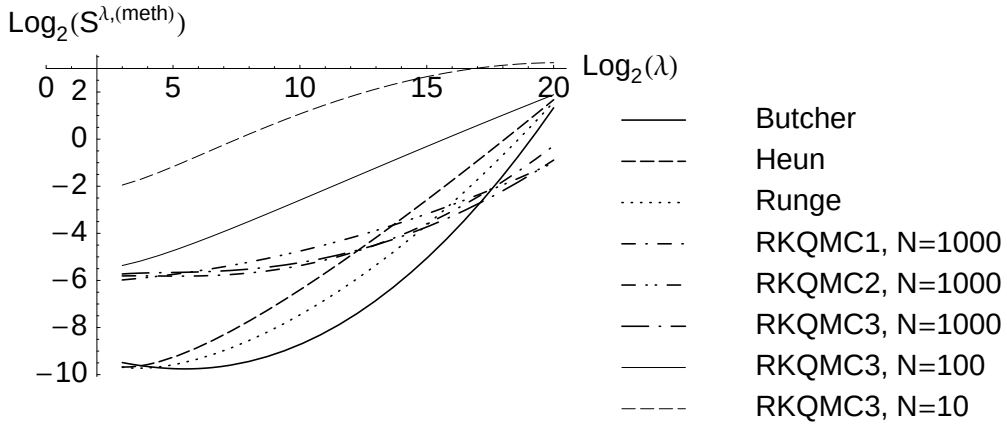

Figure 6.4: *RKQMC vs. conventional Runge-Kutta schemes for equation* (6.13).

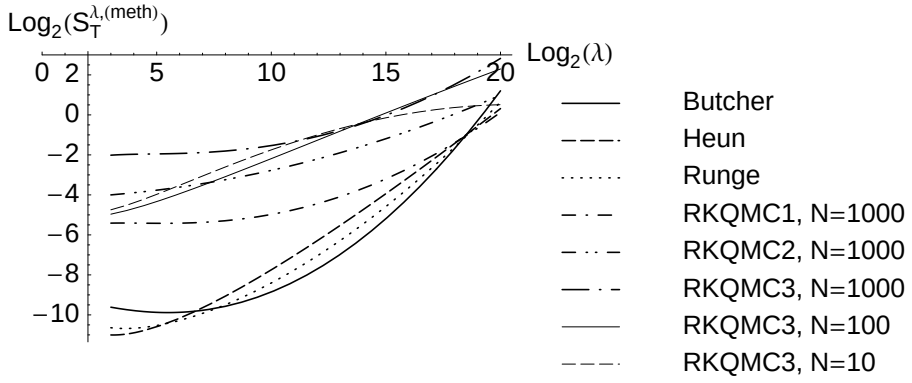

Figure 6.5: *Time-corrected comparison for equation* (6.13).

λ	Butcher	Runge	RKQMC1 $N=1000$	RKQMC2 $N=1000$	RKQMC3 $N=1000$	RKQMC3 $N=10$
2^5	−8.87234	−8.88586	−5.71718	−5.71687	−5.71446	−1.63038
2^6	−9.26313	−9.17274	−5.72643	−5.8531	−5.79093	−1.78973
2^7	−10.1041	−9.86726	−6.31692	−6.26188	−6.02894	−1.98767
2^8	−9.95295	−9.79758	−6.80873	−6.24588	−5.80225	0.31842
2^9	−11.208	−10.7435	−5.94728	−6.03407	−5.95347	0.17748
2^{10}	−11.5016	−11.6134	−5.19338	−3.91158	−4.56006	2.86029
2^{11}	−10.7611	−8.90505	−5.65698	−3.47692	−5.23753	1.29468
2^{12}	−10.6531	−5.24012	−3.95011	−3.41886	−4.53776	3.46736
2^{13}	−5.4083	−1.92602	−6.733	−6.71153	−7.15757	−0.33684
2^{14}	−2.39403	−2.37575	−3.23562	−3.83238	−4.34481	1.77534
2^{15}	−2.04463	−1.2864	−3.86274	−3.65393	−3.62819	2.62794
2^{16}	−2.24314	−1.66459	−1.00946	−0.9048	−1.49998	4.49528
2^{17}	−1.05771	−0.85625	−1.39275	−0.06356	−0.22052	4.10072
2^{18}	−0.42866	−0.42336	−1.11109	−1.2283	−1.71019	3.31797
2^{19}	−1.41526	−1.28994	−0.50625	−0.82309	−1.28871	3.3318
2^{20}	−1.97097	−0.15265	−2.66799	−3.7689	−3.32669	1.8979
time	(0.914s)	(0.52s)	(1.317s)	(3.9515s)	(13.1045s)	(0.144s)

Table 6.3: *Error for increasing values of* λ *in equation* (6.13)

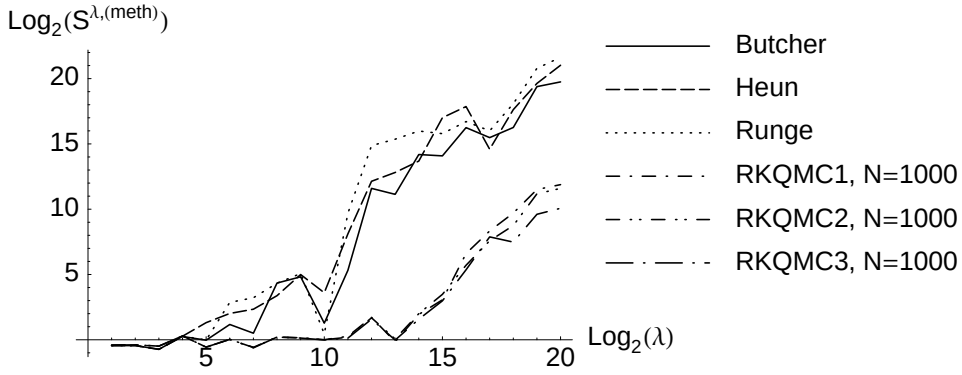

Figure 6.6: *RKQMC can delay numerical instabilities in heavily oscillating equations like equation* (6.14).

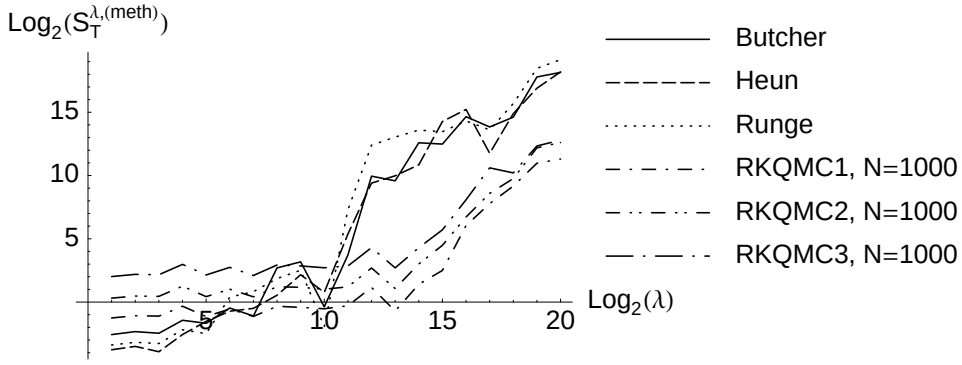

Figure 6.7: *Time-corrected comparison for equation* (6.14).

λ	Butcher	Runge	RKQMC1 $N = 1000$	RKQMC2 $N = 1000$	RKQMC3 $N = 1000$	RKQMC3 $N = 10$
2^5	−0.03254	−0.02311	−0.56156	−0.5624	−0.5612	−0.23613
2^6	1.17245	2.86221	0.02865	0.02788	0.03007	0.33568
2^7	0.50327	3.21856	−0.57879	−0.59252	−0.61976	1.79124
2^8	4.34429	4.33091	0.20416	0.21056	0.20644	1.49253
2^9	4.82238	5.07044	0.13955	0.15332	0.13093	3.3962
2^{10}	1.2717	0.44702	0.00626	0.00308	−0.01275	3.98611
2^{11}	5.32135	9.6919	0.26602	0.19461	0.13048	4.64482
2^{12}	11.5943	14.8719	1.71566	1.68044	1.57747	7.51536
2^{13}	11.1378	15.3605	−0.14943	0.06415	−0.02259	7.20202
2^{14}	14.1873	15.9873	1.93997	1.99741	1.57252	11.7018
2^{15}	14.0843	15.7865	3.05771	3.47916	2.98297	13.5787
2^{16}	16.2482	16.7093	6.6045	5.70631	5.35894	14.4304
2^{17}	15.4825	16.0032	8.33202	7.58254	7.87936	17.3266
2^{18}	16.2581	18.0802	9.70491	8.74674	7.48048	17.3213
2^{19}	19.3875	20.7898	11.5075	11.1776	9.60934	17.2286
time	(0.316s)	(0.1815s)	(0.669s)	(1.977s)	(6.486s)	(0.0725s)

Table 6.4: *All errors for equation* (6.14). *RKQMC methods stay stable for* $2^8 \leq \lambda \leq 2^{13}$, *where conventional Runge Kutta schemes become unstable.*

6.6 Conclusion

In this chapter we showed that the RKQMC methods by Stengle, Lécot, Koudiraty and Coulibaly can also be applied to retarded differential equations. While we already showed this in [46] for equations of one delayed argument, here we presented a more general proof and extended it to equations of several retarded arguments. Our numerical investigation showed, that for slowly varying differential equations, conventional Runge Kutta methods have to be preferred over RKQMC schemes, but for heavily oscillating equations or solutions, RKQMC methods can yield better results than conventional schemes. Although the RKQMC schemes are more expensive as far as computing time is concerned, a larger time step can be chosen. Furthermore, RKQMC schemes may stay stable in a region where conventional Runge Kutta schemes already give an exploding error. All in all, RKQMC solution schemes can be viewed as a good complement to classical Runge Kutta schemes with Hermite interpolation for heavily oscillating delay differential equations.

Part III

QMC Methods for Non-Uniform Integration of Singular Functions and their Applications in Finance

Chapter 7

Quasi-Monte Carlo algorithms for unbounded, weighted integration problems

Contents

In this final part of the thesis we investigate Quasi-Monte Carlo methods for multidimensional improper integrals with respect to a measure other than the uniform distribution. Additionally, the integrand is allowed to be unbounded at the lower boundary of the integration domain. We establish convergence of the Quasi-Monte Carlo estimator to the value of the improper integral under conditions involving both the integrand and the sequence used. Furthermore, we suggest a modification of an approach proposed by Hlawka and Mück for the creation of low-discrepancy sequences with regard to a given density, which are suited for singular integrands.
This chapter is based on a joint work [37] with J. Hartinger and R. Tichy.

7.1 Introduction

This chapter is devoted to Quasi-Monte Carlo (QMC) techniques for weighted integration problems of the form

$$I = \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x}) \, \mathrm{dH}(\mathbf{x}), \tag{7.1}$$

where H denotes a s-dimensional distribution with support $K = [\mathbf{a}, \mathbf{b}] \subseteq \mathbb{R}^s$ and f is a function with singularities on the left boundary of K.

Numerical integration problems of this form frequently occur in practice, e.g. in the field of computational finance. A typical example is the estimation of the mean of a random variate X with support $\mathbb{R}^s$. In case of variates with unbounded variance Monte Carlo simulation is inapplicable, but (after a transformation) QMC algorithms might be available.

Let $H(J)$ denote the probability of $J \subseteq K$ under H. A sequence $\omega = (y_1, y_2, \dots)$ is defined to be H-distributed if for all intervals $\tilde{K} \subseteq K$ the following condition holds:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \chi_{\tilde{K}}(y_n) = H(\tilde{K}).$$

For Riemann integrable (thus bounded) functions it is well known that the integral (7.1) can be approximated by

$$\int_K f(\mathbf{x}) dH(\mathbf{x}) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(y_n),$$

where $(y_n)_{n>0}$ denotes an H-distributed sequence. The aim is to establish conditions on integrands and sequences to guarantee the convergence of QMC techniques for unbounded weighted problems and to justify the commonly used strategy of

ignoring the singularity.

Furthermore techniques proposed by Hlawka and Mück for bounded, weighted integration problems are adapted for the unbounded case.

7.2 Preliminaries and basic definitions

Hlawka [40] established the first quantitative error bound for QMC-algorithms on the s-dimensional unit cube, where H equals the uniform distribution U. To gain a deep insight in the classical QMC theory we refer to the monographs of Kuipers and Niederreiter [51] or Niederreiter [53]. Quasi-Monte Carlo algorithms for (Riemann) improper integrals with regard to the uniform distribution ($H = U\,[0,1]^s$) were first investigated by Sobol. In [64] he dealt with integrands, which are unbounded for $x_1 x_2 \cdots x_s \to 0$. Asymptotic error estimates, as well as numerical examples for some special functions were presented by Klinger [49], De Doncker and Guan [18].

