

Numerical Finance

Prof. Dr. Karsten Urban

Universität Ulm
Institut für Numerische Mathematik
Sommersemester 2007

Contents

Preface	4
1 Introduction	5
2 Numerical Generation of Random Numbers	8
2.1 Congruence Methods	9
2.2 Frequency and Gap Tests	11
2.2.1 The χ^2 -Test	12
2.2.2 Gaps	13
2.3 Discrepancy	13
2.4 Transformed Random Variables	17
2.4.1 Inversion	17
2.4.2 Transformation of Random Variables	18
2.4.3 Normally Distributed Random Variables	20
2.5 The Mersenne twister	21
3 Numerical Cubature and Monte Carlo and Quasi-Monte Carlo Methods	24
3.1 Product Formulas (are useless here)	26
3.2 Monte-Carlo methods	28
3.3 Quasi-Monte-Carlo Methods	29
3.4 (t, m, s) -nets	34
3.5 The Smolyak method	37
3.5.1 Equidistant Points	44
3.5.2 Gauß-Points	44
3.5.3 The Clenshaw-Curtis grids	44

4	Numerical Computation of European Options	46
4.1	Option Pricing: A Very Short Introduction	46
4.2	Binomial Methods	48
4.3	Finite Difference Methods	52
4.3.1	A recursion method for tridiagonal matrices	56
4.3.2	Convergence Theory	57
4.4	Discretization in Time	58
5	Stochastic Differential Equations (SDE)	63
5.1	Introduction to SDEs	63
5.2	Existence and uniqueness of strong solutions	66
5.3	Stochastic Taylor Expansions	67
5.4	The Euler–Maruyama approximation	70
5.5	Approximation of Moments	74
5.6	Strong Convergence and Consistency	75
5.7	Weak Convergence and Consistency	77
5.8	Stability	79
5.9	Higher order methods	80
6	Elliptic Partial Differential Equations	84
6.1	Finite Difference Methods	85
6.2	Categories of second order PDEs	87
6.3	Variational Formulation of Elliptic PDEs	88
6.4	Ritz-Galerkin methods	90
6.5	Some Simple Finite Elements	92
6.6	Approximation Results	97
6.7	Example: 1D Finite Element Discretization for the Black-Scholes Equation	100
7	American Option Pricing	103
7.1	The Binomial Method	105
7.2	Obstacle Problem	106
7.3	Finite Difference Methods	109
7.3.1	Classical Iterative Methods	111
7.3.2	Projected SOR-method for Complementary Problems	117

8 Exotic Options	122
8.1 Asian Options	122
8.2 Convection–Diffusion Problems	124
8.3 SUPG-method	127

Preface

This is a slightly extended version of my manuscript to the lecture *Numerical Finance* that was given at the University Ulm several times.

The current version corresponds to the lecture that I gave in the summer term 2007. It was given in the framework of the international Master programme in Finance, but it was of course open to the standard diploma curriculum in Mathematik and Wirtschaftsmathematik.

This manuscript can be just a first introduction to a rapidly developing field. It is highly recommended also to consult the literature. I have given a list of text books in the bibliography. Additional information can also be found via the web page of the institute, i.e.,

www.mathematik.uni-ulm.de/numerik

Finally, I wish to thank Michael Lehn, Mario Rometsch and Dr. Shaowu Tang for their very good work as assistants to the lecture and for many very valuable comments. I am particularly grateful to my colleague Prof. Dr. Rüdiger Kiesel for providing several examples from mathematical finance. My students Timo Tonn and Johannes Ruf made several helpful remarks which I kindly acknowledge. Petra Hildebrand fought with my hand-writing and typed the first version of this manuscript in L^AT_EX.

Chapter 1

Introduction

At a first glance, one may ask why numerical methods in financial mathematics are needed at all. On one hand, traders prefer formulae since influence of parameters can easily be detected. On the other hand, sophisticated models require numerical simulations since explicit formulae are not possible. For such simulations, one may argue that highly sophisticated software packages are available that do all computations needed. As an example, we just mention two main fields in financial mathematics in which numerical methods are needed, namely:

- Calculation of prices, values etc. with a given (often complicated) formula on a computer; as an example let us mention that ‘over-the-counter’ (OTC) derivatives are tailor-made for certain specific applications. Hence, there no standard software can be used in this case.
- Computation of an approximate solution to problems that do not have a closed formula such as
 - certain linear or non-linear systems of equations,
 - ordinary and/or partial differential equations,
 - problems of optimization,
 - differential-algebraic equations (DAEs),
 - variational inequations,
 - and so on.

In all these possible applications, the user (the person that makes use of the results of a numerical computation) in particular is interested in

- Exactness and reliability of the results:
If a result of a numerical computation is the basis for further decisions, the user has to know how ‘good’ this result is, namely how close the numerical approximation is to ‘the’ ‘exact’ solution. A precise error statement (e.g. an estimate of the relative error of the desired quantity) is a necessary information for the further use of the numerical results.
- Stability:
Often a numerical computation is based on input data. Not only in applications from financial mathematics those input data are not available at all or are at least subject to stochastic influences. This means, one cannot expect to have exact input data, they will in general contain errors. Consequently, the numerical computation must not be sensitive to small errors in the input data in the sense that small errors in the input cause large errors in the output. This topic is known in Numerical Mathematics as stability.
- Efficiency:
In a large range of applications one needs a numerical computation not sometime but within a short period of time. There are even applications in which the computation has to be performed in *real time*, i.e., in the same time, the process to be simulated takes in reality. This demand is only achievable if the numerical method used is highly efficient.

From these different demands, particular numerical questions and problems arise, namely:

- Reliability of computed approximations:
In order to give a precise error estimate for the quantity under consideration, an error analysis of the corresponding numerical methods and algorithms is required. This in particular leads to the mathematical field of *Approximation Theory*.
- Stability of the numerical methods:
The study of the stability of numerical methods is an own field within *Numerical Analysis*, sometimes also called *perturbation theory*.

- Efficiency:

This topic is especially relevant for high-dimensional, highly complex, or time-critical problems (e.g. problems of control, real time problems). The study of these kind of questions is called *Complexity Theory* which is also a well-established field within Numerical Analysis.

From the above introduction, we see that a good knowledge and the correct use of numerical tools is very important for the user. In particular, also a practitioner should know which numerical tool is useful for which kind of problem. An incorrect use may not only yield to extremely large computing times (which might cause that the numerical results are worthless) but the numerical simulation may also have nothing to do with the underlying problem (which means that the numerical results are wrong).

Finally, it turns out that applications from finance sometimes require numerical methods that are well-known from other fields of application. As an example, the valuation of certain exotic options lead to so-called *convective-diffusion* or *hyperbolic* problems that are standard in fluid dynamics. In this sense, the described numerical methods are of broader interest than only finance.

Chapter 2

Numerical Generation of Random Numbers

The modelling of financial processes often requires also to take into account stochastic influences, e.g., the seemingly random development of a stock price in the future. In order to simulate a stochastic behaviour within a numerical simulation, one has to realize randomness on a computer, i.e., the generation of random numbers.

Possible applications (among others) include:

- Numerical realization and simulation of stochastic processes. This field in fact has a huge area of applications far beyond financial mathematics. Let us just mention traffic simulation, medicine, science and engineering.
- Monte–Carlo–Methods. We will come to these methods for a specific application later, they require in particular the availability of random numbers.

The main problem is that a computer is a *deterministic* calculating machine, i.e., any algorithm, any process on the computer is deterministic. Thus, the nature of a computer is in contrast to the generation of *random* numbers. Because of this one usually talks of *pseudo random numbers*, i.e., numbers that are generated in a deterministic way but that reflect a random behaviour in a ‘good’ way. Moreover, often random numbers mimicing a given distribution are required. First we analyze the generators of pseudo random numbers for uniformly distributed numbers. Other distributions will then be realized with the aid of suitable transformations.

2.1 Congruence Methods

We start with the maybe most simple family of methods, the so-called *congruence methods*.

Definition 2.1.1 For $M \in \mathbb{N}$ set $\mathbb{Z}_M := \{0, \dots, M - 1\}$. A congruence method of first order constructed by an initial value $y_0 \in \mathbb{Z}_M$ with a function

$$f : \mathbb{Z}_M \rightarrow \mathbb{Z} \quad (2.1.1)$$

is a sequence $(y_n)_{n \in \mathbb{N}} \subset \mathbb{Z}_M$ defined by the rule

$$y_{n+1} := f(y_n) \pmod{M} . \quad (2.1.2)$$

This method is called *linear*, if f is affine-linear, i.e., if there exist $a, b \in \mathbb{Z}$ such that $f(x) = ax + b$. \square

For the congruence method we can now easily prove the following properties.

Theorem 2.1.2 Let the sequence $(y_n)_{n \in \mathbb{N}}$ be generated by the congruence method. Then, the following statements hold:

- (a) The created sequence $(y_n)_{n \in \mathbb{N}}$ has a period with the maximal length M .
- (b) For the linear congruence method with $b = 0$ (the so called Prime-Modulo-Generator) $y_m = 0$ must be excluded for all $m \in \mathbb{N}$.

Proof:

- (a) Because of $\#\mathbb{Z}_M = M$ there exist at least two identical elements in $\{y_0, \dots, y_M\}$, i.e. $\exists 0 \leq i < j < M$ such that $y_i = y_j$. Therefore $y_i = y_{i+p}$ for all $n \in \mathbb{N}$.
- (b) For $y_m = 0$ (for some $m \in \mathbb{N}$) and $b = 0$ we obtain

$$f(y_m) = ay_m + b = 0 = y_m ,$$

so that $y_m = y_n$ for all $n \geq m$, i.e., we obtain a constant sequence, which of course is non-random. \square

The periodicity of the generated sequence is of course a serious drawback of the congruence method. Thus, in practice M should be chosen as large as possible in order to obtain a maximal length of the period. However, Theorem 2.1.2 (a) gives only an *upper bound* for the length of the period, in practice it could even be much smaller as we have seen in (b). The next result gives a precise statement for the length of the period of the Prime-Modulo-Generator.

Theorem 2.1.3 *Let M be a prime number. Then, the Prime-Modulo-Generator $y_{n+1} = ay_n \pmod{M}$ has the smallest period $M - 1$ if a is a primitive root of M , i.e., if*

$$(a^i - 1) \begin{cases} \not\equiv 0 \pmod{M}, & \text{if } 1 \leq i < M - 1, \\ \equiv 0 \pmod{M}, & \text{if } i = M - 1. \end{cases}$$

Proof: By Theorem 2.1.2 (b) we can assume $y_0 \neq 0$. For the sequence $(z_n)_{n \in \mathbb{N}}$ with $z_0 := y_0$, $z_n := f(z_{n-1}) = az_{n-1}$ we obviously have $z_n = a^n z_0$. Thus $y_n = z_n \pmod{M} = y_0$ holds if and only if $a^n \equiv 1 \pmod{M}$. Thus, by assumption on a we have $n = M - 1$ which is the smallest period. \square

Example 2.1.4 *We consider the case $M = 11$ with the choices of the parameters $a = y_0 = 5$. Note that in this case we have $a^5 = 3125 = 11 \times 284 + 1$, which implies $a^5 \pmod{11} \equiv 1$, i.e., we expect that the periodic length is equal to 5. In fact:*

$$\begin{aligned} y_1 &= 25 \pmod{11} = 3 \\ y_2 &= 15 \pmod{11} = 4 \\ y_3 &= 20 \pmod{11} = 9 \\ y_4 &= 45 \pmod{11} = 1 \\ y_5 &= 5 \pmod{11} = 5 = y_0 \end{aligned}$$

Example 2.1.5 *The generator RANDU, which still is often used in mathematical software packages is a Prime-Modulo-Generator with $a = 2^{16} + 3$ and $M = 2^{31}$ (see exercises).*

Example 2.1.6 *An example of a non-linear congruence method is the inverse congruence method, where we have*

$$f(x) = a\bar{x} + b, \quad a, b \in \mathbb{Z}_M, M \text{ prime,}$$

is used and \bar{x} is defined for a given $x \in \mathbb{Z}_M$ as

$$\begin{cases} \bar{x} = 0, & \text{if } x = 0, \\ x\bar{x} \equiv 1 \pmod{M}, & \text{else.} \end{cases}$$

Obviously, the calculation of \bar{x} is the most expensive part from the numerical point of view. This can e.g. be done with the euclidean algorithm.

If the length of the period is M , the calculated pseudo random numbers are obviously uniformly distributed. If they are normalized through $\frac{y_i}{M}$ on the unit interval $[0, 1]$, they can be subjected to statistical tests in order to check if the desired distribution is in fact matched. We will describe this in the next section.

2.2 Frequency and Gap Tests

Once a sequence of random numbers is generated, one of course wants to check if this sequence is of the desired distribution. For the uniform distribution one may look at a graphical visualization, where each random number within an interval is plotted with a different vertical coordinate. Such a visualization is shown in Figure 2.1. Even though this graphical visualization

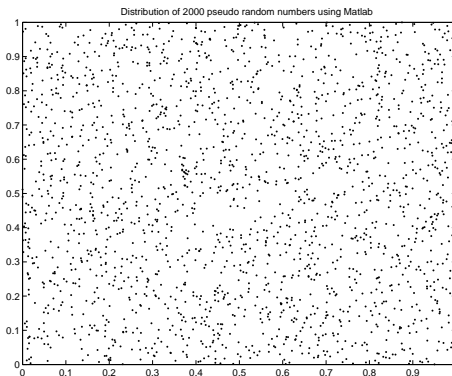


Figure 2.1: Visualization for the random number generator of MATLAB.

gives a first idea, it clearly has a number of serious drawbacks. First of all, it is more or less restricted to the uniform distribution. Moreover, and more seriously, it does not give any quantitative information. Thus, we describe in

this section how standard statistical tests can be used in order to investigate the quality of generated sequences of pseudo random numbers.

2.2.1 The χ^2 -Test

Let us briefly recall the well-known χ^2 -test from statistics:

Divide $[0, 1]$ into $m + 1$ subintervals $J_i = [x_i, x_{i+1})$, $0 = x_0 < x_1 < \dots < x_{m+1} = 1$ and define the quantity

$$B_i := \#t_\nu \text{ in the interval } J_i$$

(t_ν are the pseudo random number, $\nu = 1, \dots, n$). For the test to be meaningful every B_i should be at least of the size 5 to 10. Further let

$$E_i := \frac{n}{m + 1}$$

be the expected number of t_ν 's in J_i in case of an equal distribution. Then, we define

$$\chi_{(n)}^2 := \sum_{i=0}^m \frac{(B_i - E_i)^2}{E_i}$$

and we have

$$\chi_{(n)}^2 \xrightarrow[n \rightarrow \infty]{d} \chi_m^2$$

where χ_m^2 is the chi-square distribution with density

$$f_m(x) := \begin{cases} \left(\frac{x}{2}\right)^{\frac{m}{2}} \frac{e^{-\frac{x}{2}}}{x\Gamma(\frac{m}{2})}, & x > 0, \\ 0, & s \leq 0, \end{cases}$$

and $\Gamma(x) := \int_0^\infty t^{x-1} e^{-t} dt$ is the *Gamma function*.

A significant test results, if this test is realized for a large number (say N) of realizations of a random number generator. Then, the quantity p_i of the calculated χ^2 -values in the intervals

$$\left[i - \frac{1}{2}, i + \frac{1}{2}\right), \quad i = 1, 2, \dots$$

are counted. If the points $(i, p_i/N)$ are “close” to the probability density-function f_m , the random number generator has passed the χ^2 test. A possible quantitative measure could be the L_2 -norm.

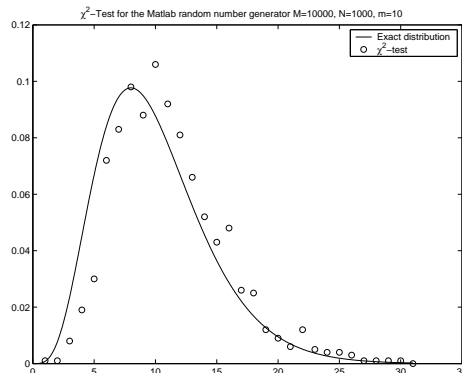


Figure 2.2: A χ^2 -test for the random number generator of MATLAB. Obviously, the test was succesful.

Example 2.2.1 *Figure 2.2 shows the result of a χ^2 -test for the random number generator of MATLAB. The code for the χ^2 -test is also written in MATLAB.*

2.2.2 Gaps

Definition 2.2.2 *For a given interval $J \subset [0, 1]$ a sequence $(t_n)_{n \in \mathbb{N}_0}$ is said to have a gap of length k , if there exists some $n \in \mathbb{N}_0$ such that $t_n, \dots, t_{n+k-1} \notin J$, but $t_{n+k} \in J$. \square*

For a corresponding test, choose $h \in \mathbb{N}$, and count the number of gaps of length $0, 1, \dots, h-1, h$. On this sequence of pseudo random numbers, the above χ^2 -test is applied.

For further information on random number generation and corresponding tests, we refer to [10].

2.3 Discrepancy

We have seen statistical tests to check the distribution of pseudo random numbers. We have concentrated on the uniform distribution. So far, we do not have a *measure* how good a uniform distribution is matched. We will now introduce such a measure.

Definition 2.3.1 Let $X := \{x_1, \dots, x_N\} \subset [0, 1]^m$ be a sequence of normalized pseudo random vectors.

(a) Let \mathcal{Q} be the set of all quads in $[0, 1]^m$. Then, we call

$$D(X) := \sup_{Q \in \mathcal{Q}} \left| \frac{\#\{x_i \in X : x_i \in Q\}}{\#X} - \text{vol}(Q) \right|$$

the extreme discrepancy of X .

(b) For $X = \{x_1, x_2, \dots\}$, we also use the abbreviation

$$D_N := D(\{x_1, \dots, x_N\}).$$

If $\lim_{N \rightarrow \infty} D_N = 0$, then we say that X consists of uniformly distributed points. \square

The idea behind the latter definition is that for a set of uniformly distributed points the portion of those points lying in a quad Q should at least almost correspond to the volume of Q . Of course the quantity $D(X)$ is not so easy to compute since the determination of the supremum over all quads might be a delicate and in particular expensive task. Thus, one also considers the following measure.

Definition 2.3.2 Let $Q^* = \prod_{i=1}^m [0, y_i)$, $0 < y_i \leq 1$, be a quad with one corner in 0 and denote by \mathcal{Q}^* the set of all these quads. Then, the quantity

$$D^*(X) := \sup_{Q^* \in \mathcal{Q}^*} \left| \frac{\#\{x_i \in X : x_i \in Q^*\}}{\#X} - \text{vol}(Q^*) \right|$$

is called star discrepancy of X . \square

Obviously, the star discrepancy is easier to access. The next result shows that it is in fact an approximation of the discrepancy.

Proposition 2.3.3 The following estimates hold

(a) $0 \leq D_N \leq 1$,

(b) $D_N^* \leq D_N \leq 2^m D_N^*$,

(c) $D_N^* \geq \frac{1}{2N}$ for $m = 1$.

Proof: We leave the proof as an easy exercise. \square

In Definition 2.3.1 (b) we just require that D_N tends to zero for $N \rightarrow \infty$. This is a statement of pure asymptotic character. In practice one is of course also interested that already a moderate number of pseudo random numbers is almost uniformly distributed. Hence, one is interested how fast D_N tends to zero, i.e., what is the rate of decay. This is reflected by the following definition.

Definition 2.3.4 A sequence $(x_k)_{k \in \mathbb{N}} \subset [0, 1]^m$ is called of low discrepancy if

$$D_N \leq C_m \frac{(\log N)^m}{N} \quad (2.3.1)$$

with a constant $0 \leq C_m < \infty$ independent of N . A deterministic sequence of numbers is called a set of quasi random numbers if (2.3.1) holds. \square

Remark: In moderate dimensions m , the above estimate basically means

$$D_N \approx \mathcal{O}(N^{-1}).$$

However, the curse of dimensionality shows up due to the term $(\log N)^m$. It is widely believed (see [10], p. 32) that

$$D_N^* \geq B_m \frac{(\log N)^m}{N}$$

with a constant B_m depending only on m . This means that

$$\mathcal{O}(N^{-1}(\log N)^m)$$

would be the optimal rate.

Some examples

Example 2.3.5 For $m = 1$ and $x_i := \frac{2i-1}{2N}$, $i = 1, \dots, N$, we obtain $D_N^* = \frac{1}{2N}$. In fact, let $Q^* = [0, y)$, $0 < y \leq 1$ so that $\text{vol}(Q^*) = y$ and

$$\begin{aligned} x_i \in Q^* &\iff \frac{2i-1}{2N} < y &\iff 2i-1 &\leq 2Ny \\ & &\iff i &\leq \frac{2Ny+1}{2}. \end{aligned}$$

Hence, we have

$$D^*(X) = \sup_{0 < y \leq 1} \left\{ \underbrace{\frac{2Ny + 1}{2N} - y}_{= \frac{2Ny+1}{2N} - \frac{2Ny}{2N} = \frac{1}{2N}} \right\} = \frac{1}{2N}.$$

By Proposition 2.3.3 (c) this is optimal. On the other hand, the sequence (x_i) , $i \in \mathbb{N}$ has to be computed for every N from scratch which of course is highly inefficient if N grows. Hence, it would be better if the numbers could be set in a dynamical way that allows for updating. The next example shows one way to achieve this.

Definition 2.3.6 Let $b \geq 2$ be an integer and for $i \in \mathbb{N}$ consider the b -adic representation of i to the base b , namely

$$i = \sum_{k=0}^j d_k b^k, \quad d_k \in \{0, 1, \dots, b-1\},$$

where the upper index j of course depends on i (or, on a computer, on the finite arithmetic). Then, the mapping ϕ_b defined by

$$\phi_b(i) := \sum_{k=0}^j d_k b^{-k-1}$$

is called radical-inverse function. \square

The radical-inverse function can be interpreted as a ‘reflection at the radix point’, i.e., $i \mapsto x \in \mathbb{Q}$, $0 < x < 1$. If the number of digits j in i is increased, the highest power of b is increased which in turns increases the fineness of the rational numbers i , i.e., new numbers are dynamically inserted. Combining different radical-inverse functions yields the following sequence.

The sequence

$$\frac{1}{2}, \frac{1}{4}, \frac{3}{4}, \frac{1}{8}, \frac{5}{8}, \frac{3}{8}, \frac{7}{8}, \frac{1}{16}, \dots$$

known as *van der Corput* sequence can be generated by $x_n := \phi_2(n)$. In general, a sequence defined by

$$x_n := \phi_b(n)$$

is called *van der Corput* sequence. It can be shown that all these sequences are low discrepancy sequences.

An extension to several space dimensions is given by the following definition.

Definition 2.3.7 Assume that p_1, \dots, p_m are coprime integers. Then, the vectors

$$x_i := (\phi_{p_1}(i), \dots, \phi_{p_m}(i)) \in \mathbb{R}^m, \quad i = 1, 2, \dots$$

are called Halton sequence.

2.4 Transformed Random Variables

So far we have considered ‘only’ uniformly distributed quasi random numbers. A (very) simple method to construct an approximately normally distributed sequence of random numbers from a uniformly distributed sequence $U_i \sim \mathcal{U}[0, 1]$ is the following

$$X := \sum_{i=1}^{12} U_i - 6.$$

One easily obtains by the central limit theorem that approximately $X \sim \mathcal{N}(0, 1)$. Obviously, this is not a very sophisticated method and, as we shall see next, transformation methods are in fact much better.

2.4.1 Inversion

The quite simple idea of this approach is to invert the particular distribution function.

Theorem 2.4.1 Let $U \sim \mathcal{U}[0, 1]$ and F be a uniformly continuous, strictly monotone distribution function. Then there exists the inverse $F^{-1} : [0, 1] \rightarrow \mathbb{R}$ and $F^{-1}(U)$ is distributed according to F .

Proof: It is easily seen that for $F^{-1}(z) = x$

$$\begin{aligned} U \sim \mathcal{U}[0, 1] &\iff P(U \leq \xi) = \xi \text{ for } 0 \leq \xi \leq 1 \\ &\iff P(F^{-1}(U) \leq x) = P(U \leq F(x)) = F(x). \quad \square \end{aligned}$$

Remark 2.4.2 The statement of Theorem 2.4.1 also applies for more general distribution functions.

Even though this straightforward approach seems to yield the desired result, there is a serious drawback. E.g. for the normal distribution there is no Gaussian error-integral, in particular neither for $F(x)$ nor for $f = F^{-1}(x)$ a closed formula exists. Thus one has to solve the non-linear problem $f(x) = u$

numerically e.g. by an iterative method (bisection, secant method, Newton). Moreover, it can easily be seen that for $\xi \approx 1$ small modifications in ξ cause large modifications in x , i.e., instabilities occur. As an alternative, one may compute a numerical approximation $G(u) \approx F^{-1}(u)$ e.g. by *rational approximation* in order to reflect the poles correctly.

2.4.2 Transformation of Random Variables

As an alternative, we now consider transformation methods. The key result behind this approach is the following theorem. For simplicity sake, we will first state and prove it in the 1D case.

Theorem 2.4.3 *Let X be a random variable with probability density function (pdf) $f(x)$ and cumulative distribution function (cdf) $F(x)$. Further let be $h : S \rightarrow B$, $S, B \subset \mathbb{R}$, where S denotes the support of f , i.e.,*

$$S := \text{supp } f := \overline{\{x \in \mathbb{R} : f(x) \neq 0\}}.$$

If h is strictly monotone, we have

- (a) $Y := h(X)$ is a random variable with cdf $F(h^{-1}(Y))$.
- (b) If h^{-1} is absolutely continuous, then

$$f(h^{-1}(y)) \left| \frac{dh^{-1}(y)}{dy} \right| \tag{2.4.1}$$

is the pdf of $h(X)$ for almost all y .

Proof:

- (a) Because h is strictly monotonously increasing, this also holds for the inverse and we obtain for the distribution of Y that $P(Y \leq y) = P(h(X) \leq y) = P(X \leq h^{-1}(y)) = F(h^{-1}(y))$.
- (b) Because h^{-1} is absolutely continuous, the density of $Y = h(X)$ is equal to the derivative of the distribution function almost everywhere. By the chain rule, we obtain

$$\frac{d}{dy} F(h^{-1}(y)) = \underbrace{F'(h^{-1}(y))}_{=f(h^{-1}(y))} \left(\frac{d}{dy} h^{-1}(y) \right)$$

and the absolute value in (2.4.1) is necessary in order to obtain a positive density, see [14]. \square

Now we apply Theorem 2.4.3 to a given sequence of random numbers $X \sim \mathcal{U}[0, 1]$. Let f be the corresponding pdf, i.e.,

$$f(x) := \begin{cases} 1, & \text{if } 0 \leq x \leq 1, \\ 0, & \text{else,} \end{cases}$$

i.e., $S = \text{supp } f = [0, 1]$. Assume that we are interested in a sequence of random numbers Y with probability density function $g(y)$. Thus, we define h^{-1} as in (2.4.1), i.e.,

$$\left| \frac{dh^{-1}(y)}{dy} \right| = g(y)$$

so that $Y := h(X)$ is the desired sequence. Let us describe two particular examples.

Example 2.4.4 (*Exponential distribution*) *It is well-known that the probability density function is*

$$g(y) = \begin{cases} \lambda e^{-\lambda y} & \text{for } y \geq 0, \\ 0 & \text{for } y < 0, \end{cases}$$

where λ is the free parameter. Thus, $B := [0, \infty) = \mathbb{R}_0^*$ and $S := [0, 1]$. Hence $h : S \rightarrow B$ is defined by $y := h(x) := -\frac{1}{\lambda} \log x$ and thus $h^{-1}(y) = e^{-\lambda y}$ for $y \geq 0$. Hence,

$$\underbrace{f(h^{-1}(y))}_{=1} \left| \frac{d}{dy} h^{-1}(y) \right| = |(-\lambda)e^{-\lambda y}| = g(y),$$

and $h^{-1} : B \rightarrow S$ (note that both sides are $= 0$ for $y < 0$). According to Theorem 2.4.3, we see that $h(x)$ is exponentially distributed.

Example 2.4.5 (*Standard normal distribution*) *It is well-known that the probability distribution function is*

$$g(y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}y^2\right) \stackrel{!}{=} \left| \frac{d}{dy} h^{-1}(y) \right|,$$

where the latter equation is the one to be satisfied by Theorem 2.4.3. This is a differential equation for h^{-1} that does not have an analytical solution. Thus one has to resort to numerical solution methods, see *Numerical Mathematics II*.

Without proof, we quote the generalization of Theorem 2.4.3 to the multivariate case.

Theorem 2.4.6 *Let X be a sequence of random variables on \mathbb{R}^n with density function $f(x) > 0$ on $S = \text{supp } f$. Moreover, assume that $h : S \rightarrow B$, $S, B \subset \mathbb{R}^n$ is explicitly invertible and $Y := h(X)$ is the transformed sequence. If h^{-1} is continuously differentiable on B , then Y has the density function*

$$f(h^{-1}(y)) |\det \mathcal{J}h^{-1}(y)|, \quad y \in B,$$

where $\mathcal{J}h^{-1}(y)$ is the Jacobi-Matrix of h^{-1} , $(\mathcal{J}h^{-1}(y))_{i,j} = \frac{\partial}{\partial y_j}(h^{-1}(y))_i$. \square

2.4.3 Normally Distributed Random Variables

Since normally distributed pseudo random variables are highly relevant in many applications, we give a corresponding number generator for this case here. We apply the above described transformation method in order to generate normally distributed random numbers. This is the method of *Box-Muller* which was introduced in 1952.

We now describe this method. Define the function $h : [0, 1]^2 \rightarrow \mathbb{R}^2$ by

$$\begin{aligned} h_1(x_1, x_2) &:= \sqrt{-2 \log x_1} \cos 2\pi x_2 = y_1 \\ h_2(x_1, x_2) &:= \sqrt{-2 \log x_1} \sin 2\pi x_2 = y_2. \end{aligned}$$

It is readily seen that

$$h^{-1}(y_1, y_2) = \begin{bmatrix} \exp \left\{ -\frac{1}{2}(y_1^2 + y_2^2) \right\} \\ \frac{1}{2\pi} \arctan \frac{y_2}{y_1} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

and the Jacobi-Matrix is given by

$$\mathcal{J}h^{-1}(y) = \left(\frac{\partial x_i}{\partial y_j} \right)_{i,j} = \begin{bmatrix} (-y_1) \exp \left\{ -\frac{1}{2}(y_1^2 + y_2^2) \right\} & \frac{1}{2\pi} \frac{1}{1 + \frac{y_2^2}{y_1^2}} \cdot \left(-\frac{y_2}{y_1^2} \right) \\ (-y_2) \exp \left\{ -\frac{1}{2}(y_1^2 + y_2^2) \right\} & \frac{1}{2\pi} \frac{1}{1 + \frac{y_2^2}{y_1^2}} \cdot \frac{1}{y_1} \end{bmatrix}.$$

Hence, we obtain

$$\begin{aligned} \det \mathcal{J}h^{-1}(y) &= \frac{1}{2\pi} \exp \left\{ -\frac{1}{2}(y_1^2 + y_2^2) \right\} \left[-y_1 \frac{1}{1 + \frac{y_2^2}{y_1^2}} \cdot \frac{1}{y_1} - y_2 \frac{1}{1 + \frac{y_2^2}{y_1^2}} \cdot \frac{y_2}{y_1^2} \right] \\ &= - \left(\frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{1}{2}y_1^2 \right\} \right) \underbrace{\left(\frac{1}{\sqrt{2\pi}} \exp \left\{ -\frac{1}{2}y_2^2 \right\} \right)}_{=-1} \end{aligned}$$

which is the probability density function of the standard normal distribution. Hence $h(X)$ is normally distributed. The corresponding algorithm then reads as follows.

Algorithm 2.4.7 (Box-Muller)

- (1) Generate two variables $U_1 \sim \mathcal{U}[0, 1]$ and $U_2 \sim \mathcal{U}[0, 1]$;
- (2) Set $\Theta := 2\pi U_2$ and $\rho = \sqrt{-2 \log U_1}$;
- (3) Compute $Z_1 := \rho \cos \Theta$ and $Z_2 := \rho \sin \Theta$, which are two independent standard normal distributed random numbers. \square

A modification of this is the method of Marsaglia, see [14, Chap. 2.3.2]

2.5 The Mersenne twister

We now briefly describe a modern pseudo random number generator which is maybe nowadays the most standard one also used in software packages like MATLAB or MATHEMATICA. The so-called *Mersenne twister* was introduced in 1997 by M. Matsumoto and T. Nishimura. The main advantages are:

- efficiency: very fast generation;
- quality: it passes several tests for statistical randomness and overcomes known drawbacks of other generators.