We consider integration problems, which have the following properties

- The improper integral (7.1) on a compact subinterval of $\mathbb{R}^s$ exists in the sense that the limes

$$\lim_{\substack{\mathbf{c}\to\mathbf{a} \\ \mathbf{c}>\mathbf{a}}} \int_{[\mathbf{c},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x}) \tag{7.2}$$

  exists independent of the $\mathbf{c}$, where $\mathbf{c} > \mathbf{a}$ should be understood component-wise. We will define this limes as the value of the improper integral.

- If the integration domain is not a compact subinterval of $\mathbb{R}^s$, i.e. at least one coordinate is unbounded ($a_i = -\infty$ or $b_i = \infty$ for a $1 \leq i \leq s$), the limes

$$\lim_{\mathbf{c}\to\mathbf{a}^+} \int_{[\mathbf{c},\mathbf{d}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})$$

has to exist for every finite fixed upper boundary $\mathbf{d} \in (\mathbf{a}, \mathbf{b})$, $\|\mathbf{d}\| < \infty$. Independently, for any fixed $\mathbf{c} \in [\mathbf{a}, \mathbf{b}]$ as lower integration boundary with $\mathbf{c} > \mathbf{a}$ component-wise, the limes

$$\lim_{\mathbf{d}\to\mathbf{b}^-} \int_{[\mathbf{c},\mathbf{d}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})$$

needs to exists. Then we let the lower and upper integration bounds independently tend to the boundaries of the integration domain K and define the value of the improper integral as

$$\lim_{\substack{\mathbf{d}\to\mathbf{b}^- \\ \mathbf{c}\to\mathbf{a}^+}} \int_{[\mathbf{c},\mathbf{d}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\,.$$

- The integrand $f(\mathbf{x})$ possibly has singularities at the left boundary of the integration domain, i.e. $\lim_{x_i\to a_i} |f(x)| = \infty$ for some $i \in \{1, \ldots, s\}$.
- These singularities are the only singularities of $f(\mathbf{x})$. In particular this means that for all $\varepsilon > 0$ there exists some $M < \infty$, such that $|f(x)| < M$ for all $x \in [a + \varepsilon, b]$.

Before we recall Hlawka's integration error bound (respectively a slight generalization for weighted integration) some more definitions are required:

Definition 7.1. The H-discrepancy of $\omega = (y_1, y_2, \ldots)$ measures the distribution properties of the sequence. It is defined as

$$D_{N,H}(\omega) = \sup_{J\subseteq K} \left| \frac{1}{N} A_N(J,\omega) - H(J) \right|, \tag{7.3}$$

where A_N counts the number of elements in $(y_1, \ldots, y_N)$ falling into the interval J, i.e.

$$A_N(J,\omega) = \sum_{n=1}^{N} \chi_J(y_n).$$

Definition 7.2. By a partition P of K we mean a set of s finite sequences ($j = 1, \ldots, s$),

$$a_j = \nu_0^{(j)} \leq \nu_1^{(j)} \leq \cdots \leq \nu_{m_j}^{(j)} = b_j.$$

In connection with such a partition one defines for each j two operators by

$$\Delta_j f(x^{(1)}, \ldots, x^{(j-1)}, \nu_i^{(j)}, x^{(j+1)}, \ldots, x^{(s)}) = \\ f(x^{(1)}, .., x^{(j-1)}, \nu_{i+1}^{(j)}, x^{(j+1)}, .., x^{(s)}) - f(x^{(1)}, .., x^{(j-1)}, \nu_i^{(j)}, x^{(j+1)}, .., x^{(s)})$$

for $0 \leq i < m_j$ and

$$\Delta_j^* f(x^{(1)}, \ldots, x^{(j-1)}, 1, x^{(j+1)}, \ldots, x^{(s)}) = \\ f(x^{(1)}, .., x^{(j-1)}, \nu_{i+1}^{(j)}, x^{(j+1)}, .., x^{(s)}) - f(x^{(1)}, .., x^{(j-1)}, 0, x^{(j+1)}, .., x^{(s)}).$$

Definition 7.3. For a function f on K, we set the variation in the sense of Vitali as follows

$$V^{(l)}(f) = \sup_P \sum_{i_1=0}^{m_1-1} \cdots \sum_{i_l=0}^{m_l-1} \left| \Delta_{1,\ldots,l} f(\nu_{i_1}^{(1)}, \ldots, \nu_{i_l}^{(l)}) \right|,$$

where the supremum is extended over all partitions P of K. The variation of f restricted to an interval $[\mathbf{a}, \mathbf{b}]$ will be denoted $V_{[\mathbf{a},\mathbf{b}]}(f)$.

Finally let $1 \le l \le s$, $1 \le i_1 < \cdots < i_l \le s$ and $V^{(l)}(f; i_1, \ldots, i_l)$ be the Vitali variation of the restriction of f to the l-dimensional face

$$\{(u_1, \ldots, u_s) \in K : u_j = b_j \text{ for } j \neq i_1, \ldots, i_l\}.$$

The variation in the sense of Hardy and Krause is defined by

$$V(f) = \sum_{l=1}^{s} \sum_{1 \le i_1 < \cdots < i_l \le s} V^{(l)}(f; i_1, \ldots, i_l). \tag{7.4}$$

Naturally f is called of bounded variation on K, when $V(f) < \infty$.

Theorem 7.1 (Koksma-Hlawka Inequality). *Let f be a function of bounded variation (in the sense of Hardy and Krause) on K and $\omega = (y_1, y_2, \ldots)$ a sequence on K. The QMC integration error can be bounded by*

$$\left| \int_K f(\mathbf{x}) dH(\mathbf{x}) - \frac{1}{N} \sum_{n=1}^{N} f(y_n) \right| \le V(f) D_{N,H}(\omega). \tag{7.5}$$

A proof of this central theorem can be found e.g. in [51] for the uniform distribution, and in [70] for general distributions H (using a slight specialization to variation in the measure sense).

In [64], Sobol looked at Quasi-Monte Carlo integration of singular functions with respect to the uniform distribution. He gave conditions for its convergence involving the function as well as the sequence used. In particular, his theorem reads:

Theorem 7.2 (Sobol, [64]). *Let $i' \subseteq \{1, \ldots, s\}$ and $\mathcal{K}_{i'}$ be the boundary of $[0,1]^s$ where all coordinates $x^{(i)} = 1$ for $i \notin i'$. Let furthermore $c < c_N$, where $c_N = \min_{1 \le \mu \le N} \left(x_\mu^{(1)}, \ldots, x_\mu^{(s)} \right)$, and $G_{i'}$ the part of $\mathcal{K}_{i'}$, where $x^{(i_1)}, \ldots, x^{(i_j)} \ge c$. If for every i' the integral*

$$\int_{\mathcal{K}_{i'}} x^{(i_1)} \cdots x^{(i_j)} \left| f^{(i')}(x^{(i')}) \right| dx^{(i')} \tag{7.6}$$

converges and

$$D_N^{i'}(x_1, x_2, \ldots) \int_{G_{i'}(c)} \left| f^{(i')}(x^{(i')}) \right| dx^{(i')} = o(1), \tag{7.7}$$

then $\lim_{N\to\infty} \frac{1}{N} \sum_{\mu=1}^{N} f(x_\mu) = \int_{[0,1]^s} f(x) dx$.

In this paper we will generalize this result to weighted integration, which means integration with respect to arbitrary densities $h(x)$. We will also investigate several ways to construct the sequences needed for QMC integration w.r.t. arbitrary densities and propose ways to overcome some problems occurring in conventional methods.

7.3 The one-dimensional case

Let us first consider the one-dimensional case. The main theorem of this section generalizes and combines the Koksma inequality (7.1) in one dimension with Sobol's convergence theorem [64] for QMC integration of singular integrands. Instead of the uniform distribution, we will assume an arbitrary distribution $H(x)$ on $[a,b]$ and show convergence of the QMC integration. The multi-dimensional case will be treated in a similar manner in the next section, so we present the simpler one-dimensional case here to give a clear picture of the ideas.

7.3.1 Convergence theorem

Definition 7.4. For a sequence $\omega = (y_1, y_2, \dots)$ let $c_N = \min_{1 \le n \le N} y_n$ be the smallest value of the first N elements of the sequence.

Theorem 7.3. *Let $a \le c \le c_N$. If a sequence $\{y_i\}_{i \in \mathbb{N}}$ and a differentiable function $f(x)$ on $[a,b]$ with a singularity only at the left boundary satisfy the condition*

$$D_{N,H}(\omega) \int_c^b \left| f'(x) \right| dx = o(1) \tag{7.8}$$

as well as $c_N \to a$ for $N \to \infty$, then the QMC estimator $\frac{1}{N} \sum_{n=1}^{N} f(y_n)$ converges to the value of the improper integral of $f(x)$ on $[a,b]$:

$$\lim_{N \to \infty} \frac{1}{N} \sum_{n=1}^{N} f(y_n) = \int_a^b f(x) dH(x) \tag{7.9}$$

Remark 7.1. For non-differentiable functions the condition is similar using the variation $V_{[c_N,b)}(f)$ instead of the integral of the derivative. The multidimensional case will be formulated in such a more general manner.

Proof. To prove (7.9) we will approximate $\int_c^b f(x) dH(x)$ with $\frac{1}{N} \sum_{n=1}^{N} f(y_n)$ and show that the remaining terms tend to zero. Without loss of generality we can assume the sequence to be sorted, and we define y_0 and y_{N+1} so that $c = y_0 \le y_1 \le \dots \le y_N \le y_{N+1} = b$.
First, we establish an identity similar to Lemma 5.1 of [51]:

$$\frac{1}{N} \sum_{n=1}^{N} f(y_n) - \int_c^b f(x) dH(x) = H(c) \cdot f(b) - \int_c^b \Big(\frac{A_N([c,x)\,, \omega)}{N} - H([c,x)) \Big) df(x) \tag{7.10}$$

The above identity can be proved by inserting terms, applying integration by parts and using

the fact that $H(x)$ is a distribution function on $[a,b]$.

$$\begin{aligned}
&\frac{1}{N}\sum_{n=1}^{N} f(y_n) - \int_c^b f(x)dH(x) \\
&\quad = \frac{1}{N}\sum_{n=1}^{N} f(y_n) - (1-H(c))\, f(b) + [H(x)-H(c)]\, f(x)\big|_c^b - \int_c^b f(x)dH(x) \\
&\quad = f(y_{N+1}) - \sum_{n=0}^{N} \frac{n}{N}\left(f(y_{n+1}) - f(y_n)\right) \\
&\qquad - f(b) + H(c)f(b) + \int_c^b [H(x) - H(c)]\, df(x) \\
&\quad = -\sum_{n=0}^{N} \int_{y_n}^{y_{n+1}} \frac{n}{N} df(x) + H(c)f(b) + \int_c^b [H(x) - H(c)]\, df(x) \\
&\quad = H(c)f(b) - \int_c^b \Big(\frac{A_N([c,x)\,,\omega)}{N} - \cdot H([c,x))\Big) df(x)
\end{aligned}$$

Once this is established, the convergence is obvious:

$$\begin{aligned}
&\left|\frac{1}{N}\sum_{n=1}^{N} f(y_n) - \int_a^b f(x)dH(x)\right| \\
&\qquad\qquad \leq \left|\int_a^c f(x)dH(x)\right| + |H(c)|\,|f(b)| + D_{N,H}(\omega)\int_c^b \left|f'(x)\right| dx
\end{aligned}$$

According to the convergence of the improper integral on the whole interval, the first and second terms tend to zero as $c \to a$, and the condition of the theorem ensures that the third term is $o(1)$ and thus also tends to zero. □

Remark 7.2. The minimal element c_N of the one-dimensional Halton-sequence in basis 2 (which coincides with the Faure and Sobol sequences) is larger than $1/2N$. In arbitrary bases p, the smallest element among the first N can be bounded from below by $1/p^{\lfloor \log_p N \rfloor + 1}$.

7.3.2 Generating suitable sequences

The main drawback, from a practical point of view, of the preceding theorem is the lack of sequences with low H-discrepancy. In practical issues one has to generate these sequences by transformation from uniform low discrepancy sequences. For distributions, where explicit inverse distribution functions are available, the transformation of the uniform distributed sequence $(x_1, x_2, \dots)$ by an inversion method $y_i = H^{-1}(x_i)$ is obvious. This transformation preserves the discrepancy, i.e

$$D_N\left(x_1, x_2, \dots\right) = D_{N,H}\left(y_1, y_2, \dots\right). \tag{7.11}$$

Unfortunately for most distributions the inverse distribution function is not given explicitly and direct numerical methods for the inversion are often inefficient.