The name comes from the fact that the period length is a Mersenne prime, i.e.,

$$M_n = 2^n - 1.$$

The commonly used is called MT19937 with $n = 19937$ with 32-bit word length, but there is also the variant MT19937-64 with 64-bit word length. Note that the period length is

$$2^{19937} - 1 \approx 4.3 \cdot 10^{6001}$$

which obviously is sufficient for many applications. Moreover, it can be shown that the generated sequence is a low discrepancy one.

The iteration is defined by

$$x_{k+n} = x_{k+m} \oplus (x_k^u | x_{k+1}^\ell)A, \quad k = 0, 1, \dots,$$

where \oplus denotes the bitwise XOR operation,

- n is the degree of recurrence (624 for MT19937),
- m the number of parallel sequences, $1 \leq m \leq n$ (397 for MT19937)
- u, ℓ additional tempering bit shifts ($u = 11$, $\ell = 18$ for MT19937).

Next, we define $(x_k^u | x_{k+1}^\ell)$ and A , to be precise

$$A = \begin{pmatrix} 0 & I_{n-1} \\ a_n & (a_{n-1}, \dots, a_0) \end{pmatrix}$$

with I_{n-1} being the $(n-1)$ -dimensional identity and in the standard matrix-vector multiplication bitwise XOR replaces addition. For MT19937 one uses

$$\mathbf{a} = 9908B0DF_{16} = (a_n, \dots, a_0), \quad n = 624.$$

This means

$$xA = \begin{cases} x \gg 1, & \text{if } x_0 = 0, \\ (x \gg 1) \oplus \mathbf{a}, & \text{if } x_0 = 1, \end{cases}$$

where $x \gg u$ denotes the u -bit shiftright. Next,

- x_k^u denote the upper $w - r$ bits of x_k
- x_{k+1}^ℓ the lower r bits of x_{k+1} ,

where $w = 32$ and $r = 31$ is used in MT19937.

Thus, if $x = (x_{w-1}, \dots, x_0)$, we have

$$\begin{aligned} x^u &= (x_{w-1}, \dots, x_r) \\ x^\ell &= (x_{r-1}, \dots, x_0) \end{aligned}$$

and $(X^u | X^\ell)$ is the concatenation.

A second step is performed in order to compensate for the reduced dimensionality of equidistribution. Such a strategy is called *tempering* and was introduced by Matsumoto and Kurita in 1994. Formally, the tempering can be written as $z = xT$. In the case of the Mersenne twister this is realized by the following successive transformations

$$\begin{aligned} y &:= x \oplus (x \ggg u) \\ y &:= y \oplus ((y \lll s) \text{ AND } b) \\ y &:= y \oplus ((y \lll t) \text{ AND } c) \\ y &:= y \oplus (g \ggg \ell) \end{aligned}$$

and in MT19937 one uses u, ℓ as above, $(s, b) = (7, 9D2C5680_{16})$ and $(t, c) = (15, EFC60000_{16})$. This form is called *twisted generalized feedback shift register (GFSR)*.

As already mentioned in [9] it was proven that this sequences reach the optimal period length.

Chapter 3

Numerical Cubature and Monte Carlo and Quasi-Monte Carlo Methods

Many important applications of mathematical modelling in financial mathematics require the computation of integrals. These usually highly dimensional integrals are often so complex that this cannot be done analytically. Hence, one has to resort to numerical methods. We start with an example that shows of which type these integrals might be. This also clearly shows the numerical challenges.

Example 3.0.1 (Mortgage-Backed Securities (MBS)) *MBS are a prominent example of asset-backed securities and widely used in the USA. An MBS is a fixed-income security whose performance is related to a pool of customer mortgages. The bank, who is giving out a mortgage to a customer, faces its prepayment risk, e.g. the risk that the customer pays back his mortgage pre (mostly to refinance in order to benefit from low interest rates). Thus there is only a supposed behaviour that depends on external parameters. Hence a stochastic modeling is appropriate and required.*

The value of the bond coincides with the expected return, i.e., an expectation rate which mathematically is an integral. Let us illustrate the numerical problem for a concrete example. Let the duration of the bond be 30 years, i.e., 360 months. We denote by r_k the interest rate per month k , $1 \leq k \leq d = 360$. This is a random variable and we assume that it is log-normally distributed,

i.e.,

$$r_k = r_{k-1} e^{\sigma Z_k - \frac{\sigma^2}{2}},$$

(Rendleman-Bartter interest-model). Here Z_k are independent and standard-normally distributed. Moreover, we denote the discounting coefficient by

$$d_k = \prod_{i=0}^{k-1} \frac{1}{1+r_i}.$$

The repay rate is modelled by

$$w_k = w_k(r_k) = K_1 + K_2 \arctan(K_3 r_k + K_4), \quad K_2 \cdot K_3 = 0.$$

This results in the following model for the cash flow per month k :

$$\begin{aligned} M_k &= C(1-w_1) \dots (1-w_{k-1}) \left(1 - w_k + w_k \prod_{i=1}^{k-1} \frac{1}{1+r_i} \right) \\ &= C(1-w_1) \dots (1-w_{k-1})(1-w_k + w_k d_k), \end{aligned}$$

where C is the investment at the beginning of the contract. Hence, we obtain the current value of a MBS as

$$W = \sum_{k=1}^d d_k M_k = \sum_{k=1}^d d_k(Z_k) M_k(Z_k).$$

Denoting by $f(z_1) \dots f(z_d)$ the d -dimensional $\mathcal{N}_{0,1}$ -density, a straightforward calculation for the expectation yields

$$\begin{aligned} \Rightarrow \mathbb{E}(W) &= \underbrace{\int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty}}_d \sum_{k=1}^d d_k(Z_k) M_k(Z_k) f(z_1) \dots f(z_d) dz_1 \dots dz_d \\ &= \int_{[0,1]^d} \tilde{W}(u) du, \end{aligned} \tag{3.0.1}$$

where $u = (u_1, \dots, u_d)$, and

$$\tilde{W}(u) = \sum_{k=1}^d M_k(\phi^{-1}(u_k)) d_k(\phi^{-1}(u_k)), \quad \tilde{W} : \mathbb{R}^d \rightarrow \mathbb{R}.$$

This shows that we have to compute an integral in a 360-dimensional space. Obviously, this has to be done by a numerical method.

3.1 Product Formulas (are useless here)

In the latter example, we have to integrate a moderately complicated function over an easy domain (a cube), but in an extremely high-dimensional space. The simplest approach to do this is to use a 1D quadrature rule from any text book on numerical analysis and to use a product formula built on this. We will now show that this would result in an extremely inefficient method.

Let us again consider the above example, i.e., we want to calculate (3.0.1) numerically. To this end, we consider the following 1D quadrature formula for a function $f : [0, 1] \rightarrow \mathbb{R}$

$$Q_n[f] = \sum_{k=1}^n \gamma_k f(t_k),$$

i.e., a quadrature formula for the approximation of $\int_0^1 f(x) dx$ with weights $\gamma_i \in \mathbb{R}$ and quadrature points $t_i \in [0, 1]$. These formulas are now put together to obtain a cubature formula for a function $F : [0, 1]^d \rightarrow \mathbb{R}$

$$\begin{aligned} I_d[F] &:= \int_{[0,1]^d} F(x_1, \dots, x_d) dx_1 \dots dx_d \\ &= \underbrace{\int_0^1 \dots \int_0^1}_{d} F(x_1, \dots, x_d) dx_1 \dots dx_d \\ &\approx \underbrace{\int_0^1 \dots \int_0^1}_{d-1} \sum_{i=1}^n \gamma_i F(t_i, x_2, \dots, x_d) dx_2 \dots dx_d \\ &\approx \dots \approx \sum_{i_1=1}^n \dots \sum_{i_d=1}^n \gamma_{i_1} \dots \gamma_{i_d} F(t_{i_1}, \dots, t_{i_d}) =: Q_{n^d}^{[d]}[F]. \end{aligned} \quad (3.1.1)$$

We now study the error of the latter product formula.

Definition 3.1.1 For a given 1D quadrature formula Q_n (3.1.1) is called product formula. The respective quadrature and cubature errors are

$$R_n[f] := I_1(f) - Q_n[f], \quad R_{n^d}^{[d]}[F] := I_d[F] - Q_{n^d}^{[d]}[F], \quad (3.1.2)$$

for the functions $f : [0, 1] \rightarrow \mathbb{R}$ and $F : [0, 1]^d \rightarrow \mathbb{R}$.

Example 3.1.2 Before we study the error, let us just give a feeling for the complexity of the numerical problem. Let us assume that the parameter n reflects the exactness and also the complexity of $Q_n[f]$ e.g. the number of quadrature points. For $n = 1$ (i.e., one quadrature point in $[0, 1]$, i.e., approximation by constants) one cannot expect a high order of exactness. Thus, let us consider the next higher case $n = 2$. For the above case of $d = 360$, this would amount to

$$2^{360} \approx 2.34 \cdot 10^{108}$$

evaluations of the function F to be integrated, which obviously is highly inefficient.

Moreover, there are also bad news concerning the behaviour of the error. One expects of course that the error decreases for increasing n . In addition, the rate of convergence of the 1D method should be preserved in the multivariate case. This however is *not* true as the following result shows.

Theorem 3.1.3 For every sequence of quadrature formulas Q_1, Q_2, \dots with

$$|R_n[f]| \leq \frac{C_p}{n^p} \|f^{(p)}\|_\infty$$

and every sequence $(\delta_n)_{n \in \mathbb{N}}$, $\delta_n \searrow 0$, there exists a function $F : [0, 1]^d \rightarrow \mathbb{R}$ with $\|F\|_\infty \leq 1$, $\|F^{(p, \dots, p)}\|_\infty \leq \infty$ and

$$R_{n^d}^{[d]}[F] \geq \frac{\delta_n}{n^p}$$

for an infinite number of n . \square

We omit the proof of the latter theorem, refer e.g. to [11] and just remark that the counterexample is closely related to the famous example given by Runge, i.e., the function $\frac{1}{1+x^2} \in C^\infty(\mathbb{R})$ for which the polynomial interpolation fails.

We conclude from the above discussion that for $N = n^d$ sampling points as for a product rule, the error is of the order $\mathcal{O}(N^{-\frac{p}{d}})$ which *cannot* be improved. Thus, product rules are useless for our kind of applications. This means that the rate of convergence becomes slower for increasing space dimensions $d \rightarrow \infty$. Hence, the high-dimensional case is as bad as it can only be, namely

- the amount of work increases exponentially and
- the rate of convergence decreases.

3.2 Monte–Carlo methods

Monte–Carlo methods are nowadays widely used in stochastic modelling and simulation. We describe their use in numerical finance and also give a precise error estimate in the sequel.

Example 3.2.1 *The midpoint rule reads*

$$Q_n^{\text{Mi}}[f] = \frac{1}{n} \sum_{i=1}^n f\left(\frac{2i-1}{2n}\right),$$

i.e., the quadrature points $t_i = \frac{2i-1}{2n}$ are uniformly distributed over the unit interval. For the corresponding product formula we would thus use the grid points

$$\left(\frac{2i_1-1}{2n}, \dots, \frac{2i_d-1}{2n}\right) \in [0, 1]^d, \quad i_j \in \{1, \dots, n\}.$$

In the above example, one would place n^d points uniformly over the unit cube $[0, 1]^d$. The idea is now to distribute these points in a random manner but in such a way that the random numbers are uniformly distributed. Hence, a method of the form

$$Q_n^{\text{MC}}[F] := \frac{1}{n} \sum_{i=1}^n F(x_i), \quad n = 1, 2, \dots$$

with independent uniformly distributed random numbers $x_i \in [0, 1]^d$ is called *Monte-Carlo-Method (MC)* for the approximation of $I_d[F]$. Correspondingly, we denote

$$R_n^{\text{MC}}[F] := Q_n^{\text{MC}}[F] - I_d[F]$$

the error of the quadrature.

Theorem 3.2.2 *If $I_d[F] < \infty$, $I_d[F^2] < \infty$, then it holds*

$$P\left(\lim_{n \rightarrow \infty} R_n^{\text{MC}}[F] = 0\right) = 1.$$

Proof: The proof is a consequence of the *Strong Law of Large Numbers*. \square

Moreover, from of the *Central Limit Theorem* it follows with

$$\sigma_F^2 := I_d[F^2] - I_d[F]^2$$

that

$$\lim_{n \rightarrow \infty} P \left(\frac{\sigma_F}{\sqrt{n}} a < R_n^{MC}[F] < \frac{\sigma_F}{\sqrt{n}} b \right) = \frac{1}{\sqrt{2\pi}} \int_a^b e^{-\frac{t^2}{2}} dt,$$

i.e., we can expect an order of $\frac{1}{\sqrt{n}} = n^{-\frac{1}{2}}$. For high dimensional problems we have that $\frac{1}{2} \gg \frac{p}{d}$, i.e., the expected rate of convergence of a Monte–Carlo method is better compared with the above mentioned product formulas. However, the order $\frac{1}{2}$ is of course a serious drawback.

Remark 3.2.3 *With the use of so-called “antithetic variates” the variance σ_F can be reduced so that the method is (quantitatively) improved. However, this affects only the constant of the \mathcal{O} -Symbol, the slope of $N^{-1/2}$ of the error stays the same.*

For the required random numbers one can use one of the already introduced random number generators. From this point of view a Monte–Carlo Method on a computer could be called pseudo Monte–Carlo method.

3.3 Quasi–Monte–Carlo Methods

The standard Monte–Carlo method obviously has some drawbacks, namely

- the convergence statement is of probabilistic nature;
- the rate of convergence is low.

The idea to improve this is now as follows. If randomly chosen quadrature points lead to the above error estimate, one can expect to find particular quadrature points (or grids) so that the error behaves better. This leads to so-called *Quasi–Monte–Carlo (QMC)* methods.

Definition 3.3.1 A cubature formula $Q_n[f] = \frac{1}{n} \sum_{i=1}^n f(x_i)$ for a sequence $X = (x_i)_{i=1, \dots, n}$ of quasi random numbers is called Quasi-Monte-Carlo formula (QMC formula). Here, quasi random means that X is a previously chosen sequence.

The ultimate goal for a QMC formula is of course to reach a best possible exactness with minimal amount of work. This means that we look for particular good sequences X , sometimes also called *nets*.

We are now going to analyze this method rigorously in order to be able to compare the different methods. To this end, we have to introduce some notation and definitions.

Definition 3.3.2 For a function $f : [0, 1]^d \rightarrow \mathbb{R}$, $f \in C^1([0, 1]^d)$ we call

$$V^{(d)}(f) := \int_{[0,1]^d} |f^{(1, \dots, 1)}(u_1, \dots, u_d)| du, \quad u = (u_1, \dots, u_d)^T \quad (3.3.1)$$

the Vitali variation of f .

Definition 3.3.3 Set $J_k^{(d)} := \{(i_1, \dots, i_k) : 1 \leq i_1 < i_2 < \dots < i_k \leq d\}$ and for $I \in J_k^{(d)}$ let $f_I(u) := f(u)_{|u_{i_k}=1, k \notin I}$, i.e., $(\{1, \dots, d\} \setminus I)$ is the index set of frozen indices)

$$f_I(u_1, \dots, u_d) := f(v_1, \dots, v_d), \quad \text{where } v_i := \begin{cases} u_i, & i \in I, \\ 1, & \text{else,} \end{cases}$$

which means that $f_I : \mathbb{R}^k \rightarrow \mathbb{R}$, $I \in J_k^{(d)}$.

Then, the quantity

$$V(f) := \sum_{k=1}^d \sum_{I \in J_k^{(d)}} V^{(k)}(f_I) \quad (3.3.2)$$

is called Hardy-Krause variation of f .

Before we can give the error estimate, we introduce some further notation: For $I \subset J_k^{(d)}$, let

$$f^{(I)} := \frac{\partial}{\partial x_{i_1}} \dots \frac{\partial}{\partial x_{i_k}} f, \quad du_I := du_{i_k} \dots du_{i_1}, \quad \int_I f = \int_{t_{i_1}}^1 \dots \int_{t_{i_k}}^1 f.$$

Now we are ready to formulate and prove the main error estimate.

Theorem 3.3.4 (Koksma-Hlawka inequality) *Let R_n denote the error of a QMC formula with respect to $X = \{x_i\}_{i=1}^n \subset \mathbb{R}^d$. Then, we have*

$$|R_n[f]| \leq D_n^*(X) V(f) . \quad (3.3.3)$$

We need some preparations for the proof of the latter theorem, which is a famous estimate.

Lemma 3.3.5 *For $t_1, \dots, t_d \in [0, 1]$ we have*

$$f(t_1, \dots, t_d) - f(1, \dots, 1) = \sum_{k=1}^d (-1)^k \sum_{I \in J_k^{(d)}} \int_I f_I^{(I)}(u) du_I. \quad (3.3.4)$$

Proof: By induction over d . For $d = 1$ the fundamental theorem of calculus gives

$$f(t) - f(1) = (-1) \int_t^1 f'(u) du,$$

which coincides with (3.3.4), since here $J_1^{(1)} = \{1\}$ and $f_I \equiv f$, $f_I^{(I)} = f'$.

For $d > 1$, we split the sum in the right-hand side of (3.3.4) into 3 parts. In the following, we refer to the notation in Definition 3.3.3 above.

a) $k \geq 1, d \notin I$ $\Rightarrow v_d = 1$, hence this part reads

$$\sum_{k=1}^{d-1} (-1)^k \sum_{I \in J_k^{(d-1)}} \int_I f_I^{(I)}(u_1, \dots, u_{d-1}, 1) du_I.$$

b) $k = 1, I = \{d\}$ $\Rightarrow v_d = u_d$, hence the sum becomes

$$(-1) \int_{t_d}^1 f^{(0, \dots, 0, 1)}(1, \dots, 1, u_d) du_d = f(1, \dots, 1, t_d) - f(1, \dots, 1).$$

c) $k > 1$, $d \in I \Rightarrow v_d = u_d$, hence we have for this part by the fundamental theorem of calculus

$$\begin{aligned}
(-1) \quad & \sum_{k=1}^{d-1} (-1)^k \sum_{I \in J_k^{(d-1)}} \int_I \int_{t_d}^1 f_{I \cup \{d\}}^{(I \cup \{d\})}(u) \, du_d \, du_I \\
&= \sum_{k=1}^{d-1} (-1)^k \sum_{I \in J_k^{(d-1)}} \int_I f_I^{(I)}(u_1, \dots, u_{d-1}, t_d) \, du_I \\
&\quad - \sum_{k=1}^{d-1} (-1)^k \sum_{I \in J_k^{(d-1)}} \int_I f_I^{(I)}(u_1, \dots, u_{d-1}, 1) \, du_I.
\end{aligned}$$

The right-hand side of (3.3.4) is the sum of these 3 parts which gives:

$$\sum_{k=1}^{d-1} (-1)^k \sum_{I \in J_k^{(d-1)}} \int_I f_I^{(I)}(u_1, \dots, u_{d-1}, t_d) \, du_I + f(1, \dots, 1, t_d) - f(1, \dots, 1).$$

We now apply the induction hypothesis on the first term of the right-hand side of (3.3.4) and obtain

$$\begin{aligned}
f(t_1, \dots, t_{d-1}, t_d) - f(1, \dots, 1, t_d) + f(1, \dots, 1, t_d) - f(1, \dots, 1) \\
= f(t_1, \dots, t_d) - f(1, \dots, 1),
\end{aligned}$$

which proves the claim. \square

Now we are ready to give the proof of the Koksma–Hlawka inequality.

Proof of Theorem 3.3.4: Since $R_n[f(1, \dots, 1)] = 0$ for $n \geq 1$ we have by (3.3.4)

$$R_n[f] = \sum_{k=1}^d (-1)^k \sum_{I \in J_k^{(d)}} R_n[h_I(f^{(I)}; \cdot)] \quad (3.3.5)$$

where

$$h_I(f^{(I)}; t_1, \dots, t_d) = \int_I f_I^{(I)}(u) \, du_I.$$

We first consider a QMC formula for such kind of functions. Since

$$h_I(f; t_1, \dots, t_d) = \int_{[0,1]^d} C_I(t, u) f_I(u) du$$

$$\text{where } C_I(t, u) := \begin{cases} 1, & \text{if } u_{i_\nu} \geq t_{i_\nu} \forall \nu, \\ 0, & \text{else,} \end{cases}$$

we have

$$\begin{aligned} R_n[h_I(f; \cdot)] &= \int_{[0,1]^d} f_I(u) R_n[C_I(\cdot, u)] du \\ &\leq \left(\sup_{u \in [0,1]^d} R_n[C_I(\cdot, u)] \right) \int_{[0,1]^d} f_I(u) du \\ &\leq D_n^*(X) \int_{[0,1]^d} f_I(u) du, \end{aligned}$$

i.e., by (3.3.5)

$$\begin{aligned} |R_n[f]| &\leq D_n^*(X) \left| \sum_{k=1}^n (-1)^k \sum_{I \in J_k^{(d)}} \int_{[0,1]^d} f_I^{(I)}(u) du \right| \\ &\leq D_n^*(X) V(f). \quad \square \end{aligned}$$

Remark 3.3.6 *One can show that the Koksma–Hlawka estimate is in fact sharp, i.e., there exist functions f for which one has “=” in (3.3.4). Further details can e.g. be found in [11], [10], p. 20.*

To be precise, one can find in [10] the following statement: For any $x_1, \dots, x_n \in [0, 1]^s$ and any $\varepsilon > 0$ there exists

$$f \in C^\infty([0, 1]^s), \quad V(f) = 1$$

and

$$\left| \frac{1}{N} \sum_{k=1}^N f(x_k) - \int_{[0,1]^s} f(x) dx \right| > D_N^*(X) V(f) - \varepsilon.$$

Now it remains to construct adequate sequences X . We will now describe some examples.

3.4 (t, m, s) -nets

Let us recall the Koksma–Hlawka inequality from Theorem 3.3.4, namely

$$|R_n[f]| \leq D_n^*(X)V(f).$$

For a given function f , its variation $V(f)$ is a given number that we cannot influence. What we can try is to minimize $D_n^*(X)$ so that the right-hand side of the above estimate is optimal at least for a class of functions f whose variance is of a similar order. Hence, we look at pseudo-random sequences of particular low discrepancy.

A first idea could be to use the Halton sequence introduced in Definition 2.3.7. However, one can show that

$$D_N^*(X) \leq \frac{S}{N} + \frac{1}{N} \prod_{i=1}^s \left(\frac{b_i - 1}{2 \log b_i} \log N + \frac{b_i + 1}{2} \right), \quad N \geq 1,$$

where b_i are the first prime numbers and s is the spatial dimension. From this, one gets

$$D_N^*(X) = \mathcal{O}(N^{-1}(\log N)^s),$$

so that the curse of dimensionality comes back into play even in the Koksma–Hlawka inequality.

Remark 3.4.1 *From several experiments one nowadays assumes that the above estimate is sharp. There are also some modifications of the Halton sequence such as the Hammersby point set, which, however, are not able to overcome the above described problem.*

Definition 3.4.2 *Let $b \geq 2$ be a base and $s \geq 1$ as above the dimension. An interval*

$$E = \prod_{i=1}^s [a_i b^{-d_i}, (a_i + 1) b^{-d_i}] \subseteq [0, 1]^s$$

*is called **elementary interval**, where $a_i, d_i \in \mathbb{N}$, $0 \leq a_i < b^{d_i}$, $1 \leq i \leq s$.*

For the volume, one obviously has

$$\lambda(E) = \prod_{i=1}^s b^{-d_i} = b^{-|\mathbf{d}|},$$

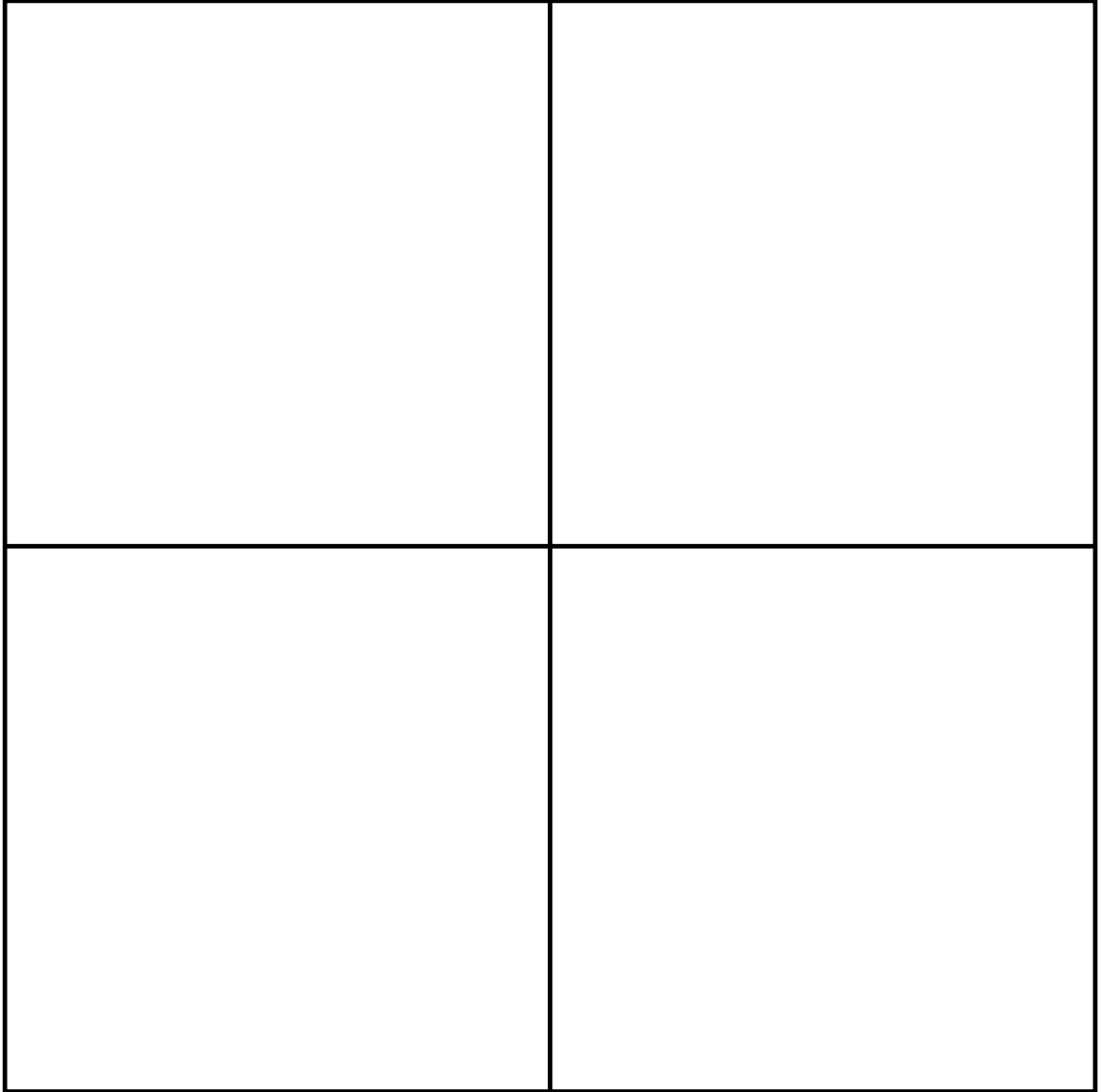


Figure 3.1: $b = 2, s = 2, |\mathbf{d}| = 2$.

where $\mathbf{d} = (d_i)_{i=1,\dots,s}$, $|\mathbf{d}| = \sum_{i=1}^s d_j$.

Definition 3.4.3 Let $0 \leq t \leq m$, $m, t \in \mathbb{N}$ and $b \geq 2$ be a base. A (t, m, s) -net in base b is a set $P \subseteq [0, 1]^s$ of b^m points such that

$$\#\{E \cap P\} = b^t \text{ for every elementary interval}$$

E in base b with $\lambda(E) = b^{t-m}$.

Example:

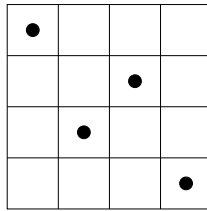


Figure 3.2: $b = 2, s = 2, t = 0, m = 2$.

Theorem 3.4.4 ([10] Thm. 4.10) One has

$$D_N^*(P) \leq B(s, b)b^t N^{-1}(\log N)^{s-1} + \mathcal{O}(b^t N^{-1}(\log N)^{s-2})$$

for $N = b^m$ and

$$B(s, b) = \begin{cases} \left(\frac{b-1}{2 \log b}\right)^{s-1}, & s = 2 \text{ or } b = 2, s = 3, 4, \\ \frac{1}{(s-1)!} \left(\frac{\lfloor b/2 \rfloor}{\log b}\right)^{s-1}, & \text{else.} \end{cases}$$

In the above sense, (t, m, s) -nets are optimal.

Remark 3.4.5 (a) For $m \geq 2$, a $(0, m, s)$ -net can only exist in the case

$$s \leq b + 1.$$

(b) There is an extension to infinite sequences, so-called (t, s) -sequences [10].

(c) Public software is available.

3.5 The Smolyak method

The Smolyak method is a deterministic method that has been used for various applications (under different names), not only for numerical cubature. The idea is starting from a 1D-method to construct an efficient n D-method using a clever setting of the grid points. In this regard, the Smolyak method can be seen as a special case of a QMC method. This is also known as *sparse grids* which is also a well-known method used for the numerical solution of partial differential equations.

Definition 3.5.1 Let $L_i[f] := \sum_{\nu=1}^{n_i} c_{\nu,i} f(x_{\nu,i})$, $x_{\nu,i} \in \mathbb{R}$, $i = 1, \dots, d$, be a linear functional, then

$$(L_1 \otimes \dots \otimes L_d)[f] := \sum_{\nu_1=1}^{n_1} \dots \sum_{\nu_d=1}^{n_d} c_{\nu_1,1} \dots c_{\nu_d,d} f(x_{\nu_1,1}, \dots, x_{\nu_d,d})$$

is called tensor product of the operators L_1, \dots, L_d .

Obviously, the standard product in formula (3.1.1) is of this form, i.e.,

$$Q_n^{[d]}[F] = \underbrace{(Q_n \otimes \dots \otimes Q_n)}_{d\text{-times}}[F] .$$

Remark 3.5.2 As already mentioned above, the concept of Smolyak can be applied to many problems having a product-structure. Here we only look at the particular application to quadrature, resp. cubature.

Definition 3.5.3 Let $Q^{(1)}, Q^{(2)}, \dots$, be a sequence of quadrature-formulae with n_i quadrature points and $Q^{(0)}[f] = 0$ (i.e. $n_0 = 0$) and set

$$\Delta^{(i)} := Q^{(i+1)} - Q^{(i)} . \quad (3.5.1)$$

Then

$$Q(k, d) := \sum_{|i| \leq k} \Delta^{(i_1)} \otimes \dots \otimes \Delta^{(i_d)}, \quad i = (i_1, \dots, i_d), \quad (3.5.2)$$

is called k -th Smolyak-Quadrature-Formula, where, as usual, we set

$$|i| := \sum_{\nu=1}^d i_\nu .$$

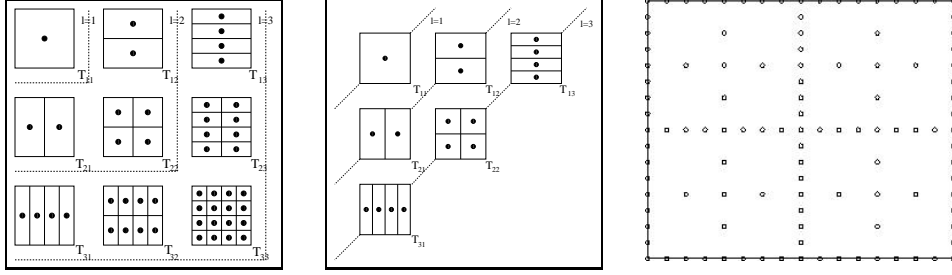


Figure 3.3: Idea for the building of a sparse grid (first two figures) and one particular example of a sparse grid.