In [43], Hlawka and Mück propose a systematic method for constructing H-distributed sequences, which just uses the distribution function but not its inverse. We focus on the case $K = [0,1]$.

Definition 7.5. Let $\omega = (x_1, x_2, \dots)$ be a sequence in $[0,1)$ with discrepancy $D_N(\omega)$ with regard to the uniform distribution. The sequence $\tilde{\omega} = (\tilde{y}_1, \tilde{y}_2, \dots)$ is defined to be the sequence consisting of the points

$$\tilde{y}_k = \frac{1}{N} \sum_{r=1}^{N} \lfloor 1 + x_k - H(x_r) \rfloor = \frac{1}{N} \sum_{r=1}^{N} \chi_{[0,x_k]} \left(H(x_r) \right). \tag{7.12}$$

Lemma 7.4. *Let the sequence $\tilde{\omega}$ be defined by Eq. (7.12) and $M = sup_{x\in[0,1]} h(x)$. Then the H-discrepancy of $\tilde{\omega}$ can be bounded by the following inequality*

$$D_{N,H}(\tilde{\omega}) \leq (1 + M) D_N(\omega).$$

All points constructed by the Hlawka-Mück method are of the form i/N, $(i = 0, \dots, N)$, in particular, some elements $\tilde{y}_k$ of the transformed sequence $\tilde{\omega}$ might assume a value of 0. Since this is the singularity of $f(x)$, according to Theorem 7.3 these sequences are not directly suited for unbounded problems.

Definition 7.6. To overcome this problem, we define the sequence $\bar{\omega}$ for $i = 1, \dots, N$ as follows:

$$\bar{y}_k = \begin{cases} \tilde{y}_k & \text{if } \tilde{y}_k \geq \frac{1}{N}, \\ \frac{1}{N} & \text{if } \tilde{y}_k = 0. \end{cases} \tag{7.13}$$

Theorem 7.5. *The H-discrepancy of $\bar{\omega}$ is bounded by*

$$D_{N,H}(\bar{\omega}) \leq (M + 1) \left(D_N(\omega) + \frac{1}{N} \right).$$

Proof. Let $(\bar{x}_1, \bar{x}_2, \dots)$ and $(\tilde{x}_1, \tilde{x}_2, \dots)$ be the sequences obtained by $\bar{x}_i = H(\bar{y}_i)$, resp. $\tilde{x}_i = H(\tilde{y}_i)$. By Eq. (7.11) it is sufficient to estimate the uniform discrepancy of $(\bar{x}_1, \bar{x}_2, \dots)$. For uniform discrepancies the following fact is well known (e.g. Niederreiter [53]): Let $(u_1, u_2, \dots)$ and $(v_1, v_2, \dots)$ be two sequences in $[0,1]$ with discrepancies D_1 and D_2, then $|D_1 - D_2| < \varepsilon$, whenever $\max_{1\leq i\leq N} |u_i - v_i| < \varepsilon$.
Now, the calculation

$$\begin{aligned} |x_k - \bar{x}_k| &\leq |x_k - \tilde{x}_k| + |\tilde{x}_k - \bar{x}_k| \\ &= \left| \int_{y_k}^{\tilde{y}_k} h(t)dt \right| + \left| \int_{\bar{y}_k}^{\tilde{y}_k} h(t)dt \right| \leq M \left(D_N(x_1, x_2, \dots) + \frac{1}{N} \right). \end{aligned}$$

completes the proof. □

Remark 7.3. The discrepancy of uniform low discrepancy sequences is typically of the order $\mathcal{O}(\frac{\log N}{N})$ (resp. $\mathcal{O}(\frac{1}{N})$ for nets). Therefore the additional factor $(1/N)$ one inherits though the shift (7.13) does not affect the asymptotic behavior of the integration error.

Remark 7.4. Another method to avoid the inversion can be obtained by a suitable integral transformation. With the help of such a transformation, in Monte Carlo algorithms often referred to as importance sampling, it might even be possible to avoid some singularities.

7.4 Multivariate singular integration

We will now look at arbitrary-dimensional integrals

$$\int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\,, \tag{7.14}$$

where the integration domain is taken as a compact subinterval $[\mathbf{a},\mathbf{b}]$ of $\mathbb{R}^s$.

Remark 7.5. This is no restriction: If any of the dimensions is unbounded, we can first carry out the calculation on compact intervals and then take the limits to infinity as indicated in Section 7.2.

7.4.1 Preliminaries

The notations and operators used in the sequel are defined in Section 7.2. In addition we need the summation symbol $\sum^*$:

Definition 7.7. Given an expression F depending on variables $x^{(r)},\ldots,x^{(s)}$ and a partition of $N_{r,s} = \{r, r+1,\ldots,s\}$ into two subsets $L = \{x^{(l_1)},\ldots,x^{(l_p)}\}$ and $N_{r,s} \setminus L = \{x^{(l_{p+1})},\ldots,x^{(l_{s-r})}\}$, we use the notation

$$F(L) = F\left(x^{(l_1)},\ldots,x^{(l_p)};x^{(l_{p+1})},\ldots,x^{(l_{s-r})}\right).$$

The summation operator $\sum^*$ is defined as the sum over all elements of the set $\mathcal{P}_p = \{L \subseteq N_{r,s} : \mathrm{card}(L) = p\}$, i.e.

$$\sum_{r,\ldots,s;p}^{*} F = \sum_{L\in\mathcal{P}_p} F(L).$$

The integration by parts formula used in the proof of the previous section for differentiable one-dimensional functions can be generalized to arbitrary s-dimensional functions. It is commonly referred to as Abel's summation formula:

Lemma 7.6 (Abel's summation formula, e.g. [51]). *Let $f(\mathbf{x})$ and $g(\mathbf{x})$ two functions on $[\mathbf{a},\mathbf{b}]$, and let $\left(\eta_0^{(j)},\eta_1^{(j)},\ldots,\eta_{m_j}^{(j)}\right)$ and $\left(\xi_0^{(j)},\xi_1^{(j)},\ldots,\xi_{m_j}^{(j)}\right)$ with $j = 1,\ldots,s$ be two partitions of the interval $[\mathbf{a},\mathbf{b}]$. Then*

$$\begin{aligned}
&\sum_{i_1=0}^{m_1-1}\cdots\sum_{i_s=0}^{m_s-1} f\left(\xi_{i_1+1}^{(1)},\ldots,\xi_{i_s+1}^{(s)}\right)\Delta_{1,\ldots,s}g\left(\eta_{i_1}^{(1)},\ldots,\eta_{i_s}^{(s)}\right)\\
&\qquad=\sum_{p=0}^{s}(-1)^p\sum_{1,\ldots,s;p}^{*}\Delta^*_{p+1,\ldots,s}\sum_{i_1=0}^{m_1-1}\cdots\sum_{i_p=0}^{m_p-1} g\left(\eta_{i_1}^{(1)},\ldots,\eta_{i_p}^{(p)},x^{(p+1)},\ldots,x^{(s)}\right)\\
&\qquad\qquad\times\Delta_{1,\ldots,p}f\left(\xi_{i_1}^{(1)},\ldots,\xi_{i_p}^{(p)},x^{(p+1)},\ldots,x^{(s)}\right)\,.
\end{aligned}\tag{7.15}$$

For a proof of this important equation we refer to the monograph [51] of Kuipers and Niederreiter.

7.4.2 Convergence theorem

Using Abel's summation formula (7.15), we can now prove the convergence of the s-dimensional Quasi-Monte Carlo estimator to the value of the improper integral (7.14), even though the integrand can be singular on the whole left boundary of the integration area. Conventional methods usually apply the Koksma-Hlawka inequality (7.5). Here this inequality does not give an upper bound for the integration error, because the singularity causes the function to be of unbounded variation on $[\mathbf{a}, \mathbf{b}]$. We only require the function to be of bounded variation on every compact subinterval of $(\mathbf{a}, \mathbf{b}]$. The proof will follow the lines of the proof of the Koksma-Hlawka inequality given in [51], so for some parts of the proof we just refer to that book.

Theorem 7.7 (Convergence of the multidimensional QMC estimator). *Let $f(\mathbf{x})$ be a function on $[\mathbf{a}, \mathbf{b}]$ with singularities only at the left boundary of the definition interval (i.e. $f(\mathbf{x}) \to \pm\infty$ only if $x^{(j)} \to a_j$ for at least one j), and let furthermore $c_{N,j} = \min_{1 \le n \le N} y_n^{(j)}$ and $a_j < c_j \le c_{N,j}$. If the improper integral (7.14) exists in the sense of Section 7.2, and if*

$$D_{N,H}(\omega) \cdot V_{[\mathbf{c},\mathbf{b}]}(f) = o(1), \tag{7.16}$$

then the QMC estimator converges to the value of the improper integral:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{y}_n) = \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\, \mathrm{dH}(\mathbf{x}) \; . \tag{7.17}$$

Proof. Like in the one-dimensional case we estimate the integration error on the interval $[\mathbf{c}, \mathbf{b}]$, where the function $f(\mathbf{x})$ is regular for any choice of $\mathbf{c}$, and show that the remaining terms vanish as $\mathbf{c} \to \mathbf{a}$. Again similar to the proof of Theorem 7.3 we use a function

$$g(\mathbf{x}) = \frac{1}{N} A([\mathbf{c}, \mathbf{x}), \omega) - H([\mathbf{c}, \mathbf{x})) \tag{7.18}$$

for a given sequence $\omega = (y_n^{(j)})$, $1 \le n \le N$, $1 \le j \le s$. One has to notice that this function is merely the function used in the definition of the discrepancy, so that $\sup_{\mathbf{x}\in[\mathbf{c},\mathbf{b}]} |g(\mathbf{x})| \le D_{N,H}(\omega)$.
Using a double partition $c_j = \xi_0^{(j)} = \eta_0^{(j)} \le \xi_1^{(j)} < \eta_1^{(j)} \le \cdots < \eta_{m_j}^{(j)} = \xi_{m_j+1}^{(j)} = b_j$ $(j = 1, \ldots, s)$ of the interval $[\mathbf{c}, \mathbf{b}]$ with the additional condition that the $\xi_1^{(1)}, \ldots, \xi_{m_j}^{(j)}$ contain at least our sequence ω, we now apply Lemma 7.6 to the function $g(\mathbf{x})$. As argued in [51], the left hand side of Eq. (7.15),

$$\sum_{i_1=0}^{m_1-1} \cdots \sum_{i_s=0}^{m_s-1} f\left(\xi_{i_1+1}^{(1)}, \ldots, \xi_{i_s+1}^{(s)}\right) \Delta_{1,\ldots s} g\left(\eta_{i_1}^{(1)}, \ldots \eta_{i_s}^{(s)}\right),$$

can be simplified to

$$\mathrm{LHS} = \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{y}_n) - \sum_{i_1=0}^{m_1-1} \cdots \sum_{i_s=0}^{m_s-1} f\left(\xi_{i_1+1}^{(1)}, \ldots, \xi_{i_s+1}^{(s)}\right) \Delta_{1,\ldots s} H\left(\bigotimes_{l=1}^{s} \left[c_l, \eta_{i_l}^{(l)}\right)\right), \tag{7.19}$$

where $\otimes$ denotes the Cartesian product.