This is shown in Figure 3.3 above.

Definition 3.5.4 A sequence $Q^{(1)}, Q^{(2)}, \dots$ of quadrature formulae with respect to the quadrature points $X^{(i)}$ is called nested, if $X^{(i)} \subseteq X^{(i+1)}$ holds for the sampling points $X^{(i)}, i = 1, 2, \dots$

To analyze the error of a Smolyak formula, we define the following space of functions of mixed maximal smoothness order $r \in \mathbb{N}$ on \mathbb{R}^d

$$\mathcal{F}_d^r := \left\{ g : \mathbb{R}^d \rightarrow \mathbb{R} : \|g\|_r := \left\| \frac{\partial^{|i|} g}{\partial x_1^{i_1} \dots \partial x_d^{i_d}} \right\| < \infty, \text{ if } i_\nu \leq r \right\}.$$

For a linear functional $L \in L(\mathcal{F}_d^r, \mathbb{R})$, consider the standard operator norm, namely

$$\|L\|_r := \sup_{0 \neq f \in \mathcal{F}_d^r} \frac{|L[f]|}{\|f\|_r}.$$

Now we start with 1D quadrature formulae satisfying the following error estimate

$$|R_n^{(i)}[f]| \leq \frac{c_r}{n_i^r} \|f\|_r, \quad f \in \mathcal{F}_1^r, \quad (3.5.3)$$

see Theorem 3.1.3.

Theorem 3.5.5 Let $Q^{(1)}, Q^{(2)}, \dots$ be nested such that (3.5.3) holds and the number of knots can be bounded by

$$a2^i \leq n_i \leq A2^i. \quad (3.5.4)$$

Then,

$$|R(k, d)[f]| \leq C_{d,r} \frac{(\log n(k, d))^{(d-1)(r+1)}}{n(k, d)^r} \|f\|_r, \quad (3.5.5)$$

holds for all $f \in \mathcal{F}_d^r$, where $n(k, d)$ is the number of quadrature points of $Q(k, d)$ and the constant $C_{d,r}$ does not depend on f .

For the proof we again need some preparations:

Lemma 3.5.6 *Let L_i be linear functionals on \mathcal{F}_1^r , then*

$$\|L_1 \otimes \cdots \otimes L_d\|_r = \|L_1\|_r \cdots \|L_d\|_r. \quad (3.5.6)$$

Proof: We proceed by induction over d . For $d = 1$ nothing has to be done. For $d \geq 2$ and $f \in \mathcal{F}_d^r$, we consider the following function

$$g := (L_2 \otimes \cdots \otimes L_d)[f] \in \mathcal{F}_1^r,$$

since one component is left free.

If the functions have the representation

$$L_i h = \sum_{\nu_i=1}^{n_i} C_{\nu_i,i} h(x_{\nu_i,i}),$$

we easily obtain

$$\begin{aligned} \frac{\partial}{\partial x_1} g(x_1) &= \sum_{\nu_2=1}^{n_2} \cdots \sum_{\nu_d=1}^{n_d} c_{\nu_2,2} \cdots c_{\nu_d,d} \frac{\partial}{\partial x_1} f(x_1, x_{\nu_2,2}, \dots, x_{\nu_d,d}) \\ &= (L_2 \otimes \cdots \otimes L_d) \left[\frac{\partial}{\partial x_1} f(x_1, \cdot, \dots, \cdot) \right] (x_{\nu_2,2}, \dots, x_{\nu_d,d}), \end{aligned}$$

where $h = \frac{\partial}{\partial x_1} f(x_1, \cdot, \dots, \cdot) \in \mathcal{F}_{d-1}^r$ with $\|h\|_r \leq \|f\|_r$. Then, using the induction hypothesis, we get

$$\|g\|_r \leq \|(L_2 \otimes \cdots \otimes L_d)\|_r \|f\|_r = \|L_2\|_r \cdots \|L_d\|_r \|f\|_r.$$

Now, we get

$$|(L_1 \otimes \cdots \otimes L_d)[f]| = |L_1[g]| \leq \|L_1\|_r \|g\|_r \leq (\|L_1\|_r \cdots \|L_d\|_r) \|f\|_r.$$

This finally implies

$$\|L_1 \otimes \cdots \otimes L_d\|_r \leq \|L_1\|_r \cdots \|L_d\|_r . \quad (3.5.7)$$

Using the definition of the operator norm, we get that for any $\varepsilon > 0$, there exists a function $f_i = f_i(\varepsilon)$ such that

$$L_i[f_i] \geq \|L_i\|_r \|f_i\|_r (1 - \varepsilon).$$

Using these functions f_i for all i , we set

$$f(u_1, \dots, u_d) := f_1(u_1) \cdots f_d(u_d) = (f_1 \otimes \cdots \otimes f_d)(u_1, \dots, u_d),$$

thus

$$\|f\|_r = \|f_1\|_r \cdots \|f_d\|_r$$

and

$$\begin{aligned} |(L_1 \otimes \cdots \otimes L_d)[f]| &= |L_1[f_1] \cdots L_d[f_d]| \\ &\geq \|L_1\|_r \cdots \|L_d\|_r \|f_1\|_r \cdots \|f_d\|_r (1 - \varepsilon)^d , \end{aligned}$$

which yields the assertion with (3.5.7) if we consider $\varepsilon \rightarrow 0+$. \square

The next result will be needed in order to estimate the number of cubature points which in turns is required to analyze the rate of convergence.

Lemma 3.5.7 *Under the hypothesis of Theorem 3.5.5 we have*

$$n(k, d) \leq A^d 2^{k+d} \binom{d+k-1}{d-1}.$$

Proof: Let $X(k, d)$ be the cubature points of $Q(k, d)$, $X^{(i)}$ the cubature points of $Q^{(i)}$ as before and $Y^{(i)}$ the cubature points of $\Delta^{(i)} = Q^{(i+1)} - Q^{(i)}$ in (3.5.1). Because of the nestedness we have

$$\#Y^{(i)} = \#\Delta^{(i)} = \#(Q^{(i+1)} - Q^{(i)}) = \#X^{(i+1)} ,$$

thus by the bound on n_i in (3.5.4) and the nestedness of the cubature points

$$\begin{aligned} \#X(k, d) &= \# \left(\sum_{|i|=k} Y^{(i_1)} \otimes \cdots \otimes Y^{(i_d)} \right) \\ &= \sum_{|i|=k} (\#X^{(i_1+1)}) \cdots (\#X^{(i_d+1)}) \\ &= \sum_{|i|=k} n_{i_1+1} \cdots n_{i_d+1} \\ &\leq \sum_{|i|=k} A \cdot 2^{i_1+1} \cdots A 2^{i_d+1} = \sum_{|i|=k} A^d 2^{|i|+d}. \end{aligned}$$

Because of

$$\#\{(i_1, \dots, i_d) \in \mathbb{N}^d : |i| = k\} = \binom{d+k-1}{d-1} = \binom{d+k-1}{k} \quad (3.5.8)$$

(for a proof see below) we conclude

$$\#X(k, d) \leq A^d 2^{k+d} \binom{d+k-1}{d-1}.$$

It remains to prove (3.5.8). We first show that

$$\sum_{n=0}^k \binom{n}{d} = \binom{k+1}{d+1} \quad (3.5.9)$$

for all $d \geq 1$ by induction. For $k = 0$ the claim holds because of $\binom{0}{d} = \binom{1}{d+1} = 0$ for all $d \geq 1$. For $k \geq 1$, we conclude by the induction hypothesis

$$\sum_{n=0}^{k+1} \binom{n}{d} = \binom{k+1}{d} + \sum_{n=0}^k \binom{n}{d} = \binom{k+1}{d} + \binom{k+1}{d+1} = \binom{k+2}{d+1}$$

so that (3.5.9) is shown.

Now let $N_k^d := \#\{(i_1, \dots, i_d) \in \mathbb{N}^d : |i| = k\}$, then obviously we have

$$N_k^1 = \#\{(k)\} = 1 = \binom{1+k-1}{1-1} = \binom{k}{0} = 1$$

and again by induction

$$N_k^d = \sum_{m=0}^k N_{k-m}^{d-1} = \sum_{m=0}^k N_m^{d-1} = \sum_{m=0}^k \binom{d-1+m-1}{d-2} = \binom{d+k-1}{d-1},$$

which proves (3.5.8) in view of (3.5.9). \square

Now one final auxiliary result in preparation for the proof of the main result.

Lemma 3.5.8 *Under the hypotheses of Theorem 3.5.5 we have*

$$\|R(k, d)\|_r \leq \tilde{C}_r 2^{-r(k+d)} (1+2^r)^{d-1} \binom{d+k}{d-1}.$$

Proof: Again by induction we obtain for a multi-index $i = (i_1, \dots, i_d)$

$$\begin{aligned} Q(k, d+1) &= \sum_{|i| \leq k} \left(\Delta^{(i_1)} \otimes \dots \otimes \Delta^{(i_d)} \otimes \sum_{\nu=0}^{k-|i|} \Delta^{(\nu)} \right) \\ &= \sum_{|i| \leq k} \left(\Delta^{(i_1)} \otimes \dots \otimes \Delta^{(i_d)} \otimes Q^{(k+1-|i|)} \right) \end{aligned}$$

thus

$$I_{d+1} - Q(k, d+1) = (I_d - Q(k, d)) \otimes I_1 + \sum_{|i| \leq k} \Delta^{(i_1)} \otimes \dots \otimes \Delta^{(i_d)} \otimes (I_1 - Q^{(k+1-|i|)}).$$

Because of Lemma 3.5.6, (3.5.3) and (3.5.4) we have by the triangle inequality

$$\begin{aligned} \|\Delta^{(i_\nu)}\|_r &= \|Q^{(i_\nu+1)} - Q^{(i_\nu)}\|_r \leq \|Q^{(i_\nu+1)} - I_1\|_r + \|Q^{(i_\nu)} - I_1\|_r \\ &= \|R_{n_{i_\nu+1}}^{(i_\nu+1)}\|_r + \|R_{n_{i_\nu}}^{(i_\nu)}\|_r \\ &\leq \frac{c_r}{n_{i_\nu+1}^r} + \frac{c_r}{n_{i_\nu}^r} \\ &\leq \frac{c_r}{a^r} (2^{-r(i_\nu+1)} + 2^{-ri_\nu}) = \frac{c_r}{a^r} 2^{-r(i_\nu+1)} (1 + 2^r). \end{aligned}$$

Next using $\|I_1\|_r = 1$ we have using the triangle inequality and Lemma 3.5.6

$$\begin{aligned} \|R(k, d+1)\|_r &= \|I_{d+1} - Q(k, d+1)\|_r \\ &\leq \|R(k, d)\|_r + \sum_{|i| \leq k} \|\Delta^{(i_1)} \otimes \dots \otimes \Delta^{(i_d)} \otimes R(k-|i|, 1)\|_r \\ &\leq \|R(k, d)\|_r + \sum_{|i| \leq k} \|\Delta^{(i_1)}\|_r \dots \|\Delta^{(i_d)}\|_r \|R(k-|i|, 1)\|_r \\ &\lesssim \|R(k, d)\|_r + \underbrace{\sum_{|i| \leq k} (1+2^r)^d \underbrace{2^{-r(|i|+d)} 2^{-r(k+1-|i|)}}_{=2^{-r(k+d+1)}}}_{=(1+2^r)^d 2^{-r(k+d+1)} \binom{d+k}{d}}. \end{aligned}$$

Now, finally

$$\begin{aligned}
\|R(k, d)\|_r &\lesssim \sum_{m=0}^{d-1} 2^{-r(k+d+1)} \underbrace{(1+2^r)^m}_{\leq (1+2^r)^{d-1}} \binom{m+k}{m} \\
&\leq 2^{-r(k+d+1)} (1+2^r)^{d-1} \underbrace{\sum_{m=0}^{d-1} \binom{m+k}{m}}_{\binom{d+k}{d-1}},
\end{aligned}$$

which completes the proof. \square

Now we have all tools at hand and come to the proof of the main result in this section.

Proof of Theorem 3.5.5: Set $q := d + k$. According to Lemma 3.5.7 we have for the number of cubature points

$$n(k, d) \leq A^d 2^{k+d} \binom{q-1}{d-1} \leq A^d 2^{k+d} \frac{q^{d-1}}{(d-1)!}, \quad (3.5.10)$$

thus in the worst case $q \sim \log n$ and $n \sim 2^{k+d} \frac{q^{d-1}}{(d-1)!}$, or, equivalently $2^{-(k+d)} \sim n^{-1} (\log n)^{d-1}$. Now it results from Lemma 3.5.8:

$$\begin{aligned}
\|R(k, d)\|_r &\leq \tilde{C}_r (1+2^r)^{d-1} 2^{-r(k+d)} \binom{q}{d-1} \\
&= C_{r,d} \underbrace{2^{-(k+d)(r+1)}}_{\sim n} \underbrace{2^{(k+d)} \binom{q}{d-1}}_{\sim n} \\
&\sim \left(\frac{(\log n)^{d-1}}{n} \right)^{r+1} \\
&\lesssim C_{r,d} \frac{(\log n)^{(d-1)(r+1)}}{n^r},
\end{aligned}$$

which proves the desired statement. \square

So far, the Smolyak method can be based upon any sequence of univariate quadrature formulae as long as the assumptions of Theorem 3.5.5 are satisfied. That still leaves some freedom to choose particular formulae, i.e., to choose particular sequences of quadrature points (knots). Let us now describe some well-known and widely used sequences.

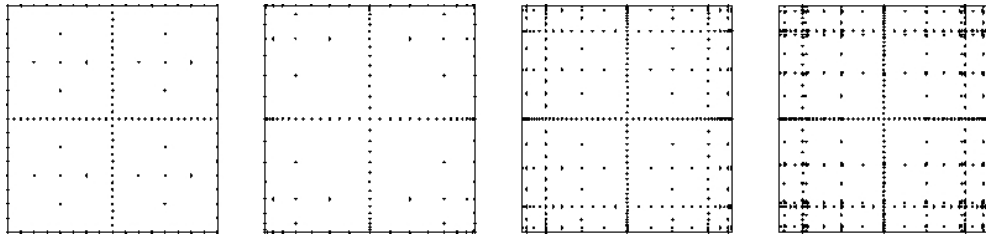


Figure 3.4: Sparse grids based on the trapezoidal, the Clenshaw-Curtis, Patterson and Gauss-Legendre rules, see also Table 3.1. This picture was taken from the homepage of T. Gerstner, Univ. Bonn.

3.5.1 Equidistant Points

Typically, one uses Newton-Cotes formulae in 1D, in the most simple situation, this is just the trapezoidal rule. Subdividing $[0, 1]$ into r_i subintervals results in

$$n_i = r_i + 1$$

knots. The Degree of Exactness (DoE) is then also $n + 1$.

3.5.2 Gauß-Points

As well-known, we obtain the optimal DoE of $2n_i - 1$ with n_i knots.

3.5.3 The Clenshaw-Curtis grids

These are widely used grids that have been introduced already in 1960. The CC-grids correspond to the settings

$$n_1 = 1, \quad n_k = 2^{k-1} + 1 \quad \text{for } k \geq 2,$$

[3]. For the cubature points $X^{CC}(k, 2)$ one typically chooses the roots of the Chebyshev orthogonal polynomials.

One further example is the Konrad-Patterson sequence, which uses the Stieltjes quadrature points. In Figure 3.4, we show the sparse grid points induced by the different constructions and Table 3.1 gives a summary of these.

Name	Abscissas	DoE
Newton-Cotes	equidistant	$n_i + 1$
Chenshow-Curtis	Chebyshev	$n_i - 1$
Patterson (1968)	Stieltjes	$\frac{3}{2}n_i - 1$
Gauss	Legendre	$2n_i - 1$

Table 3.1: Univariate quadrature formulae used for sparse grids construction.

Chapter 4

Numerical Computation of European Options

One key subject in mathematical finance is the modelling of stocks, derivatives and in particular option pricing. Nowadays there is a whole variety of different financial products, all of which require a careful mathematical modelling. Here we describe the numerical simulation of the pricing problem of European options that will lead us also to the numerical solution of (stochastic) partial differential equations.

4.1 Option Pricing: A Very Short Introduction

Since this is a lecture on *numerical* finance, we do not go into the details of the modelling of certain financial derivatives and refer to the lectures concerning mathematical finance. However, we give a very short introduction in order to describe which kind of mathematical problems occur in the option pricing problem. Here, we particularly focus on the description of the particular nature of problems that have to be treated numerically.