For the right hand side we notice that $g(\mathbf{x}) = 0$ if any of the $x^{(j)} = c_j$. Also, $g(\mathbf{b}) = 1 - H([\mathbf{c}, \mathbf{b}))$, so that the summand of $p = 0$ can be simplified as

$$\Delta^*_{1,\ldots,s} g\left(x^{(1)}, \ldots x^{(s)}\right) f(x^{(1)}, \ldots, x^{(s)}\Big) = g(\mathbf{b}) f(\mathbf{b}). \tag{7.20}$$

Similarly, for $1 \le p \le s$, only terms with $x^{(p+1)} = b_{p+1}, \ldots, x^{(s)} = b_s$ contribute (if any $x^{(p+j)} = c_{p+j}$, the function value of $g(\mathbf{x})$ vanishes):

$$\begin{aligned} |\mathrm{RHS}| &\le |g(\mathbf{b}) f(\mathbf{b})| + \sum_{p=1}^{s} |(-1)^p| \sum_{1,\ldots,s;p}^{*} \sum_{i_1=0}^{m_1} \cdots \sum_{i_s=0}^{m_s} \left| g\left(\eta^{(1)}_{i_1}, \ldots, \eta^{(p)}_{i_p}\right)\right| \\ &\qquad \times \left|\Delta_{1,\ldots,p} f\left(\xi^{(1)}_{i_1}, \ldots, \xi^{(p)}_{i_p}, b_{p+1}, \ldots, b_s\right)\right| \qquad (7.21) \\ &\le |g(\mathbf{b}) f(\mathbf{b})| + \sum_{p=1}^{s} \sum_{1,\ldots,s;p}^{*} D_{N,H}(\omega_{p+1,\ldots,s}) V^{(p)}(f(x^{(1)}, ..., x^{(p-1)}, b_{p+1}, ..., b_s). \end{aligned}$$

with $\omega_{p+1,\ldots,s}$ denoting the projection of the sequence ω on the upper boundary of $[\mathbf{a}, \mathbf{b}]$ so that the components $i_{p+1}, \ldots, i_s$ are set to $b^{(i_{p+1})}, \ldots, b^{(i_s)}$, and the discrepancy is computed on the face of $[\mathbf{a}, \mathbf{b}]$, in which $\omega_{p+1,\ldots,s}$ is contained.

Since the second term in Eq. (7.19) is nothing else but a Riemann-Stieltjes sum to the integral $\int_{[\mathbf{c},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})$, and the rest is independent of the double partition, we let the mesh size of the double partition tend to zero

$$\max_{1\le j\le s} \max_{0\le i\le m_j} (\eta^{(j)}_{i+1} - \eta^{(j)}_i) \to 0$$

to obtain the multidimensional version of Eq. (7.10):

$$\left|\frac{1}{N}\sum_{n=1}^{N} f(\mathbf{y}_n) - \int_{[\mathbf{c},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\right| \le |(1 - H([\mathbf{c}, \mathbf{b})))f(\mathbf{b})| + |D_{N,H}(\omega)| \cdot \left|V_{[\mathbf{c},\mathbf{b}]}(f)\right| \tag{7.22}$$

The existence of the improper integral in the sense of Section 7.2 guarantees that

$$I_{\mathrm{rest}} := \left|\int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x}) - \int_{[\mathbf{c},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\right|$$

tends to zero as we let $\mathbf{c} \to \mathbf{a}$. Thus, we have

$$\begin{aligned} \left|\frac{1}{N}\sum_{n=1}^{N} f(\mathbf{y}_n) - \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\right| &\le \left|\frac{1}{N}\sum_{n=1}^{N} f(\mathbf{y}_n) - \int_{[\mathbf{c},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x})\right| + I_{\mathrm{rest}} \\ &\le |[1 - H([\mathbf{c}, \mathbf{b}))]\, f(\mathbf{b})| + D_{N,H}(\omega) \cdot V_{[\mathbf{c},\mathbf{b}]}(f) + I_{\mathrm{rest}}\,. \end{aligned} \tag{7.23}$$

For $N \to \infty$ and so $\mathbf{c} \to \mathbf{a}$, the first and third terms obviously tend to zero, while the conditions of the theorem guarantee this also for the second term. Thus the proof is finished. □

7.4.3 Generation suitable multi-dimensional sequences

As in the one-dimensional case, one faces the problem of generating H-distributed sequences. Multidimensional versions for inversion methods are well known (e.g. Devroye [22]). Under

weak, Lipschitz type conditions, Hlawka and Mück [42] showed the following bound for the H-discrepancy of sequences $\tilde{\omega}$ generated by multidimensional inversion methods:

$$D_{N,H}(\tilde{\omega}) \leq c\,(D_N(\omega))^{1/s}\,, \tag{7.24}$$

where ω denotes the original uniformly distributed sequence. Furthermore in [42] a multidimensional approach similar to Eq. (7.12) is given.

We again focus on distributions on $[0,1]^s$ and specialize our analysis to distributions with independent marginals, i.e. $H(\mathbf{x}) = \prod_{i=1}^{s} H_i(x^{(i)})$.
In this case one can transform each dimension separately by one-dimensional inversion methods and improve the bound (7.24) similar to the one-dimensional case to

$$D_{N,H}(\tilde{\omega}) = D_N(\omega). \tag{7.25}$$

Hlawka [41] suggested to apply the construction (7.12) to every dimension individually to avoid an inversion of $H_1, \ldots, H_s$ and gave a bound on the H-discrepancy of sequences $\bar{\omega}$ generated by this procedure:

$$D_{N,H}(\bar{\omega}) \leq (1 + 3sM) D_N(\omega),$$

where $M = \sup h(\mathbf{x})$.

Similar to one dimension, Hlawka's method might lead to sequences, which are not suited for unbounded integrands. Our final theorem shows a modification, which leads to QMC estimators for a wide range of functions. Before we can state it, we need to recall a lemma, which will be essential for the proof.

Lemma 7.8 (e.g. [43]). *Let $\Omega_1 = (u_1, \ldots, u_N)$ and $\Omega_2 = (v_1, \ldots, v_N)$ be two sequences in $[0,1]^s$. If the condition*

$$|u_i^{(j)} - v_i^{(j)}| \leq \varepsilon_j,$$

holds for all $1 \leq j \leq s$ and all $1 \leq i \leq N$, we get the following bound on the difference of the discrepancies

$$|D_N(\Omega_1) - D_N(\Omega_2)| \leq \prod_{j=1}^{s}(1 + 2\varepsilon_j) - 1. \tag{7.26}$$

Theorem 7.9. *Let H be a s-dimensional distribution with independent marginal distributions $H_1, \ldots, H_s$ and $M = \sup h(\mathbf{x}) < \infty$. Let furthermore $\omega = (x_1, x_2, \ldots)$ be a sequence with (uniform) discrepancy $D_N(\omega)$ and define the sequence $\tilde{\omega} = (\tilde{y}_1, \tilde{y}_2, \ldots)$ by*

$$\tilde{y}_i^{(j)} = \frac{1}{N} \sum_{n=1}^{N} \lfloor 1 + x_i^{(j)} - H_i(x_n^{(j)}) \rfloor,$$

for $j = 1, \ldots, s$ and $n = 1, \ldots, N$. Then the sequence $\bar{\omega}$

$$\bar{y}_k^{(j)} = \begin{cases} \tilde{y}_k^{(j)} & \text{if } \tilde{y}_k^{(j)} \geq \frac{1}{N}, \\ \frac{1}{N} & \text{if } \tilde{y}_k^{(j)} = 0 \end{cases} \tag{7.27}$$

has the following two properties:

$$D_{N,H}(\bar{\omega}) \leq (1 + 4M)^s D_N(\omega)$$

$$\min_{1 \leq j \leq s, 1 \leq i \leq N} \tilde{y}_i^{(j)} \geq \frac{1}{N}$$

Proof. Let $\bar{x}_i^{(j)} = H_i^{-1}(\bar{y}_i^{(j)})$. Similar to Theorem 7.5 we get the inequality

$$|x_i^{(j)} - \bar{x}_i^{(j)}| \leq M \left(D_N(\omega) + \frac{1}{N} \right).$$

Using Lemma 7.8 we obtain

$$D_N(\bar{x}_1, \bar{x}_2, \dots) \leq D_N(\omega) - 1 + \left(1 + M \left(D_N(\omega) + \frac{1}{N} \right) \right)^s.$$

By Eq. (7.25) and the inequality $\frac{1}{N} \leq D_N(\omega) \leq 1$ it follows that

$$D_{N,H}(\bar{\omega}) \leq D_N(\omega) - 1 + \left(1 + M \left(D_N(\omega) + \frac{1}{N} \right) \right)^s \leq (1 + 4M)^s D_N(\omega).$$

□

7.5 Conclusion

In this article we successfully showed that Quasi-Monte Carlo methods can also be applied to improper s-dimensional integrals, where the integrand function $f(\mathbf{x})$ becomes singular at the integration boundary, assuming the function and the low-discrepancy sequence in use display certain properties as listed in Theorem 7.7. While similar conditions have long been known [64] for integration with respect to the uniform distribution (i.e. using uniformly distributed low-discrepancy sequences), we were able to generalize these results to integration with respect to arbitrary multidimensional densities, often called weighted integration.
In [43] and [41] Hlawka and Mück proposed a scheme to transform a uniformly distributed low-discrepancy to a low-discrepancy sequence with given density. However, the new sequence does not in general fulfill the conditions set forth in our convergence theorems and so cannot be used for the QMC integration of singular integrands. We were able to give a slight modification of the transformed sequence so that it does not loose its low discrepancy, but can be used for QMC weighted integration of singular integrands as our convergence theorems show.
We looked at one-dimensional densities and multidimensional densities which can be factored. Arbitrary multidimensional densities lead to the problem that even the inversion method does in general not preserve the discrepancy of a low-discrepancy sequence. Therefore the creation of low-H-discrepancy sequences still poses an open problem, in particular sequences suited for singular integrands.

Chapter 8

Option Pricing as an Unbounded Non-Uniform Quasi-Monte Carlo Integration Problem

Contents

In various applications in computational finance the implementation of efficient Quasi-Monte Carlo algorithms is faced with (at least) two non-standard problems. Firstly one has two create variates of miscellaneous non-uniform variates, i.e. sequences with low discrepancy with respect to non-uniform distributions. Secondly many integration problems considered in finance lead to unbounded integrands, so that one has to adapt Koksma-Hlawka's inequality to guarantee the convergence of QMC algorithms. The aim of this chapter is to study the performance of Hlawka-Mück methods (as presented in the previous chapter) in a typical financial application and compare the convergence behavior with Monte Carlo methods and algorithms which use integral transformations to avoid singular integrands and the generation of special variates.
The results presented here stem from a joint work [36] with J. Hartinger and M. Predota.

8.1 Introduction

Recently, Hartinger et al. [37] studied Quasi-Monte Carlo integration (QMC) for unbounded, weighted problems, i.e

$$\int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\,\mathrm{dH}(\mathbf{x}), \tag{8.1}$$

where H denotes an m-dimensional distribution with support $[\mathbf{a},\mathbf{b}] \subseteq \mathbb{R}^m$ and f is a function with singularities at the left boundary of $[\mathbf{a},\mathbf{b}]$. They gave sufficient conditions on integrand-sequence combinations to guarantee the convergence of the QMC estimates and proposed adapted Hlawka-Mück techniques to allow integration of functions with singularities at the integration boundary. QMC integration for singular functions was first considered by Sobol [64], where he dealt with integrands on $I^m = [0,1]^m$, which are unbounded for $x_1 x_2 \cdots x_s \to 0$. Klinger [49] and De Doncker and Guan [18] presented asymptotic error estimates as well as numerical examples for special functions.
Acceptance-rejection sampling methods (e.g. Devroye [22]) used in most Monte Carlo implementations for the creation of non-uniform variates are not admissible for QMC-techniques as they induce discontinuous functions and disadvantageous discrepancy effects (e.g. Jäckel [45]). To solve this problem, Hlawka and Mück [41, 42, 43] proposed a systematic approach to generate non-uniform low discrepancy (double) sequences. Since the calculation of low discrepancy (double) sequences using Hlawka-Mück techniques is computationally expensive for many distributions, we will finally investigate variation reduction methods to increase the efficiency of our algorithms.

8.2 Financial market model

We assume a Black-Scholes type model with one bank account B, which compounds continuously with (constant) interest rate r, i.e. $B_t = B_0 e^{rt}$ and one stock S, which is driven by a Lévy process Z_t on $(\Omega, \mathcal{A}, P, \mathcal{F})$,

$$S_t = S_0 e^{Z_t}. \tag{8.2}$$

Lévy processes can be characterized by the distribution of Z_1. Various models of this type have been proposed in the last years. Typical examples for the distribution of Z_1 are hyperbolic (e.g. [25]), normal inverse Gauss, variance gamma, and Meixner distributions. For a comprehensive monograph on so-called Non-Gaussian-Black-Merton-Scholes-Theories we refer to [13].
According to the fundamental theory of asset pricing [21], the value C_t of an S-derivative at time t is given by

$$C_t = e^{-r(T-t)} \mathbb{E}^Q \left[C_T(S) | \mathcal{F}_t \right],$$

where $C_T(S)$ is the so-called payoff of the derivative, which in this setting coincides with its value at time T, $\mathcal{F}_t$ denotes the filtration induced by Z_t, and Q stands for an equivalent martingale measure (i.e. Q is equivalent to P and $S_t = e^{-r(l-t)} \mathbb{E}^Q [S_l | \mathcal{F}_t]$ for all $l \geq t$.). Here we always consider the measure obtained by the Esscher transforms (e.g. [32]) as these preserve the type of the distribution of Z_1 in our cases.
In the following we consider the the evaluation of so-called (discrete sampled) Asian options, the simplest type of options, which has to be valued by simulation. The payoff of an Asian

call is defined by

$$C_T = \left(\frac{1}{m} \sum_{i=1}^{m} S_{t_i} - K \right)^+ = \max \left\{ \frac{1}{m} \sum_{i=1}^{m} S_{t_i} - K, 0 \right\},$$

with $0 \leq t_1 < t_2 < \ldots < t_m \leq T$. The constant factor $K \geq 0$ is called the strike price.