An option is a financial instrument depending on an *underlying* (e.g. a share, packets of shares, an index or a currency). Often options have a limited short term lifetime. The customer acquires the right to buy (Call) or sell (Put) the underlying for a previously agreed exercise price K (strike) at the date T (maturity). Let us collect some notation first.

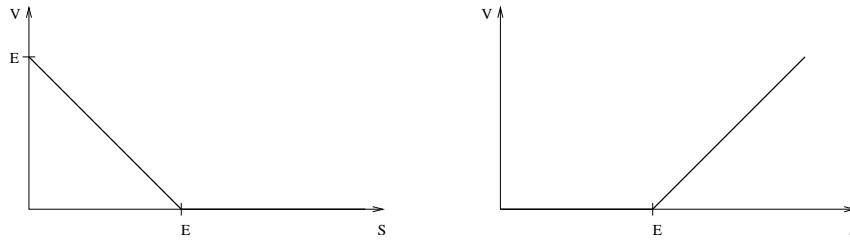


Figure 4.1: Payoff functions for a call (left) and a put (right) option.

- $S = S(t) = S_t$ denotes the stock price of the underlying;
- if an exercise of the option is only allowed at the date T , we talk of an *European option*;
- for a *call* there are two scenarios, namely if
 - $K < S = S(T)$: the option is exercised and the benefit is $S - K$;
 - $K \geq S$: the option will not be exercised and is hence worthless.

This shows that the *value* of the call at maturity can be described as

$$V(S, T) = \left\{ \begin{array}{ll} 0, & \text{if } S_T \leq K, \\ S_T - K, & \text{if } S_T > K \end{array} \right\} = (S_T - K)^+, \quad (4.1.1)$$

where $f^+ = \max\{f, 0\}$ is the broken power function. Often $V(S, T)$ is also called *payoff function*. Correspondingly, the payoff function of a put is given by

$$V(S, T) = (K - S_T)^+ = (S_T - K)^-. \quad (4.1.2)$$

Before we proceed, let us collect some standard notation. Typically $r > 0$ denotes the riskless interest rate which also reflects the return that can be gained e.g. for fixed-interested bonds. The margin of fluctuation of S is typically denoted by σ , which is known as the *volatility* (always per year).

Under certain assumptions (e.g. that S_T is lognormally distributed for details, we refer e.g. to [14]), this leads to the famous *Black-Scholes-Equation* for the value function V , which is the following differential equation

$$\frac{\partial}{\partial t}V(S, t) + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2}{\partial S^2}V(S, t) + rS \frac{\partial}{\partial S}V(S, t) - rV(S, t) = 0 \quad (4.1.3)$$

equipped with the *end condition*

$$V(S, T) = \text{“payoff” like (4.1.1) or (4.1.2)} \quad (4.1.4)$$

and *boundary conditions*

$$V(0, t) = 0, \quad V(S, t) \rightarrow S \text{ for } S \rightarrow \infty \text{ (for the Call)}. \quad (4.1.5)$$

This is a linear initial boundary value problem (PDE) for V . For (4.1.3, 4.1.4, 4.1.5) an analytical solution is known. However, when cost for deal (charges, taxes) k are also modelled, the additional non-linear term

$$-\sqrt{\frac{2}{\pi}} \frac{k\sigma S^2}{\sqrt{\sigma t}} \left| \frac{\partial^2}{\partial S^2} V \right|$$

is added on the left-hand side of (4.1.3). For this no analytical solution is known and one has to resort to numerical solution techniques.

The next sections are governed with different numerical methods for solving problems like the Black-Scholes equations. We start with the most simple ones.

4.2 Binomial Methods

Binomial methods are the first, very simple approach for the following special case of the above mentioned problem. In financial mathematics, it is also termed as *binomial model* since it is a simplified model for the option, in particular simpler than Black-Scholes. In many applications, the user is only interested in $V(S_0, 0)$, the today's value of the option with respect to the actual rate S_0 . One uses a simple tree-like structure to model the behaviour and to develop a solution method.

The first step is to introduce a discretization in time, i.e., the continuous interval $[0, T]$ is now splitted by introducing knots $t_i = i\Delta t$, $i = 0, \dots, M$. Here, M denotes the number of time steps and $\Delta t = \frac{T}{M}$ denotes the time step size. Then, $S(t)$ is approximated by (approximate) values $S_i := S(t_i)$ of S at the knots t_i .

The method relies on a number of assumptions which are now collected.

Assumption 4.2.1 (i) *Within a time period Δt of time, the value of S can only jump to uS ($u > 1$) or dS ($0 < d < 1$) (u means an increase of the rate -up-, d represents a decrease of the rate -down-);*

(ii) The probability for the increase of the stock is p (note that this is merely a notational assumption since p will drop out from the formulas);

(iii) The expected return corresponds to the riskfree rate of interest r , i.e.,

$$\mathbb{E}(S_{i+1}) = S_i e^{r\Delta t}. \quad (4.2.1)$$

(iv) No dividends are paid.

Note that sometimes the notation $1+u$, $1+d$ is used since this is consistent with $1+r$ in the standard model for the return.

The idea is now to compare expectation rates and variances for the continuous and the discrete model. Using (i) and (ii) in Assumption 4.2.1, we obtain by (4.2.1)

$$puS_i + (1-p)dS_i = \mathbb{E}(S_{i+1}) = S_i e^{r\Delta t},$$

so

$$e^{r\Delta t} = pu + (1-p)d. \quad (4.2.2)$$

For variances in the continuous model it holds (for details see exercises)

$$\mathbb{E}(S_{i+1}^2) = S_i^2 e^{(2r+\sigma^2)\Delta t},$$

thus

$$\begin{aligned} \text{Var}(S_{i+1}) &= \mathbb{E}(S_{i+1}^2) - \mathbb{E}(S_{i+1})^2 \\ &= S_i^2 e^{(2r+\sigma^2)\Delta t} - S_i^2 e^{2r\Delta t} \\ &= S_i^2 e^{2r\Delta t} (e^{\sigma^2\Delta t} - 1). \end{aligned}$$

In the discrete model we have

$$\text{Var}(S_{i+1}) = p(uS_i)^2 + (1-p)(dS_i)^2 - S_i^2 (pu + (1-p)d)^2,$$

so that we obtain by (4.2.2)

$$e^{2r\Delta t} (e^{\sigma^2\Delta t} - 1) = pu^2 + (1-p)d^2 - \underbrace{(pu + (1-p)d)^2}_{=e^{2r\Delta t}},$$

(where the assumption on the expectation is used) thus

$$e^{2r\Delta t + \sigma^2\Delta t} = pu^2 + (1-p)d^2. \quad (4.2.3)$$

Now (4.2.2, 4.2.3) are 2 equations for the 3 unknowns d, u, p and one additional equation is required to close the system. Often one uses (a little bit arbitrarily)

$$u \cdot d = 1. \quad (4.2.4)$$

One could also use $p = \frac{1}{2}$ which offers the advantage that the corresponding process is a Martingale in that case. If (4.2.4) holds, we have by (4.2.2) with the abbreviation $\alpha := e^{r\Delta t}$

$$\begin{aligned} \alpha &= pu + (1-p)\frac{1}{u} \\ &= p\left(u - \frac{1}{u}\right) + \frac{1}{u} \\ &= p\left(\frac{u^2 - 1}{u}\right) + \frac{1}{u}, \end{aligned}$$

hence

$$\begin{aligned} p &= \left(\alpha - \frac{1}{u}\right) \frac{u}{u^2 - 1} = \frac{\alpha u - 1}{u^2 - 1} \\ 1 - p &= \frac{u^2 - 1 - \alpha u + 1}{u^2 - 1} = \frac{u^2 - \alpha u}{u^2 - 1}. \end{aligned}$$

Thus, we obtain for the right-hand side of (4.2.3)

$$\begin{aligned} pu^2 + (1-p)d^2 &= \frac{\alpha u - 1}{u^2 - 1}u^2 + \frac{u^2 - \alpha u}{u^2 - 1} \frac{1}{u^2} \\ &= \frac{1}{u^2 - 1} \left[\alpha u^3 - u^2 + 1 - \frac{\alpha}{u} \right] \\ &= \alpha u - 1 + \frac{\alpha}{u} \end{aligned}$$

and with (4.2.3)

$$\begin{aligned} \alpha^2 e^{\sigma^2 \Delta t} &= \alpha u - 1 + \frac{\alpha}{u} \\ \iff u\alpha^2 e^{\sigma^2 \Delta t} &= \alpha u^2 - u + \alpha \\ \iff 0 &= \alpha u^2 - u(1 - \alpha^2 e^{\sigma^2 \Delta t}) + \alpha \\ \iff 0 &= u^2 - u \underbrace{(\alpha^{-1} + \alpha e^{\sigma^2 \Delta t})}_{=: 2\beta} + 1 \end{aligned} \quad (4.2.5)$$

which yields

$$u = \beta + \sqrt{\beta^2 - 1}, \quad d = \frac{1}{u} = \beta - \sqrt{\beta^2 - 1}, \quad (0 < d < 1 < u).$$

We may summarize our findings as follows

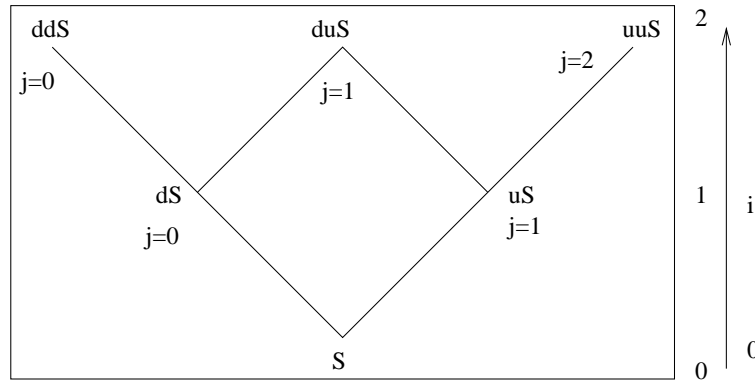
$$\begin{cases} \alpha = e^{r\Delta t} \\ \beta = \frac{1}{2}\left(\frac{1}{\alpha} + \alpha e^{\sigma^2 \Delta t}\right) \\ u = \beta + \sqrt{\beta^2 - 1} \\ d = \frac{1}{u} = \beta - \sqrt{\beta^2 - 1} \\ p = \frac{\alpha u - 1}{u^2 - 1} = \frac{\alpha - \frac{1}{u}}{u - \frac{1}{u}} = \frac{\alpha - d}{u - d}. \end{cases} \quad (4.2.6)$$

The algorithm consists of three different phases, namely the *forward phase*, the *valuation of the tree* and the *backward phase* which we now describe.

Forward phase: Calculation of the grid, initialization of the tree

- u and d are known, hence $S(t_1), \dots, S(t_M)$ can be computed, by using S_0 as the root of the tree;
- for every time $t_i, i = 1, \dots, M$ there are $i + 1$ possibilities as shown in the following figure.

$$S_{j,i} = S_0 u^j d^{i-j}, \quad j = 0, \dots, i \quad i = 1, 2, \dots, M. \quad (4.2.7)$$



On these grid points, we now compute approximate values for V , i.e., $V_{j,i} = V(t_i, S_{i,j})$, and we search for $V_{0,0} = V(t_0, S_0)$.

Evaluation of the tree: The value $V(S, t_M)$ of the option at the final time t_M is known through the final condition (the payoff function), i.e.

$$V_{j,M} = (S_{j,M} - K)^+ \text{ (Call)}, \quad V_{j,M} = (K - S_{j,M})^+ \text{ (Put)}. \quad (4.2.8)$$

Backward phase: compute $V_{j,i}$, $i = M-1, M-2, \dots, 0$ from $V_{j,M}$. Because of (4.2.2) we obtain

$$\begin{aligned} S_{j,i}e^{r\Delta t} &= puS_{j,i} + (1-p)dS_{j,i} \\ &= pS_{j+1,i+1} + (1-p)S_{j,i+1}, \end{aligned}$$

which we also transfer for V :

$$V_{j,i} = e^{-r\Delta t}(pV_{j+1,i+1} + (1-p)V_{j,i+1}). \quad (4.2.9)$$

This corresponds to the *Martingale property*, see Exercises.

Putting all pieces together, we obtain the following algorithm:

Algorithm 4.2.2 Input: r, σ, S_0, T, K, M , choice if Call or Put

- $\Delta t = \frac{T}{M}$, u, d, p like (4.2.6)
- $S_{0,0} := S_0$
- $S_{j,M} = S_{0,0}u^j d^{M-j}$, $j = 0, 1, \dots, M$, $S_{j,i}$ like (4.2.7), $i = 1, \dots, M$,
 $j = 0, \dots, i$
 $V_{j,M}$ like (4.2.8)
- for $i = M-1, \dots, 0$: $V_{j,i}$ like (4.2.9), $j = 0, \dots, i$

Output: $V_{0,0}$: as approximation for $V(S_0, 0)$.

This obvious advantage of binomial methods is their easy realization. On the other hand, the model is very simple and thus restricted.

4.3 Finite Difference Methods

Finite difference methods are maybe the simplest numerical methods for approximately solving ordinary and partial differential equations. In this section, we give an introduction to these methods focussing on their application to the above mentioned examples from mathematical finance.

Before we do so, we reformulate the boundary conditions in (4.1.5). The problem in (4.1.5) for the numerical treatment is the asymptotic behaviour of the boundary condition which cannot be handled directly. However, (4.1.5) can also be written as

$$V(0, t) = 0, \quad V(S_\infty, t) = S_\infty - K \quad (4.3.1)$$

for the call and

$$V(0, t) = K, \quad V(S_\infty, t) = 0 \quad (4.3.2)$$

for the put, where $K, S_\infty < \infty$. If we neglect the derivative with respect to time (i.e., we consider the *stationary process*), then (4.1.3) with boundary conditions like (4.3.1, 4.3.2) takes the following form

$$\begin{cases} Lu(x) := -(a(x)u'(x))' + b(x)u'(x) + c(x)u(x) = f(x), & x \in (0, 1), \\ u(0) = u(1) > 0, \end{cases} \quad (4.3.3)$$

where we assume $c(x) \geq 0$ for all x in order to ensure that a unique solution exists.

To be precise, the unknown is

$$u(x) \equiv V(S) = V(S, t) \quad \text{for fixed } t,$$

and the coefficients read

$$\begin{aligned} a(x) &= -\frac{1}{2}\sigma^2x^2 \\ b(x) &= (r - 2\sigma^2)x \\ c(x) &= -r \end{aligned}$$

and the right-hand side is $f \equiv 0$.

The most simple method for the numerical solution of (4.3.3) is the *classical finite difference method* in which the interval $(0, 1)$ is replaced by a set of *gridpoints* (or *nodes*) and the derivatives are approximated by differential quotients. For simplicity, we first consider an *equidistant grid*, i.e.,

$$x_i = i \cdot h, \quad N \in \mathbb{N}, \quad N \geq 1, \quad i = 0, \dots, N, \quad (4.3.4)$$

where $h = \frac{1}{N}$ denotes the *stepsize*. Then,

$$\omega_h := \{x_i = ih : i = 1, \dots, N - 1\} \quad (4.3.5)$$

denotes the set of *interior gridpoints*, $\gamma_h := \{x_0, x_N\}$ the *boundary points* and $\bar{\omega}_h := \omega_h \cup \gamma_h$ the *full grid*. For the approximation of the derivatives one uses

- the forward difference

$$(D^+u)(x) := \frac{1}{h}(u(x+h) - u(x)),$$

- the backward difference

$$(D^-(u)(x) := \frac{1}{h}(u(x) - u(x-h)),$$

- the symmetric or central difference

$$(D^0u)(x) := \frac{1}{2h}(u(x+h) - u(x-h))$$

- the second difference

$$(D^-D^+u)(x) := \frac{1}{h^2}(u(x+h) - 2u(x) + u(x-h)).$$

For the diffusion part with non-constant coefficients, i.e., $-(a(x)u'(x))'$ one usually uses

$$\begin{aligned} D^-(a(x)D^+u)(x) &= \frac{1}{h}(a(x)D^+u(x) - a(x-h)D^+u(x-h)) \\ &= \frac{1}{h^2}(a(x)u(x+h) - a(x)u(x) \\ &\quad - a(x-h)u(x) + a(x-h)u(x-h)) \\ &= \frac{1}{h^2}(a(x)u(x+h) - (a(x) + a(x-h))u(x) \\ &\quad + a(x-h)u(x-h)) \end{aligned}$$

or an approximation of the flux $a(x)u'(x)$ like in Numerik II.