Therefore we get the following integration problem:

$$I = \int_{\mathbb{R}^m} \underbrace{\left(\frac{S_0}{m} \sum_{i=1}^{m} e^{\sum_{j=1}^{i} x_j} - K \right)^+}_{p(\mathbf{x})} \mathrm{dH}(\mathbf{x}), \tag{8.3}$$

where $H(\mathbf{x}) = H_1(x_1) \cdot \ldots \cdot H_m(x_m)$ and $H_i(x_i)$ denotes for $i = 1, \ldots, m$ the distribution of the so-called log returns induced by Z_1. It should be observed that we have a problem of the same type as in (8.1), only that the singularities appear on the upper integration boundary, i.e. as at least one $x_i \to \infty$. Furthermore, it will turn out useful that the distribution and the density functions can be factored into one-dimensional components.

8.3 Weighted Koksma-Hlawka inequality

To recall Hlawka's integration error bound (respectively a slight generalization for weighted integration) we need some more definitions:

Definition 8.1. The H-discrepancy of $\omega = (y_1, y_2, \ldots)$ measures the distribution properties of the sequence. It is defined as

$$D_{N,H}(\omega) = \sup_{J \subseteq K} \left| \frac{1}{N} A_N(J, \omega) - H(J) \right|,$$

where A_N counts the number of elements in $(y_1, \ldots, y_N)$ falling into the interval J, i.e. $A_N(J, \omega) = \sum_{n=1}^{N} \chi_J(y_n)$.

Theorem 8.1 (Koksma-Hlawka Inequality). *Let f be a function of bounded variation (in the sense of Hardy and Krause, see e.g. [53]) on $[\mathbf{a}, \mathbf{b}]$ and $\omega = (y_1, y_2, \ldots)$ a sequence on $[\mathbf{a}, \mathbf{b}]$. Then the QMC integration error can be bounded by*

$$\left| \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x}) \, \mathrm{dH}(\mathbf{x}) - \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{y_n}) \right| \leq V(f) D_{N,H}(\omega). \tag{8.4}$$

For unbounded functions the Hardy-Krause variation is infinite, hence (8.4) does not lead to an useful bound on the integration error. In the following sections we describe several QMC algorithms for the problem (8.3) and investigate in how far they are beneficial.

8.4 Transformation by inversion

As for most (even one-dimensional) probability measures the distribution function F is not given explicitly, the classical inversion method is in general very inefficient (e.g. if one obtains F^{-1} through backtracking and numerical integration). Nevertheless if the inverse distribution

function H_i^{-1} or approximations thereof are available, the substitution $H_i^{-1}(1-u_i) = x_i$ in equation (8.3) leads to

$$\int_{[\mathbf{0},\mathbf{1}]} \underbrace{\left(\frac{S_0}{m} \sum_{i=1}^{m} e^{\sum_{j=1}^{i} H_j^{-1}(1-u_j)} - K \right)^+}_{f(\mathbf{u})} \mathrm{d}\mathbf{u} \,, \tag{8.5}$$

where the integrand f is still unbounded. To show the convergence of the QMC algorithm we apply the following theorem:

Theorem 8.2 (Convergence of the QMC estimator, [37]). *Let $f(\mathbf{x})$ be a function on $[\mathbf{a},\mathbf{b}]$ with singularities only at the left boundary of the definition interval (i.e. $f(\mathbf{x}) \to \pm\infty$ only if $x_j \to a_j$ for at least one j) and $\omega = (\mathbf{y}_1, \mathbf{y}_2, \ldots)$ be a sequence on $[\mathbf{a},\mathbf{b}]$. Furthermore let $c_{N,j} = \min_{1\le n\le N} y_{n,j}$ and $a_j < c_j \le c_{N,j}$. If the improper integral (8.1) exists in the sense of [37], and if*

$$D_{N,H}(\omega)\, V_{[\mathbf{c},\mathbf{b}]}(f) = o(1), \tag{8.6}$$

then the QMC estimator converges to the value of the improper integral:

$$\lim_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} f(\mathbf{y}_n) = \int_{[\mathbf{a},\mathbf{b}]} f(\mathbf{x})\, \mathrm{dH}(\mathbf{x}) \,.$$

To establish the convergence we thus have to estimate the order of $\mathbf{c_N}$, the minimal element of the used sequence, and the variation of f on $[\mathbf{cq},\mathbf{1}]$.

Lemma 8.3.

- *Let $\omega = (x_1, x_2, \ldots)$ be a Van der Corput sequence in base p (e.g. [53]). Then for the minimal element $c_N^p = \min_{1\le i\le N} x_i$ hold the inequalities*

$$N^{-1} \ge c_N^p = p^{-\lfloor \log_p N \rfloor - 1} \ge (Np)^{-1}.$$

- *For all $(0,m)$-sequences (e.g. [53]) with $\mathbf{x}_0 = \mathbf{0}$ bounds for the minimum*

$$c_N = min_{1\le i\le N, 1\le j\le m}\, x_{i,j}$$

are given by

$$\frac{2}{N} \ge c_N \ge \frac{1}{pN}.$$

Proof. Let p be the base of the Van der Corput sequence. Then x_n is defined by its p-adic expansion $n = \sum_{i=0}^{\infty} a_{i,p}^{(n)} p^i$ and $x_n = \sum_{i=0}^{\infty} a_{i,p}^{(n)} p^{-i-1}$. We have $a_{i,p}^{(n)} = 0$ for all $i > \log_p n$, and at least one $a_{i,p}^{(n)}$ is strictly positive for $i = 1, \ldots, \lfloor \log_p n \rfloor$. Therefore $x_n \ge p^{-(\log_p n)-1}$, with equality if $\lfloor \log_p n \rfloor = \log_p n$.
Consider a $(0,l,m)$-net in base p. Then we have per definition (see e.g. [53]): Any elementary interval with measure p^{-l} contains exactly one element of the net. For $i = 1, \ldots, m$ the interval $J_i = [0, p^{-l}) \times \prod_{j=1, j\ne i}^{m} [0,1]$ is elementary with $\lambda(J) = p^{-l}$. The element $\mathbf{x}_0$ lies in all of these intervals, so no other point $\mathbf{x}_i, (i = 1, \ldots, p^l)$ is in the interval $J = \prod_{i=1}^{m} [0, p^{-l})$. Now let $l = \lfloor \log_b n \rfloor + 1$. For a $(0,m)$-sequence the point set $\mathbf{x}_i, 0 \le i < b^l$ has to be a $(0,l,m)$-net. So $c_n \ge p^{-\lfloor \log_b n \rfloor - 1}$. The same argument shows that there has to be at least one point in $[p^{-l}, 2p^{-l}) \times [0,1)^{m-1}$. Thus it follows that $c_n \le 2p^{-\lfloor \log_b n \rfloor - 1}$. □

The order of the minimal element c_N of a Sobol sequence was given in [64] by $\Theta(1/N)$. The preceding lemma establishes the same order also for Halton and Faure sequences.

Henceforth we will use the notation $\mathbf{c_N} = c_N\mathbf{1}$. Let us estimate the variation $V_{[\mathbf{c_N},\mathbf{1}]}(f)$. We denote by H_λ the one-dimensional double-exponential distribution with parameter $\lambda > 1$, i.e. $h_\lambda(x) = \lambda/2\exp(-\lambda|x|)$. Then

$$H_\lambda^{-1}(x) = \begin{cases} \frac{1}{\lambda}\log(2x) & \text{if } x \le \frac{1}{2}, \\ -\frac{1}{\lambda}\log(2-2x) & \text{if } x > \frac{1}{2}. \end{cases}$$

The variation in the sense of Hardy and Krause can always be bounded by 2^m-times the Vitaly variation (see e.g. [53]) of the whole function. Since f is monotone in all variates, continuous and positive, the Vitaly variation can be calculated as the difference of the function value at the end points:

$$V_{[\mathbf{c_N},\mathbf{1}]}(f) \le 2^m \left(\frac{S}{m} \sum_{i=1}^{m} \prod_{j=1}^{i} (2c_N)^{-1/\lambda} - K \right) \tag{8.7}$$

Theorem 8.4. *Let H_λ be the double-exponential distribution ($\lambda > m$). The QMC algorithm for Sobol, Halton and Faure sequences for the integration problem* (8.5) *with $H(\mathbf{x}) = \prod_{i=1}^m H_\lambda(x_i)$ converges with order*

$$\mathcal{O}\left(\frac{\log^m N}{N^{1-m/\lambda}}\right).$$

Proof. Following [64] and [37] the integration error can be bounded by

$$\begin{aligned} &\left| \frac{1}{N}\sum_{n=1}^{N} f(\mathbf{y}_n) - \int_{[\mathbf{0},\mathbf{1}]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right| \\ &\qquad \le \left| \frac{1}{N}\sum_{n=1}^{N} f(\mathbf{y}_n) - \int_{[\mathbf{c_N},\mathbf{1}]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right| + \left| \int_{I^m\setminus[\mathbf{c_N},\mathbf{1}]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right| \\ &\qquad \le |[1-\lambda([\mathbf{c_N},\mathbf{1}))]\, f(\mathbf{1})| + D_N(\omega) V_{[\mathbf{c_N},\mathbf{1}]}(f) + \left| \int_{I^m\setminus[\mathbf{c_N},\mathbf{1}]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right|. \end{aligned} \tag{8.8}$$

$f(\mathbf{1})$ is zero and therefore also the first summand. For the second summand we obtain using (8.7), Lemma 8.3, and $D_N(\omega) = \mathcal{O}(\log^m N/N)$:

$$D_N(\omega) V_{[\mathbf{c_N},\mathbf{1}]}(f) \le \mathcal{O}\left(\frac{\log^m N}{N} \sum_{i=1}^{m} \prod_{j=1}^{i} N^{1/\lambda} \right) \le \mathcal{O}\left(\frac{\log^m N}{N^{1-m/\lambda}} \right).$$

Finally, we analyze the convergence order of the third term. Therefore we divide the interval $I = [0,1]$ in the three (disjoint) subsets $A_1 = [0, c_N)$, $A_2 = [c_N, 1/2)$ and $A_3 = [1/2, 1]$. This induces a partition of I^m in 3^m subsets $I_j = \bigotimes_{s=1}^m A_{j,s}$, where $A_{j,s}$ is A_i if the projection of I_j on the s-dimension coincides with A_i. Furthermore let $l_{j,i}, (i = 1,2,3)$ be the number A_i-elements in the product.

Now we define $\mathcal{A}$ as the set of all I_j that lie in $I^m \setminus [\mathbf{c_N},\mathbf{1}]$. So we get

$$\left| \int_{I^m\setminus[\mathbf{c_N},\mathbf{1}]} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right| = \sum_{I_j\in\mathcal{A}} \left| \int_{I_j} f(\mathbf{x})\,\mathrm{d}\mathbf{x} \right|$$

Now define the function

$$B(x_i) = \begin{cases} x_i^{-1/\lambda} & \text{if } x_i \in A_1 \cup A_2, \\ 1 & \text{else,} \end{cases}$$

so that we get the inequality $f(\mathbf{x}) \leq \bar{C} \prod_{i=1}^m B(x_i)$ for some constant $\bar{C}$. This yields

$$\left| \int_{I_j} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} \right| \leq \mathcal{O}\left(\left(\int_0^{c_N} x^{-1/\lambda} \, \mathrm{d}x \right)^{l_{1,j}} \left(\int_{c_N}^{1/2} x^{-1/\lambda} \, \mathrm{d}x \right)^{l_{2,j}} \frac{1}{2^{l_{3,j}}} \right) \leq \mathcal{O}\left(c_N^{(1-1/\lambda)l_{1,j}} \right).$$

For the Halton, Faure and Sobol sequences we get with the preceding lemma $c_N = \Theta(1/N)$ and finally with $l_{1,j} > 0$ for $I_j \in \mathcal{A}$:

$$\left| \int_{I^m \setminus [\mathbf{c_N}, \mathbf{1}]} f(\mathbf{x}) \, \mathrm{d}\mathbf{x} \right| = \mathcal{O}\left(N^{-(1-1/\lambda)} \right) \leq \mathcal{O}\left(\frac{\log^m N}{N^{1-m/\lambda}} \right).$$

□

This result can be generalized to all distributions, which have lighter tails than the double-exponential distribution. The following obvious lemma will provide the base for the generalization.