Our next aim is to study the convergence of finite difference methods. As a preparation, we have

Lemma 4.3.1 *The following error estimates hold*

$$(i) \quad (D^0u)(x) = u'(x) + Rh^2 \text{ with } |R| \leq \frac{1}{6}\|u'''\|_{C[0,1]}, \text{ if } u \in C^3[0,1].$$

$$(ii) \quad (D^+D^-u)(x) = u''(x) + Rh^2 \text{ with } |R| \leq \frac{1}{12}\|u''''\|_{C[0,1]}, \text{ if } u \in C^4[0,1].$$

Proof: Using Taylor's formula, we have

$$\begin{aligned} u(x \pm h) &= u(x) \pm hu'(x) + h^2 \frac{u''(x)}{2} \pm h^3 R_3, \\ u(x \pm h) &= u(x) \pm hu'(x) + h^2 \frac{u''(x)}{2} \pm h^3 \frac{u'''(x)}{6} + h^4 R_4, \end{aligned}$$

with the following remainder terms

$$\begin{aligned} R_3 &= \frac{h^{-3}}{3!} \int_x^{x \pm h} [u''(\xi) - u''(x)](x \pm h - \xi) d\xi, \\ R_4 &= \frac{h^{-4}}{4!} \int_x^{x \pm h} \frac{1}{2} [u'''(\xi) - u'''(x)](x \pm h - \xi)^2 d\xi. \end{aligned}$$

This already yields the desired claim. \square

Remark 4.3.2 Note that the above estimates require $u \in C^3[0, 1]$ and $u \in C^4[0, 1]$, respectively.

Setting $g_i := g(x_i)$ for any function $g : [0, 1] \rightarrow \mathbb{R}$, and $Du_i := (Du)(x_i)$, we obtain the *classical finite difference method*

$$\begin{aligned} -D^- a_i D^+ u_i + b_i D^0 u_i + c_i u_i &= f_i, \quad i = 1, \dots, N-1, \\ u_0 &= u_N = 0. \end{aligned}$$

Due to

$$\begin{aligned} -D^- a_i D^+ u_i + b_i D^0 u_i + c_i u_i &= \\ &= \frac{1}{h^2} (-a_i u_{i+1} + (a_i + a_{i-1}) u_i - a_{i-1} u_{i-1}) \\ &\quad + \frac{b_i}{2h} (u_{i+1} - u_{i-1}) + c_i u_i \\ &= \underbrace{\left(-\frac{a_{i-1}}{h^2} - \frac{b_i}{2h} \right)}_{=:r_i} u_{i-1} + \underbrace{\left(\frac{a_i + a_{i-1}}{h^2} + c_i \right)}_{=:d_i} u_i + \underbrace{\left(-\frac{a_i}{h^2} + \frac{b_i}{2h} \right)}_{=:t_i} u_{i+1}, \end{aligned}$$

we obtain a *tridiagonal* system $L_h u_h = f_h$ to determine the unknown vector $u_h = (u_h(x_1), \dots, u_h(x_N)) = (u_{h,1}, \dots, u_{h,N-1})$, where the matrix L_h is given

by

$$L_h = \begin{bmatrix} d_1 & t_1 & & & \\ r_2 & d_2 & t_2 & & \\ & \ddots & \ddots & \ddots & \\ & & r_{N-2} & d_{N-2} & t_{N-2} \\ & & & r_{N-1} & d_{N-1} \end{bmatrix} \in \mathbb{R}^{(N-1) \times (N-1)}$$

and the right-hand side reads $f_h = \left(f(x_i) \right)_{i=1, \dots, N-1} = (f_{h,1})_{i=1, \dots, N-1}$.

In order to actually compute the approximation u_h , we obviously have to numerically solve a linear system of equations with a tridiagonal matrix. Using a direct method like Gauß would fill up the sparse matrix and yield an overall amount of work of $\mathcal{O}(N^3)$ for a $N \times N$ -tridiagonal matrix. Since the number of non-zero elements in the matrix is $\mathcal{O}(N)$, this is highly inefficient, especially when N grows. One could resort to iterative methods, but for tridiagonal matrices there is also a very efficient direct method. Note that this approach is restricted to the 1D-case $x \in [0, 1] \subset \mathbb{R}$ only!

4.3.1 A recursion method for tridiagonal matrices

The idea to solve a tridiagonal system of the form $L_h u_h = f_h$ is that the LU-decomposition of a tridiagonal matrix can be given explicitly, namely

$$L_h = \begin{pmatrix} \alpha_1 & & & & & & 0 \\ r_2 & \alpha_2 & & & & & \\ & \ddots & \ddots & & & & \\ & & \ddots & \ddots & & & \\ & & & r_{N-2} & \alpha_{N-2} & & \\ 0 & & & & r_{N-1} & \alpha_{N-1} & \end{pmatrix} \begin{pmatrix} 1 & \gamma_1 & & & & & 0 \\ & \ddots & \ddots & & & & \\ & & \ddots & \ddots & & & \\ & & & \ddots & \ddots & & \\ & & & & \ddots & \ddots & \\ & & & & & 1 & \gamma_{N-2} \\ 0 & & & & & & 1 \end{pmatrix} \quad (\alpha_i \neq 0)$$

where

$$\begin{aligned} \alpha_1 &= d_1, \\ \gamma_i &= \frac{t_i}{\alpha_i}, \quad i = 1, \dots, N-2, \\ \alpha_i &= d_i - \gamma_{i-1} r_i, \quad i = 2, \dots, N-1. \end{aligned}$$

We leave the proof as an exercise. From this we obtain the following recursion

$$\begin{aligned} v_1 &= \frac{f_1}{\alpha_1}, & v_i &= \frac{1}{\alpha_i} (f_i - r_i v_{i-1}), & i &= 2, \dots, N-1, \\ u_{N-1} &= v_{N-1} & u_i &= v_i - \gamma_i u_{i+1} & i &= N-2, \dots, 1. \end{aligned}$$

One can easily see that this method amounts to $\mathcal{O}(N)$ operations which clearly is optimal.

4.3.2 Convergence Theory

Setting as usual

$$\|v\|_\infty := \max_{1 \leq i \leq n} |v_i|, \quad v := (v_1, \dots, v_n)^T \in \mathbb{R}^n,$$

we have

Definition 4.3.3 (i) *The finite difference method is called convergent of order k , if there exists $C > 0$ such that*

$$\|R_h u - u_h\|_\infty \leq Ch^k$$

where $R_h : C[0, 1] \rightarrow \mathbb{R}$, $R_h u = (u(x_1), \dots, u(x_{N-1}))$ is the restriction of the exact solution to the computational grid and $u_h = (u_1, \dots, u_{N-1})$ denotes the numerical approximation.

(ii) *The finite difference method is called consistent of order k (with respect to $\|\cdot\|_\infty$) if there exist $C > 0$ with*

$$\|L_h R_h u - R_h L u\|_\infty \leq Ch^k .$$

(iii) *The method is called stable if $L_h u_h = f_h$ always implies the estimate $\|u_h\|_\infty \leq C \|R_h f\|_\infty$ (continuous dependence of the solution on the data).*

Now we can give the first result which is essential for the convergence analysis.

Theorem 4.3.4 *If $u \in C^4[0, 1]$, then the classical finite difference method is consistent of order 2.*

Proof: An easy calculation shows

$$\begin{aligned} L_h R_h u - R_h L u &= \frac{1}{h^2} (-a(x_i - h)u(x_i + h) + (a(x_i) + a(x_{i-1}))u(x_i) \\ &\quad - a(x_i - h)u(x_i - h)) \\ &\quad + b(x_i) \frac{1}{2h} (u(x_i + h) - u(x_i - h)) + c(x_i)u(x_i) \\ &\quad + (a(x_i)u'(x_i))' - b(x_i)u'(x_i) - c(x_i)u(x_i) \\ &= (a(x_i)u'(x_i))' - (D^- a D^- u)u(x_i) \\ &\quad + b(x_i)(D^0 u(x_i) - u'(x_i)) , \end{aligned}$$

so that we obtain by Lemma 4.3.1 the estimate

$$|(L_h R_h u - R_h L u)(x_i)| \leq \frac{1}{12} \|a\|_{C[0,1]} h^2 \|u''''\|_{C[0,1]} + \frac{1}{6} h^2 \|b\|_{C[0,1]} \|u''''\|_{C[0,1]}$$

which proves the assertion. \square

Remark 4.3.5 *The proof of Lemma 4.3.1 shows that the statement of Theorem 4.3.4 also holds if u'''' is only Lipschitz continuous, i.e., $u \in C^{3,1}[0, 1]$.*

Remark 4.3.6 *It is a central statement of the analysis of finite difference methods that consistency and stability imply convergence of the particular method. However, a rigorous proof of this goes far beyond the scope of the present lecture.*

Remark 4.3.7 *Again it should be noted that the above regularity assumptions ($u \in C^4[0, 1]$ or $u \in C^{3,1}[0, 1]$) pose serious restrictions. An alternative is the weak formulation and Finite Elements.*

4.4 Discretization in Time

Now, we consider the time dependent problem (where here for simplicity we assume $a(x) \equiv 1, b(x) = c(x) = 0$): determine $u(x, t)$, with $x \in (0, 1)$, $t \in (0, T)$ such that

$$\begin{cases} \frac{\partial}{\partial t} u(x, t) - \frac{\partial^2}{\partial x^2} u(x, t) = f(x, t) & \text{in } (0, 1) \times (0, T), \\ u(x, 0) = u_0(x), & x \in (0, 1) \\ u(0, t) = u_1(t), & u(1, t) = u_2(t), \quad t \in (0, T). \end{cases} \quad (4.4.1)$$

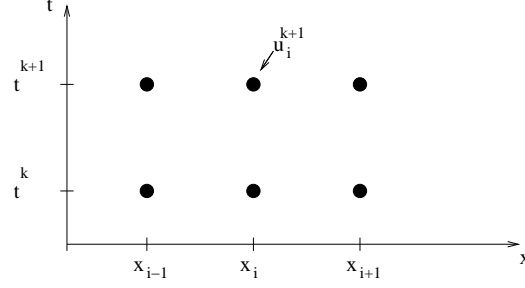
Problem (4.4.1) is known as *heat equation*.

The simple idea is to use a finite difference method both with respect to space *and* time:

$$\begin{aligned} x_i &= ih, & i &= 0, \dots, N, & h &= \frac{1}{N}, \\ t^k &= k\Delta t, & k &= 0, \dots, M, & \Delta t &= \frac{T}{M}. \end{aligned}$$

Setting $f_i^k := f(x_i, t^k)$, we are looking for an approximation u_i^k of $u(x_i, t^k)$. Note that we always use a subscript to denote the discretization index in space and a superscript for the time discretization.

The grid points can be visualized as follows



For a fixed time t^k we again consider

$$D^- D^+ u_i^k := \frac{1}{h^2} (u_{i-1}^k - 2u_i^k + u_{i+1}^k)$$

and define a *six point scheme* (with a free parameter $0 \leq \sigma \leq 1$) by

$$\begin{aligned} \frac{1}{\Delta t} (u_i^{k+1} - u_i^k) &= D^- D^+ (\sigma u_i^{k+1} + (1 - \sigma) u_i^k) + \tilde{f}_i^k, & (4.4.2) \\ i &= 1, \dots, N - 1, \quad k = 1, \dots, M - 1 \\ u_i^0 &= u_0(x_i), \\ u_0^k &= u_1(t^k), \quad u_N^k = u_2(t^k), \end{aligned}$$

where \tilde{f}_i^k denotes an approximation of $f(x_i, t^k)$ (e.g. $\tilde{f}_i^k = f_i^k$).

For certain choices of σ , we obtain the following important special cases:

(i) **Explicit Euler method:** (with $\gamma := \frac{\Delta t}{h^2}$) : $\sigma = 0$, $\tilde{f}_i^k := f_i^k$:

$$u_i^{k+1} = (1 - 2\gamma)u_i^k + \gamma(u_{i-1}^k + u_{i+1}^k) + \Delta t f_i^k$$

(ii) Purely **implicit Euler method:** $\sigma = 1$, $\tilde{f}_i^k := f_i^k$:

$$(1 + 2\gamma)u_i^{k+1} - \gamma(u_{i+1}^{k+1} + u_{i-1}^{k+1}) = u_i^k + \Delta t f_i^k$$

(iii) **Crank-Nicolson method:** $\sigma = \frac{1}{2}$, $\tilde{f}_i^k := f(x_i, t^k + \frac{\Delta t}{2})$

$$\begin{aligned} 2(\gamma + 1)u_i^{k+1} - \gamma(u_{i+1}^{k+1} + u_{i-1}^{k+1}) &= 2(1 - \gamma)u_i^k + \gamma(u_{i+1}^k + u_{i-1}^k) \\ &\quad + 2\Delta t f(x_i, t^k + \frac{\Delta t}{2}). \end{aligned}$$

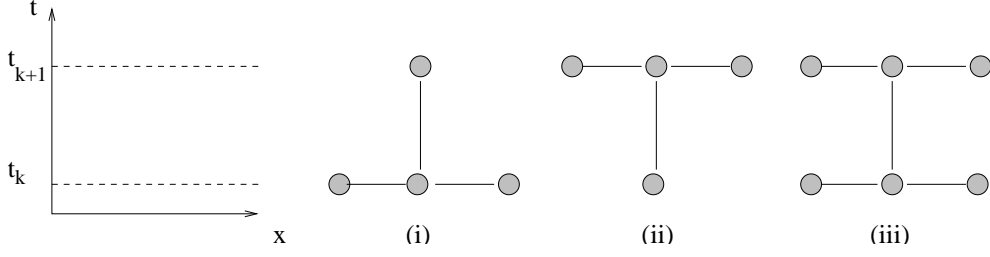


Figure 4.2: Finite difference methods in time for the special choices of σ .

In (i), the approximation $(u_i^{k+1})_{i=1,\dots,N-1}$ on the new time level can be computed directly from the data $(u_i^k)_{i=1,\dots,N-1}$ on the previous time step. In the other two cases, one has to solve a linear system of equations to proceed in time. The three variants are shown in Figure 4.2.

Theorem 4.4.1 *For the consistency error on $Q := (0, 1) \times (0, T)$, we obtain the following orders*

(i) $\mathcal{O}(h^2 + \Delta t)$ for arbitrary σ , $\tilde{f}_k^i = f(x_i, t^k)$ and for $u \in C^{4,2}(\bar{Q})$.

(ii) $\mathcal{O}(h^2 + \Delta t^2)$ for the Crank-Nicholson method for $u \in C^{4,3}(\bar{Q})$.

Proof: We only prove (ii) here. Using Taylor's formula, we obtain

$$\frac{1}{\Delta t}(u(x, t + \Delta t) - u(x, t)) = u_t(x, t) + \frac{1}{2}u_{tt}(x, t)\Delta t + \mathcal{O}(\Delta t^2)$$

as well as

$$\begin{aligned} \frac{1}{2h^2}\{ & u(x-h, t + \Delta t) - 2u(x, t + \Delta t) + u(x+h, t + \Delta t) \\ & - u(x-h, t) + 2u(x, t) - u(x+h, t)\} = \\ & = \frac{1}{2}\{2u_{xx} + \Delta t u_{xxt} + \mathcal{O}(\Delta t^2 + h^2)\}, \\ f(x, t + \frac{\Delta t}{2}) & = f(x, t) + \frac{\Delta t}{2}f_t + \mathcal{O}(\Delta t^2). \end{aligned}$$

Thus, we obtain for the consistency error

$$C_{\text{cons}} = \underbrace{u_t - u_{xx} - f + \frac{1}{2}\Delta t(u_{tt} - u_{xxt} - f_t)}_{=0} + \mathcal{O}(\Delta t^2 + h^2)$$

$$\text{since } u_t = u_{xx} + f \Rightarrow u_{tt} = u_{xxt} - f_t,$$

which proves the theorem. \square

Now, in view of Remark 4.3.6, we come to the analysis of the stability.