Lemma 8.5. *If there exist an $x_0 \in \mathbb{R}$ and two densities f_1 and f_2 on $\mathbb{R}$ such that $f_1(x) \geq f_2(x)$ for all $x \geq x_0$, then the inverse distribution functions act in the same manner $F_1^{-1}(x) \geq F_2^{-1}(x)$ for all $x \geq x_0$.*

Theorem 8.6. *For all distributions $H(\mathbf{x}) = H_1(x_1)H_2(x_2)\cdots H_m(x_m)$ with the property that there exist a $\lambda > m$ and an x_0 for all $1 \leq i \leq m$ such that $H_i(x) < H_\lambda(x)$ for all $x > x_0$, the convergence order of the QMC algorithm with inversion method using Sobol, Halton or Faure sequences for the problem (8.5) is*

$$\mathcal{O}\left(\frac{\log^m N}{N^{1-m/\lambda}} \right).$$

Proof. Without loss of generality we assume $H_i(x_i) = F(x_i)$ for $1 \leq i \leq m$. Again we take inequality (8.8). The first term equals zero. If c_N is small enough we have $F^{-1}(c_N) < H_\lambda^{-1}(c_N)$ so that the second term can be bounded like in the proof of Theorem 8.4. For the third term we divide I in $A_1 = [0, c_N]$, $A_2 = (c_N, \bar{x}]$, and $A_3 = (\bar{x}, 1]$, where $\bar{x}$ is the solution of $F(x) = 0$. We define

$$B(x_i) = \begin{cases} e^{F^{-1}(x_i)} & \text{if } x_i \in A_1 \cup A_2 \\ 1 & \text{else} \end{cases}$$

So $f(\mathbf{x}) < \prod_{i=1}^m B(x_i)$ and with the same notations as before and the observation that $e^{H_\lambda^{-1}(x)} > e^{F^{-1}(x)}$ in A_1, we finish

$$\left| \int_{I_j} f(\mathbf{x}) \, \mathrm{d}x \right| \leq \mathcal{O}\left(\left(\int_0^{c_N} x^{-1/\lambda} \, \mathrm{d}x \right)^{l_{1,j}} \left(\int_{c_N}^{\bar{x}} x^{-1/\lambda} \, \mathrm{d}x \right)^{l_{2,j}} (1-\bar{x})^{l_{3,j}} \right) \leq \mathcal{O}\left((c_N)^{(1-1/\lambda)l_{1,j}} \right)$$

The convergence order is now calculated analogous to Theorem 8.4. □

8.5 Other integral transformations

The inversion method is a rather special integral transformation and seldom available. Instead of the inverse distribution function one can try other functions, which might be easier to implement. Additionally, if the tails are heavy enough, one can force the integrand to be of bounded variation. Such integral transforms are closely connected to importance sampling methods.
Let G be a distribution on $\mathbb{R}^m$ with density $g(\mathbf{x}) = g_1(x)g_2(x)\cdots g_m(x)$, then we can transform the integral (8.3) to

$$\int_{[\mathbf{0},\mathbf{1}]} \left(\frac{S_0}{m} \sum_{i=1}^{m} e^{\sum_{j=1}^{i} G_j^{-1}(1-x_j)} - K \right)^+ \prod_{i=1}^{m} \frac{h_i(G_i^{-1}(1-x_i))}{g_i(G_i^{-1}(1-x_i))} \, \mathrm{dG}(\mathbf{x}), \tag{8.9}$$

To obtain a small variation, the distribution G should have the following properties: $g(\mathbf{x}) > 0$ for all $\mathbf{x} \in \mathbb{R}^m$ and the density g should have a similar shape as $|f(\mathbf{x})h(\mathbf{x})|$. Here we consider two *transformation distributions*: on the one hand the (shifted) double-exponential distribution with $\lambda_i = \mathrm{Var}[H_i]$ and $\mu = \mathrm{argmax}_{\mathbf{x}\in\mathbb{R}^m} p(\mathbf{x}) \prod_{i=1}^{m} h_i(x_i)$, where h_i denotes the density of H_i and on the other hand the (rescaled) hat function of the ratio of uniforms method (e.g [22]). Both distributions are computationally easy to handle and have tails such that the integrand is of bounded variation. So asymptotically these algorithms perform better than the inversion method algorithms.

8.6 The Hlawka-Mück method

In the previous sections we transformed the integral to the unit interval using a suitable density to be able to directly apply Quasi-Monte Carlo methods with respect to the uniform distribution. Such a procedure, however, might lead to practical problems, for example if the resulting product term in (8.9) is sharply peaked, which might happen for some parameter choices. In these cases, one cannot expect uniformly distributed QMC methods to lead to acceptable results. In [43] and [42] Hlawka and Mück proposed a method for the direct creation of F-distributed low-discrepancy sequences, where F is a distribution with support I^m and density f.
In our case (see (8.3)), the dimensions of the H-distributed m-dimensional variates are independent and thus the density can be factored into a product of one-dimensional densities. Consequently we apply the simpler construction:

Lemma 8.7 (Hlawka, [41]). *Let $F(\mathbf{x}) = F_1(x_1) \cdot \ldots \cdot F_m(x_m)$ denote a cumulative distribution function with density $f(\mathbf{x})$ defined on I^m and $M_f = \sup f(\mathbf{x})$. Let furthermore $\omega = (\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x_N})$ be a sequence in I^m with discrepancy $D_N(\omega)$. Then the point set $\tilde{\omega} = (\tilde{\mathbf{y}}_\mathbf{1}, \ldots, \tilde{\mathbf{y}}_\mathbf{N})$ with*

$$\tilde{y}_{k,j} = \frac{1}{N} \sum_{r=1}^{N} \lfloor 1 + x_{k,j} - F_j(x_{r,j}) \rfloor = \frac{1}{N} \sum_{r=1}^{N} \chi_{[0,x_{k,j}]}(F_j(x_{r,j})) \tag{8.10}$$

has an F-discrepancy of

$$D_{N,F}(\tilde{\omega}) \leq (1 + 3mM_f) D_N(\omega) \, .$$

As the H_i-distributions in problem (8.3) typically have unbounded support, we need to transform them to the unit interval, generate low-discrepancy sequences there, and then convert them back to H_i-distributed low-discrepancy sequences on $\mathbb{R}$. Therefore we write the densities as $h_i(x_i) = h_i(H_\lambda^{-1}(H_\lambda(x_i)))$, where H_λ denotes again the (one-dimensional) exponential distribution with parameter λ. Denoting $\bar{g}(y) = h_i(H_\lambda^{-1}(y))$ we generate for each dimension sequences $\tilde{\omega}_i$ on I which are distributed with density $\bar{g}(y)$ using the algorithm of Lemma 8.7. Then we transform them back to $\mathbb{R}$ using the distribution function of the double-exponential distribution. Since a transformation using a quantile function of any distribution F preserves the discrepancy, i.e. $D_N(\omega) = D_{N,F}(F^{-1}(\omega))$, this last step can be justified, and the discrepancy is not affected. We obtain H_i-distributed low-discrepancy sequences $\bar{\omega}_i = H_\lambda^{-1}(\tilde{\omega}_i)$ by this process.
The H-discrepancy of the sequence $\bar{\omega}$, generated by pasting the dimensions $\bar{\omega}_i$ together again, can be estimated as

$$D_{N,H}(\bar{\omega}) = D_{N,\bar{G}}(\tilde{\omega}) \leq (1 + 3mM_{\bar{g}})D_N(\omega)$$

using the uniform discrepancy of the original sequence.
Once we have these H-distributed variates $\bar{\omega} = (\mathbf{y}_i)_{i\geq 1}$, the integral can be estimated as

$$\int_{\mathbb{R}^m} p(\mathbf{x})\,\mathrm{d}H(\mathbf{x}) \approx \frac{1}{N}\sum_{i=1}^{N} p(\mathbf{y}_i). \tag{8.11}$$

Theorem 8.8. *Using the Hlawka-Mück-type method as outlined above to create H-distributed variates from Sobol, Halton or Faure sequences, the convergence order of the direct QMC algorithm* (8.11) *for the improper integration problem $\int_{\mathbb{R}^m} p(\mathbf{x})\,\mathrm{d}H(\mathbf{x})$ can be bounded by*

$$\mathcal{O}\left(\frac{\log^m N}{N^{1-m/\lambda}}\right).$$

Here λ denotes the parameter of the double-exponential distribution used to transform to the interval I and back.

Proof. In the original integral in (8.11), the integrand has a singularity for $x_i \to \infty$, i.e. at the upper integration boundary. In that case, a theorem similar to Theorem 8.2 holds with the upper and lower boundaries exchanged, and

$$\tilde{c}_{N,j} = \max_{1\leq n\leq N} y_{n,j}, \qquad \tilde{c}_{N,j} \leq \tilde{c}_j < a_j.$$

Also, the error bound (8.8) holds with the same changes:

$$\|E\| \leq |1 - H(\tilde{\mathbf{c}})|\,|p(-\infty)| + D_{N,H}(\bar{\omega})V_{[-\infty,\tilde{\mathbf{c}})}(p) + \int_{\mathbb{R}^m\setminus(-\infty,\tilde{\mathbf{c}}]} p(\mathbf{x})\,\mathrm{dH}(\mathbf{x}) \tag{8.12}$$

We have

$$p(-\infty) = 0, \quad |1 - H(\tilde{\mathbf{c}})| \leq 1, \quad D_{N,H}(\bar{\omega}) \leq (1 + 3mM_{\bar{g}})D_N(\omega)\ .$$

Also, the payoff function is monotone in all variables and continuous, so we choose $\tilde{c}_i = H_\lambda^{-1}(c_{N,i}) = -\frac{1}{\lambda}\log(2 - 2c_{N,i})$ for N and thus $c_{N,i}$ and $\tilde{c}_i$ sufficiently large. Then we obtain

$$V_{[-\infty,\tilde{\mathbf{c}})}(p) \leq 2^m\,(p(\tilde{\mathbf{c}}) - p(-\infty)) = 2^m p(\tilde{\mathbf{c}}).$$

For c_i sufficiently large, the $(\dots)^+$-Operator in $p(\mathbf{x})$ can also be omitted, and using the bounds $1-\frac{1}{N} \le c_{N,j} \le 1-\frac{1}{\alpha N}$ (with a suitable constant α) for the Sobol, Halton and Faure sequences leads to

$$V_{[-\infty,\tilde{\mathbf{c}})}(p) \le 2^m \left(\frac{S_0}{m} \sum_{i=1}^{m} \left(\prod_{j=1}^{i} \frac{1}{(2\alpha N)^{-\frac{1}{\lambda}}} \right) - K \right) = \mathcal{O}\left(\frac{1}{N^{-\frac{m}{\lambda}}} \right) .$$

The order of the remaining term $I_{\text{rest}} = \int_{\mathbb{R}^m \setminus [-\infty,\tilde{\mathbf{c}})} p(\mathbf{x})\, \mathrm{d}H(\mathbf{x})$ was already determined as $\mathcal{O}\left(N^{-(1-1/\lambda)}\right)$ in Theorem 8.6, because

$$\left| \int_{\mathbb{R}^m \setminus [-\infty,\tilde{\mathbf{c}})} p(\mathbf{x})\, \mathrm{dH}(\mathbf{x}) \right| = \left| \int_{I^m \setminus [\mathbf{0},\mathbf{1}-\mathbf{c}_N)} f(1-\mathbf{x})\, \mathrm{d}\mathbf{x} \right|$$
$$= \left| \int_{I^m \setminus [\mathbf{c}_N,\mathbf{1})} f(\mathbf{x})\, \mathrm{d}\mathbf{x} \right| = \mathcal{O}\left(N^{(1-\frac{m}{\lambda})} \right) .$$

Putting all these terms together, the second term in (8.12) has the largest order in N of $\mathcal{O}\left((\log^m N)/N^{1-m/\lambda}\right)$, which proves the theorem. □

8.7 Importance sampling

As the generation of Hlawka-Mück-(double)-sequences is rather expensive, we try to improve the convergence speed of our algorithms through importance sampling. The idea is to use the properly shifted original distribution H_i as transformation distribution G_i in equation (8.9). Properly means in this case that we shift the distribution H_i such that its mean coincides with the i-th coordinate of the argmax of the product payoff times density, i.e. $g_i(x) = h_i(x-(m_i-\mu_i))$, where $m = \text{argmax}_{\mathbf{x}\in\mathbb{R}^m} p(\mathbf{x}) \prod_{i=1}^m h_i(x_i)$ and $\mu_i = \mathbb{E}^Q[H_i]$. This procedure was inspired by a method proposed by Glasserman et. al in [33] when valuing Asian options in a Gaussian framework and successfully adapted for the hyperbolic case in [38]. Here it will be combined with Hlawka-Mück methods.

8.8 Numerical results

We will now investigate the various numerical methods presented in the previous chapters. As an example inspired from a real share we use an Asian option on a stock with normal inverse Gauss-distributed log increments (i.e. Z_1 in (8.2) has density $f_{NIG}(x)$) using parameters as given in Table 8.1.

$$f_{NIG}(x) = \frac{\alpha}{\pi} \exp\left(\delta\sqrt{\alpha^2-\beta^2} + \beta(x-\mu)\right) \frac{\delta \mathrm{K}_1(\alpha\sqrt{\delta^2+(x-\mu)^2})}{\sqrt{\delta^2+(x-\mu)^2}},$$

where $\mathrm{K}_1(x)$ denotes the modified Bessel function of third type of order 1 (Macdonald function).

We sample the increments of the option value in weekly intervals, with the options having a maturity of one, two, and three months (4, 8, and 12 weeks). As a result, our problem is a 4-, 8-, and 12-dimensional integral over the payoff function. In our calculations, we use a Mersenne twister as random number generator, and the Sobol sequence for all QMC integration methods, with the exception of the Hlawka-Mück method, which employs the

$\alpha = 136.29$	$\beta = -15.1977$	$\delta = 0.0059 \cdot 5$	$\mu = 0.00079 \cdot 5$
$r = 0.0375$	250 days per year		
$\mathbb{E}[H_i] = 0.00063981$	$\sigma[H_i] = 0.014851$	$\lambda = 95.2271$	

Table 8.1: *Parameters of the NIG distribution (under the martingale measure obtained by Esscher transform) and the option, as well as the resulting moments of the NIG and double-exponential distributions. As the NIG distribution is stable under convolution, we sample the increments weekly and use a factor* 5 *for* δ *and* μ*.*

Halton-sequence. Calculations using the Sobol or Faure sequence show no significant difference (although the Faure sequence might be interpreted as slightly inferior compared to the others), so we will omit their results in the sequel.
Figure 8.1 shows log-log plots of the relative error for some of the methods discussed earlier, Table 8.2 gives the corresponding numerical values. In particular, we compare several QMC methods to a crude Monte Carlo estimator (using acceptance-rejection from pseudo-random numbers to generate NIG-distributed variates). Independent of the dimension of the integration problem, this crude Monte Carlo approach yields the worst results for all N. Some improvement can be achieved by using the importance sampling idea discussed in Section 8.7, where the mean of the NIG-distributed variates is shifted to the maximum of payoff times density function.
In contrast to these two Monte Carlo methods, we apply QMC methods using an integral transformation. We give two examples of such a transformation: On one hand, we approximate the NIG distribution by the hat function of the ratio-of-uniforms method, and on the other hand we use the double exponential distribution. Both distributions have the advantage that the distribution function and its inverse are explicitly known, so the integrand in (8.9) can be easily calculated. In these two cases, we employ the Sobol sequence as low-discrepancy sequences.
Finally, we directly generate NIG-distributed low-discrepancy sequences with the Hlawka-Mück method described in Section 8.6 using the Halton sequence. Additionally, we also employ the importance sampling of Section 8.7 in an attempt to improve the results even more.

As seen in the figure, the direct Monte Carlo method (crude MC and importance sampling) cannot compete with the QMC methods. While the transformation using the ratio-of-uniforms function gives excellent results in four dimensions, it looses its good features in 12 dimensions. This is probably a consequence of the fact that the multi-dimensional hat function cannot approximate the density so well any more, and the numeric calculation of the needed parameters is relatively error-prone. The double-exponential distribution displays a similar problem, where for smaller values of N it does not seem to have any advantage over crude Monte Carlo. Only for larger numbers of points it is able to gain advantage, an effect which gets even more pronounced as the number of dimensions increases.
Finally, the Hlawka-Mück transformation from the Halton sequence to NIG-distributed low-discrepancy sequences can give considerable improvement over the other methods presented. While in four dimensions all QMC algorithms have similar errors, in 12 dimensions only the Hlawka-Mück transformation leads the other methods by some orders of magnitude. Although the generation of the NIG-distributed sequences is numerically quite expensive due to the N^2

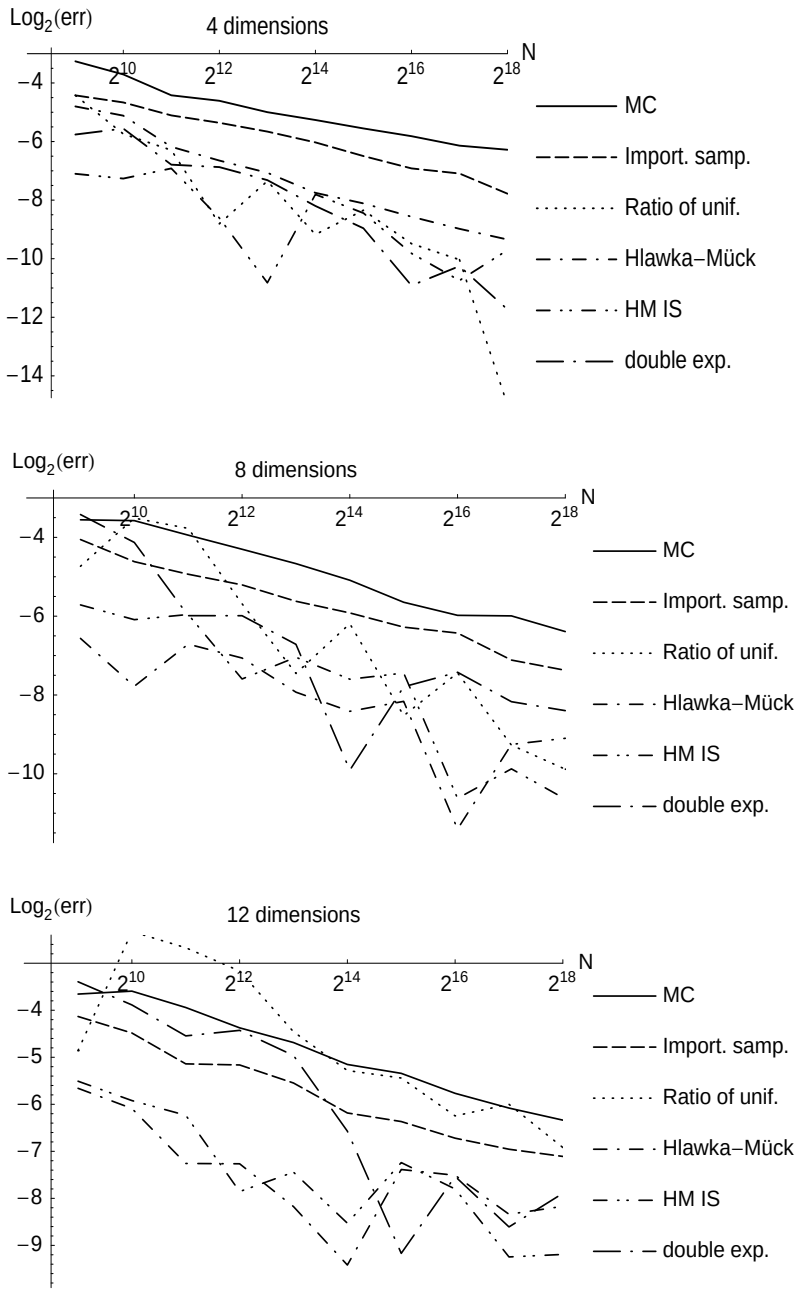

Figure 8.1: *Relative errors of the simulations in 4, 8, and 12 dimensions. Monte Carlo results are average errors of 25 independent runs, all QMC results come from a single run.*

4 dimensions	32	128	512	2048
Monte Carlo	0.12072217	0.06147889	0.03838273	0.01202358
Importance sampling	0.05876176	0.02490847	0.01199762	0.0060517
Ratio of uniforms	0.0766848	0.03733064	0.01201625	0.00184711
Hlawka-Mück	0.06527011	0.01074534	0.00822849	0.00207232
HM Import. sampling	0.0112777	0.01467202	0.00082417	0.00099044
Double exponential	0.06153541	0.0010981	0.00314842	0.00112535
4 dimensions	4096	16384	65536	262144
Monte Carlo	0.00996698	0.00517292	0.00297738	0.00187234
Importance sampling	0.00470025	0.00240813	0.00099072	0.00041771
Ratio of uniforms	0.00014952	0.00010376	0.00007589	0.00000031
Hlawka-Mück	0.00129508	0.00043016	0.00019038	0.00008742
HM Import. sampling	0.00018239	0.00040792	0.0000545	0.00006087
Double exponential	0.00103612	0.00027488	0.00001819	0.00000803

8 dimensions	32	128	512	2048
Monte Carlo	0.12370532	0.06464402	0.02860497	0.01937286
Importance sampling	0.05797634	0.02903363	0.017333	0.00718672
Ratio of uniforms	0.19733215	0.02058577	0.00879434	0.02300414
Hlawka-Mück	0.18544944	0.05211664	0.00141103	0.00121882
HM Import. sampling	0.03022395	0.00752137	0.00329977	0.00258743
Double exponential	0.09533048	0.03148169	0.03270188	0.00251103
8 dimensions	4096	16384	65536	262144
Monte Carlo	0.01354622	0.00617114	0.00253223	0.00167647
Importance sampling	0.00548498	0.00269138	0.00161492	0.0006278
Ratio of uniforms	0.00339376	0.00204859	0.00058707	0.00005102
Hlawka-Mück	0.00085671	0.0002217	0.00001117	0.00011178
HM Import. sampling	0.00050351	0.0005005	0.00002434	0.0000239
Double exponential	0.0025084	0.0000498	0.00059723	0.00022552

12 dimensions	32	128	512	2048
Monte Carlo	0.12688204	0.06009656	0.02583161	0.01941282
Importance sampling	0.04667326	0.02706982	0.01604223	0.00585588
Ratio of uniforms	1.3300692	0.29538864	0.0077653	0.06920285
Hlawka-Mück	0.19649027	0.0850785	0.00348351	0.00070439
HM Import. sampling	0.100306	0.02977137	0.00405115	0.00196351
Double exponential	0.14756909	0.01798477	0.0335691	0.01061674
12 dimensions	4096	16384	65536	262144
Monte Carlo	0.01259962	0.00578311	0.00312989	0.00176997
Importance sampling	0.00572169	0.00206127	0.0012058	0.00081798
Ratio of uniforms	0.04219506	0.00508302	0.00193717	0.00098897
Hlawka-Mück	0.00069954	0.00008126	0.00054785	0.00028312
HM Import. sampling	0.00038741	0.00019782	0.00040928	0.00010181
Double exponential	0.01194394	0.00139974	0.0005346	0.00037327

Table 8.2: *Numerical values of the relative errors in 4, 8, and 12 dimensions. The exact value is* 0.898087 *in 4 dimensions,* 1.22436 *in 8 dimensions, and* 1.50384 *in 12 dimensions.*

effort in the double sum in (8.10), only a very small number of points is needed to achieve an error of a size which is obtained by crude Monte Carlo with about 100 times as many points. For such small values of N, even the generation of the Hlawka-Mück variates is not too expensive and can be considered competitive. Interestingly, importance sampling cannot improve the quality of the Hlawka-Mück results any more. In contrary, in eight dimensions, the error even increases when one tries to employ importance sampling to the NIG-distributed sequences.

8.9 Conclusion

In this chapter we successfully showed that the evaluation of the integral (8.3) can be considerably improved compared to crude Monte Carlo evaluation. Several integral transformations and subsequent application of QMC integration techniques are of advantage, especially in low-dimensional problems. However, the direct generation of NIG-distributed low-discrepancy sequences using Hlawka and Mück's idea provided the best basis – especially in higher dimensions – to evaluate the problem with excellent results. Although this generation is rather expensive, it only needs a fraction of the number of points needed for the other methods.
Here we only looked at two specific integral transformations, namely the double-exponential distribution function, and the hat function inspired by the ratio-of-uniforms transformation. An interesting question to be investigated in the future is the search for better or even optimal integral transformations for improper integration problems like the one discussed in this article. A similar question is the choice of transformation for the Hlawka-Mück generation of low-discrepancy sequences.

Acknowledgments

At this place I would like to thank my thesis advisor o.Univ.-Prof. Dr. Robert F. Tichy for his help and his support, be it through encouragement, tips, and discussions, but also through his FSP project. But most of all I have to thank him for the casual, but very productive atmosphere in which I was able to work.

Furthermore, I would like to thank several colleagues, namely Dr. Hansjörg Albrecher, Dipl.-Ing. Jürgen Hartinger, and Dr. Martin Predota. Not only was it a lot of fun sharing an office with them, but also a very fruitful collaboration which has this thesis as one of its results.

Of course, also have to thank all the other colleagues at the Department of Mathematics for providing such a good environment there.

Last, but certainly not least, I want to say thank you to my parents, my whole family and all my friends for supporting me.

List of Figures

List of Tables

Bibliography

[1] J. Aczél. *Vorlesungen über Funktionalgleichungen und ihre Anwendungen.* Birkhäuser, Basel, 1960.

[2] H. Albrecher and R. Kainhofer. Risk Theory with a Nonlinear Dividend Barrier. *Computing*, 68:289–311, 2002.

[3] H. Albrecher, R. Kainhofer, and R. Tichy. Efficient simulation techniques for a generalized ruin moodel. *Grazer Math. Ber.*, 345:79–110, 2002.

[4] H. Albrecher, R. Kainhofer, and R. Tichy. Simulation methods in ruin models with non-linear dividend barriers. *Math. Comput. Simulation*, 62:277–287, 2003.

[5] H. Albrecher and R. Tichy. On the convergence of a solution procedure for a risk model with gamma-distributed claims. *Schweiz. Aktuarver. Mitt.*, 2:115–127, 2000.

[6] A. Alegre, M. Claramunt, and M. Marmol. Dividend policy and ruin probability. *Proceedings of the Third Int. Congress on Insurance: Mathematics & Economics, London*, 1999.

[7] F. Allen, A. Bernardo, and I. Welch. A theory of dividends based on tax clienteles. *Wharton School Working Paper 15-98, Pennsylvania*, 1999.

[8] S. Asmussen. *Ruin probabilities.* World Scientific, Singapore, 2000.

[9] S. Asmussen and K. Binswanger. Simulation of ruin probabilities for subexponential claims. *ASTIN Bull.*, 27(2):297–318, 1997.

[10] R. Bali and A. Hovakimian. Taxes and stock returns. *Journal of Financial Education*, 25:14–20, 1999.

[11] R. Bellman. On the Computational Solution of Differential-Difference Equations. *J. Math. Anal. and Appl.*, 2:108–110, 1961.

[12] P. Boogaert, F. Delbaen, and J. Haezendonck. Macro-economic influences on the crossing of dividend barriers. *Scandinavian Actuarial Journal*, pages 231–245, 1988.

[13] S. I. Boyarchenko and S. Z. Levendorskiĭ. *Non-Gaussian Merton-Black-Scholes theory*, volume 9 of *Advanced Series on Statistical Science & Applied Probability.* World Scientific Publishing Co. Inc., River Edge, NJ, 2002.

[14] P. Bratley, B. L. Fox, and H. Niederreiter. Implementation and tests of low discrepancy sequences. *ACM Trans. Model. Comp. Simul.*, 2:195–213, 1992.

[15] H. Bühlmann. *Mathematical Methods in Risk Theory.* Springer, New York, 1970.

[16] R. Caflisch. Monte Carlo and Quasi-Monte Carlo methods. *Acta Numerica*, pages 1–46, 1998.

[17] I. Coulibaly and C. Lécot. A Quasi-randomized Runge-Kutta Method. *Mathematics of Computation*, 68(226):651–659, 1999.

[18] E. de Doncker and Y. Guan. Error bounds for the integration of singular functions using equidistributed sequences. *Journal of Complexity*, 2003. In press.

[19] B. de Finetti. Su un'impostazione alternativa della teoria collettiva del rischio. *Transactions of the 15th Int. Congress of Actuaries*, 2:433–443, 1957.

[20] F. Delbaen and J. Haezendonck. Classical risk theory in an economic environment. *Insurance: Mathematics and Economics*, 6:85–116, 1987.

[21] F. Delbaen and W. Schachermayer. A general version of the fundamental theorem of asset pricing. *Math. Ann.*, 300(3):463–520, 1994.

[22] L. Devroye. *Nonuniform random variate generation.* Springer-Verlag, New York, 1986.

[23] R. D. Driver. Existence and stability of solutions of a delay-differential system. *Arch. Rational Mech. Anal.*, 10:401–426, 1962.

[24] M. Drmota and R. Tichy. *Sequences, Discrepancies and Applications*, volume 1651 of *Lecture Notes in Mathematics.* Springer, New York, Berlin, Heidelberg, Tokyo, 1997.

[25] E. Eberlein and U. Keller. Hyperbolic distributions in finance. *Bernoulli*, 1:281–299, 1995.

[26] H. Faure. Discrépance de suites associées à un système de numération (en dimension s). *Acta Arithmetica*, 41:337–351, 1982.

[27] H. U. Gerber. Martingales in risk theory. *Mitteilungen der Schweizer Vereinigung der Versicherungsmathematiker*, pages 205–216, 1973.

[28] H. U. Gerber. The dilemma between dividends and safety and a generalization of the Lundberg-Cramér formulas. *Scandinavian Actuarial Journal*, pages 46–57, 1974.

[29] H. U. Gerber. *An Introduction to Mathematical Risk Theory.* Huebner Foundation Monograph 8, Homewood, Ill, 1979.

[30] H. U. Gerber. On the probability of ruin in the presence of a linear dividend barrier. *Scandinavian Actuarial Journal*, pages 105–115, 1981.

[31] H. U. Gerber and E. Shiu. On the time value of ruin. *North American Actuarial Journal*, 2(1):48–72, 1998.

[32] H. U. Gerber and E. S. W. Shiu. Actuarial bridges to dynamic hedging and option pricing. *Insurance Math. Econom.*, 18(3):183–218, 1996.

[33] P. Glasserman, P. Heidelberger, and P. Shahabuddin. Asymptotically optimal importance sampling and stratification for pricing path-dependent options. *Math. Finance*, 9(2):117–152, 1999.

[34] E. Hairer, S. P. Norsett, and G. Wanner. *Solving Ordinary Differential Equations I*, volume 8 of *Springer Series in Computational Mathematics.* Springer, Berlin, Heidelberg, 1987.

[35] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numer. Math.*, 2:84–90, 1960.

[36] J. Hartinger, R. Kainhofer, and M. Predota. Option Pricing as an Unbounded Non-Uniform Quasi-Monte Carlo Integration Problem. 2003. Submitted.

[37] J. Hartinger, R. Kainhofer, and R. Tichy. Quasi-Monte Carlo algorithms for unbounded, weighted integration problems. 2003. Submitted.

[38] J. Hartinger and M. Predota. Pricing Asian option in the hyperbolic model: A fast Quasi-Monte Carlo approach. *Grazer Math. Ber.*, 345:1–33, 2002.

[39] L. Heersink. Ruinmodelle mit linearen und nichtlinearen Dividendenschranken. Diplomarbeit, TU-Graz, Graz, Austria, 2001.

[40] E. Hlawka. Funktionen von beschränkter Variation in der Theorie der Gleichverteilung. *Ann. Mat. Pura Appl. (4)*, 54:325–333, 1961.

[41] E. Hlawka. Gleichverteilung und Simulation. *Österreich. Akad. Wiss. Math.-Natur. Kl. Sitzungsber. II*, 206:183–216, 1997.

[42] E. Hlawka and R. Mück. A transformation of equidistributed sequences. In *Applications of number theory to numerical analysis*, pages 371–388. Academic Press, New York, 1972.

[43] E. Hlawka and R. Mück. Über eine Transformation von gleichverteilten Folgen. II. *Computing*, 9:127–138, 1972.

[44] J. Irbaeck. Asymptotic theory for a risk process with a high dividend barrier. *Licentiate thesis, Royal Institute of Technology Stockholm*, 2000.

[45] P. Jäckel. *Monte Carlo Methods in finance.* John Wiley and Sons, LTD, West Sussex, 2002. Wiley Finance Series.

[46] R. Kainhofer. QMC methods for the solution of delay differential equations. *J. Comp. Appl. Math.*, 2003. In press.

[47] R. Kainhofer and R. Tichy. QMC methods for the solution of differential equations with multiple delayed arguments. *Grazer Math. Ber.*, 345:111–129, 2002.

[48] R. Kainhofer and R. Tichy. RKQMC methods for the solution of delay differential equations. *PAMM, online edition*, 2003. Submitted.

[49] B. Klinger. Numerical integration of singular integrands using low-discrepancy sequences. *Computing*, 59(3):223–236, 1997.

[50] A. A. Koudiraty. *Numerical analysis of some quasi-Monte Carlo Methods.* PhD thesis, Université de Savoie, 2001.

[51] L. Kuipers and H. Niederreiter. *Uniform distribution of sequences.* Wiley-Interscience Publ., New York, 1974. Pure and Applied Mathematics.

[52] C. Lécot. Quasi-randomized numerical methods for systems with coefficients of bounded variation. *Mathematics and Computers in Simulation*, 55:113–121, 2001.

[53] H. Niederreiter. *Random Number Generation and Quasi-Monte Carlo Methods.* Society for industrial and applied mathematics (SIAM), Philadelphia, 1992.

[54] E. Novak and H. Woźniakowski. When are integration and discrepancy tractable? In *Foundations of computational mathematics (Oxford, 1999)*, pages 211–266. Cambridge Univ. Press, Cambridge, 2001.

[55] H. J. Oberle and H. J. Pesch. Numerical Treatment of Delay Differential Equations by Hermite Interpolation. *Numer. Math.*, 37:235–255, 1981.

[56] J. Oppelstrup. The RKFHB4 Method for Delay-Differential Equations. In R. D. G. Roland Burlirsch and J. Schröder, editors, *Numerical Treatment of Differential Equations*, volume 631 of *Lecture Notes in Mathematics*, pages 133–146. Springer, New York, Berlin, Heidelberg, Tokyo, 1976. Proceedings, Oberwolfach, 1976.

[57] S. H. Paskov and J. Traub. Faster Valuation of Financial Derivatives. *Journal of portfolio management*, pages 113–120, 1995.

[58] J. Paulsen and H. Gjessing. Optimal choice of dividend barriers for a risk process with stochastic return on investments. *Insurance Math. Econom.*, 20:215–223, 1997.

[59] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C.* Cambridge University Press, Cambridge, 1992.

[60] I. Radovic, M. Sobol', and R. F. Tichy. Quasi-monte carlo sequences for numerical integration: Comparison of different low discrepancy sequences. *Monte Carlo Methods and Applications*, 2(1):1–14, 1996.

[61] C.-O. Segerdahl. On some distributions in time connected with the collective theory of risk. *Skand. Aktuarietidskr.*, pages 167–192, 1970.

[62] T. Siegl and R. Tichy. Lösungsmethoden eines Risikomodells bei exponentiell fallender Schadensverteilung. *Schweiz. Aktuarver. Mitt.*, 1:85–118, 1996.

[63] I. M. Sobol'. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Comput. Math. Math. Phys.*, 7:86–112, 1967.

[64] I. M. Sobol'. Calculation of improper integrals using uniformly distributed sequences. *Soviet Math. Dokl.*, 14(3):734–738, 1973.

[65] G. Stengle. Numerical methods for systems with measureable coefficients. *Appl. Math. Lett.*, 3:25–29, 1990.

[66] G. Stengle. Error analysis of a randomized numerical method. *Numer. Math.*, 70:119–128, 1995.

[67] O. Thorin. Probabilities of ruin. *Scandinavian Actuarial Journal*, pages 65–102, 1982.

[68] R. F. Tichy. Über eine zahlentheoretische Methode zur numerischen Integration und zur Behandlung von Integralgleichungen. *Sitzungsberichte der Österreichischen Akademie der Wissenschaften*, 193, 1984.

[69] R. F. Tichy. Bemerkung zu einem versicherungsmathematischen Modell. *Mitt. Verein. Schweiz. Versicherungsmath.*, (2):237–241, 1987.

[70] B. Tuffin. *Simulation accélérée par les méthodes de Monte Carlo et quasi-Monte Carlo: théorie et applications.* Thèse de doctorat, Université de Rennes, 1997.

[71] F. D. Vylder. *Advanced risk theory.* Editions de l'universite de Bruxelles, Bruxelles, 1996.