Theorem 4.4.2 *We have*

$$\max_k \max_i |u_i^{k+1}| \leq \max_x |u_0(x)| + \Delta t \sum_{j=0}^k \max_i |\tilde{f}_i^j|,$$

i.e., the method is stable with respect to the discrete supremum-norm, provided that $1 - 2(1 - \sigma)\gamma \geq 0$.

Proof: Rewrite (4.4.2) in the following form

$$-\gamma\sigma u_{i-1}^{k+1} + (2\sigma\gamma + 1)u_i^{k+1} - \sigma\gamma u_{i+1}^{k+1} = F_i^k,$$

where the right-hand side only contains known terms, i.e.,

$$F_i^k := (1 - \sigma)\gamma u_{i-1}^k + (1 - 2(1 - \sigma)\gamma)u_i^k + (1 - \sigma)\gamma u_{i+1}^k + \Delta t \tilde{f}_i^k.$$

For simplicity we consider homogeneous boundary conditions, i.e., $u(0) = u(1) = 0$. Since the matrix

$$A = \begin{bmatrix} 2\sigma\gamma + 1 & -\gamma\sigma & & 0 \\ -\gamma\sigma & \ddots & \ddots & \\ & \ddots & \ddots & -\gamma\sigma \\ 0 & -\gamma\sigma & & 2\sigma\gamma + 1 \end{bmatrix}$$

is *strictly diagonal dominant*, i.e.,

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^n |a_{ij}|$$

we obtain

$$\|A^{-1}\| \leq \frac{1}{\min_k \left(a_{kk} - \sum_{j \neq k} |a_{j,k}| \right)} = 1.$$

We leave the proof of the latter inequality as an exercise. Then we have $\max_i |u_i^{k+1}| \leq \max_i |F_i^k|$ as well as

$$\max_i |F_i^k| \leq \max_i |u_i^k| + \Delta t \max_i |\tilde{f}_i^k|$$

which is now applied to $k, k-1, \dots, 0$. \square

Now we obtain the following error estimates.

Theorem 4.4.3 *Let $(1 - \sigma)\frac{\Delta t}{h^2} \leq \frac{1}{2}$ and $u \in C^{4,2}(\bar{Q})$ and $\tilde{f}_i^k = f(x_i, t^k)$, the we have*

$$\max_{i,k} |u(x_i, t^k) - u_i^k| \leq C(h^2 + \Delta t) .$$

For the Crank-Nicholson method ($\sigma = \frac{1}{2}$) we have for $\frac{\Delta t}{h^2} \leq 1$ the estimate

$$\max_{i,k} |u(x_i, t^k) - u_i^k| \leq C(h^2 + \Delta t^2) .$$

Proof: The statement is an immediate consequence of the Theorems 4.4.1 and 4.4.2. \square

Remark 4.4.4 *The stability condition $(1 - \sigma)\frac{\Delta t}{h^2} \leq \frac{1}{2}$ is always valid for the purely implicit method ($\sigma = 1$). For $\sigma \neq 1$ this condition yields a restriction for the relation of the stepsize with respect to time and space.*

Chapter 5

Stochastic Differential Equations (SDE)

Many applications — not only in finance — lead to a model that also includes probabilistic effects. We have seen one example in option pricing where e.g. a model for the underlying of the form

$$dS = rS dt + \sigma S dW$$

was used. In this chapter, we give an introduction to numerical methods for SDEs.

5.1 Introduction to SDEs

There are two main classes of equations that include random effects in differential equations, namely

- random differential equations (RDE),
- stochastic differential equations (SDE).

An RDE takes the form

$$\dot{x}(t) = \frac{d}{dt}x(t) = a(\omega)x(t) + b(t, \omega), \quad x(t) = x(t, \omega) \quad (5.1.1)$$

where $a(\cdot)$ and $b(t, \cdot)$ are random variables and b is continuous for all t and ω . For a given initial value $x(\theta, \omega) = x_0(\omega)$, the solution is given as

$$x(t, \omega) = e^{a(\omega)t} \left(x_0(\omega) + \int_0^t e^{-a(\omega)s} b(s, \omega) ds \right) \quad (5.1.2)$$

i.e., the sample paths are obviously differentiable with respect to t . Hence, RDEs are solved sample by sample and this also holds for corresponding numerical methods. I.e., as soon as an $\omega \in \Omega$ is realized, any numerical method for ordinary differential equations can be used.

The solution of an SDE however, inherits the nondifferentiability of sample paths from the stochastic process. In many applications, SDE model the random fluctuation in the dynamics of the system.

Example 5.1.1 *If X_t denotes the velocity of a particle in one direction, the Langerin equation reads*

$$\frac{d}{dt} X_t = -aX_t + b\xi_t, \quad (5.1.3)$$

where aX_t denotes the velocity-dependent force and $b\xi_t$ the molecular force with intensity b driven by a white noise process ξ_t . In this model, it is assumed that external forces do not depend on the state X_t of the system. Symbolically, (5.1.3) is written as a SDE as

$$dX_t = -aX_t dt + bW_t \quad (5.1.4)$$

which is a short-hand notation for the stochastic integral equation

$$X_t = X_{t_0} - \int_{t_0}^t aX_s ds + \int_{t_0}^t b dW_s, \quad (5.1.5)$$

where the second integral is an Itô stochastic integral.

The above example is a particular case of the general SDE

$$dX_t = a(X_t) dt + b(X_t) dW_t \quad (5.1.6)$$

where the $b(X_t) \equiv b$ models *additive* noise, otherwise *multiplicative* noise. Again, (5.1.6) is a short-hand notation of the integral equation

$$X_t = X_{t_0} + \int_{t_0}^t a(X_s) ds + \int_{t_0}^t b(X_s) dW_s. \quad (5.1.7)$$

In the case of constant or linear coefficients, i.e., $b(X_s) \equiv b$ and $b(X_s) = bX_s$, respectively, the Itô calculus can be used to derive analytical solutions, namely

$$X_t = e^{-at} X_0 + e^{-at} \int_0^t e^{as} b dW_s, \quad b(X_s) \equiv b, \quad (5.1.8)$$

$$X_t = X_0 \exp \left\{ \left(a - \frac{1}{2} b^2 \right) t + bW_t \right\}, \quad b(X_s) = bX_s. \quad (5.1.9)$$

These solutions are *pathwise unique*, i.e.,

$$P \left(\sup_{0 \leq t \leq T} |\tilde{X}_t - X_t| > 0 \right) = 0, \quad \forall t > 0, \quad (5.1.10)$$

where \tilde{X}_t is any other solution with the same initial value and continuous sample paths.

Definition 5.1.2 (a) *If the Wiener process $W_t, t \geq 0$ is given, the solution X_t of the SDE is called a strong solution.*

(b) *If we are given $a(\cdot)$ and $b(\cdot)$, a point (X_t, W_t) (i.e., we can choose an appropriate Wiener process) is called weak solution.*

Remark 5.1.3 (a) *Some SDEs have no strong but only weak solutions.*

(b) *Obviously, (5.1.8) and (5.1.9) correspond to special cases of constant and linear coefficients. In the general situation, no closed formula for the solution exists, one has to resort to numerical methods.*

5.2 Existence and uniqueness of strong solutions

We consider the general SDE

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t, \quad (5.2.1)$$

i.e.,

$$X_t = X_{t_0} + \int_{t_0}^t a(x, X_s)ds + \int_{t_0}^t b(x, X_s)dW_s. \quad (5.2.2)$$

In order to obtain existence and uniqueness, we pose the following assumptions

A1 (Measurability): $a = a(t, x)$ and $b = b(t, x)$ are jointly L_2 -measurable in $(t, x) \in [t_0, T] \times \mathbb{R}$.

A2 (Lipschitz condition): There exists a constant $K > 0$ such that

$$\begin{aligned} |a(t, x) - a(t, y)| &\leq K(x - y) \\ |b(t, x) - b(t, y)| &\leq K(x - y) \end{aligned}$$

for all $t \in [t_0, T]$ and $x, y \in \mathbb{R}$.

A3 (Linear growth bound): There exists a constant $K > 0$ such that

$$|a(t, x)|^2 \leq K^2(1 + |x|^2), \quad |b(t, x)|^2 \leq K^2(1 + |x|^2)$$

for all $t \in [t_0, T]$ and $x, y \in \mathbb{R}$.

A4 (Initial value): X_{t_0} is \mathcal{A}_{t_0} -measurable with $E(|X_{t_0}|^2) < \infty$.

Theorem 5.2.1 (a) *If A1 and A2 hold, the solutions to (5.2.2) correspond to the same initial value and the same Wiener process are pathwise unique.*

(b) *Under assumption A1–A4 the SDE (5.2.1) has a pathwise unique strong solution X_t on $[t_0, T]$ with*

$$\sup_{t_0 \leq t \leq T} E(|X_t|^2) < \infty.$$

Proof: (a) Using the Gronwall lemma, (b) with a method of successive approximations. For details, we refer to [7], 129–134. \square

Remark 5.2.2 *Under some stronger assumptions, one can obtain Gronwall-like estimates for moments of X_t , in particular the existence of such moments. One can also show stability results, i.e., the continuous dependence of the solution of a SDE on the data.*

Remark 5.2.3 *For special cases, one can show even stronger results e.g. for diffusion processes, or certain linear SDEs.*

5.3 Stochastic Taylor Expansions

Several numerical methods are based upon a Taylor expansion, e.g. Newton’s method or some single step methods for ordinary differential equations (ode). Hence, we now describe the stochastic analogon of the classical Taylor expansion.

In order to do so, we review the classical Taylor expansion in a slightly different way. Let X_t be the solution of the initial value problem

$$\frac{d}{dt}X_t = a(X_t), \quad t \in [t_0, T], \quad X_{t_0} = x_0, \quad (5.3.1)$$

i.e.,

$$X_t = X_{t_0} + \int_{t_0}^t a(X_s) ds. \quad (5.3.2)$$

Now, consider a function $f : \mathbb{R} \rightarrow \mathbb{R}$. The chain rule gives

$$\frac{d}{dt}f(X_t) = \frac{d}{dt}X_t \cdot \frac{\partial}{\partial x}f(X_t) = a(X_t) \frac{\partial}{\partial x}f(X_t), \quad (5.3.3)$$

i.e.,

$$f(X_t) = f(X_{t_0}) + \int_{t_0}^t Lf(X_s) ds, \quad (5.3.4)$$

with the operator $L := a \frac{\partial}{\partial x}$. Now, we can apply this formula for $f = L = \frac{\partial}{\partial x}$

$$\begin{aligned} \int_{t_0}^t Lf(X_s) ds &= \int_{t_0}^t \left\{ Lf(X_{t_0}) + \int_{t_0}^s L(Lf)(X_z) dz \right\} ds \\ &= (t - t_0)Lf(X_{t_0}) + \int_{t_0}^t \int_{t_0}^s L^2 f(X_z) dz ds \end{aligned}$$

and again

$$\begin{aligned} \int_{t_0}^t \int_{t_0}^s L^2 f(X_z) dz ds &= \int_{t_0}^t \int_{t_0}^s \left\{ L^2 f(X_{t_0}) + \int_{t_0}^z L^3 f(X_u) du \right\} dz ds \\ &= L^2 f(X_{t_0}) \int_{t_0}^t (s - t_0) ds + \mathcal{R}_3 \\ &= \frac{1}{2} (t - t_0)^2 L^2 f(X_{t_0}) + \mathcal{R}_3 \end{aligned}$$

with the remainder

$$\mathcal{R}_3 = \int_{t_0}^t \int_{t_0}^s \int_{t_0}^z L^3 f(X_u) du dz ds.$$

Proceeding like this results in the classical **Taylor formula in integral form**

$$\begin{aligned} f(X_t) &= f(X_{t_0}) + \sum_{\ell=1}^r \frac{(t - t_0)^\ell}{\ell!} L^\ell f(X_{t_0}) \\ &\quad + \int_{t_0}^t \int_{t_0}^{s_{r+1}} \dots \int_{t_0}^{s_2} L^{r+1} f(X_{s_1}) ds_1 \dots ds_{r+1}. \end{aligned} \tag{5.3.5}$$

The choice $L \equiv \frac{\partial}{\partial x}$ results in the well-known formula. In order to develop a stochastic counterpart we consider

- X_t to be an Itô process

- the stochastic counterpart of the chain rule, namely the Itô formula.

Theorem 5.3.1 (Itô formula) *Let $U : [0, T] \times \mathbb{R} \rightarrow \mathbb{R}$ have continuous partial derivatives $\frac{\partial U}{\partial t}, \frac{\partial U}{\partial x}, \frac{\partial^2 U}{\partial x^2}$. Moreover, let $Y_t = U(t, X_t)$, $0 \leq t \leq T$, where X_t satisfies*

$$dX_t(\omega) = e(t, \omega)dt + f(t, \omega)dW_t(\omega)$$

with $\sqrt{|e|}, f \in \mathcal{L}_T^\omega := \{g : g \text{ is jointly } \mathcal{L} \times \mathcal{A}\text{-measurable, } g(t, \cdot) \text{ is } \mathcal{A}_t\text{-measurable } \forall t \in [0, T] \text{ and } \int_0^T g(s, \omega)^2 ds < \infty \text{ w.p. } 1\}$. Then,

$$\begin{aligned} Y_t - Y_s &= \int_s^t \left\{ \frac{\partial U}{\partial t}(u, X_u) + e_u \frac{\partial U}{\partial x}(u, X_u) + \frac{1}{2} f u^2 \frac{\partial^2 U}{\partial x^2}(u, X_u) \right\} du \\ &\quad + \int_s^t f u \frac{\partial U}{\partial x}(u, X_u) dW_u \end{aligned} \quad (5.3.6)$$

w.p.1 for any $0 \leq s \leq t \leq T$.

Proof: [7], 92–95. □

The deterministic ode in (5.3.2) is replaced by the SDE

$$X_t = X_{t_0} + \int_{t_0}^t a(X_s) ds + \int_{t_0}^t b(X_s) dW_s, \quad (5.3.7)$$

where a and b are smooth real-value functions with linear growth. Similar as above (exercise), one deduces the following *Itô expansion* for $f \in C^2(\mathbb{R})$:

$$f(X_t) = f(X_{t_0}) + \int_{t_0}^t L^0 f(X_s) ds + \int_{t_0}^t L^1 f(X_s) s W_s \quad (5.3.8)$$

for $t \in [t_0, T]$, where

$$L^0 = a \frac{\partial}{\partial x} + \frac{b^2}{2} \frac{\partial^2}{\partial x^2}, \quad L^1 = b \frac{\partial}{\partial x}. \quad (5.3.9)$$

Generalizations and more details can be found in [7] and also in [14].

5.4 The Euler–Maruyama approximation

Recall the explicit Euler method for the initial value problem

$$\frac{d}{dt}X_t = a(t, X_t) \quad t \in [t_0, T], \quad X_{t_0} = x_0,$$

which reads

$$Y_{n+1} = Y_n + \Delta_n a(t_n, Y_n), \quad Y_0 = x_0,$$

where $t_0 = \tau_0 < \tau_1 < \dots < \tau_n < \dots < \tau_N = T$ is a discretization in time and

$$\Delta_n := \tau_{n+1} - \tau_n \tag{5.4.1}$$

is the time step. Then, $Y_n \approx X(\tau_n)$ and the explicit Euler scheme converges of linear order.

Let us consider an analogous approach for the SDE

$$dX_t = a(t, X_t)dt + b(t, X_t)dW_t,$$

which is called *Euler Maruyama method*.

$$Y_{n+1} = Y_n + a(\tau_n, Y_n)\Delta_n + b(\tau_n, Y_n)(W_{\tau_{n+1}} - W_{\tau_n}), \quad Y_0 = X_0. \tag{5.4.2}$$

The main difference to the deterministic Euler method is that we now need to generate the random increments

$$\Delta W_n := W_{\tau_{n+1}} - W_{\tau_n}, \tag{5.4.3}$$

e.g. by a random number generator. If W is a Wiener process, these increments are independent Gaussian with

$$E(\Delta W_n) = 0, \quad \text{Var}(\Delta W_{\tau_n}) = \Delta_n. \tag{5.4.4}$$

Example 5.4.1 *In order to illustrate some of the main features of this numerical method consider X_t as the solution of the linear SDE*

$$dX_t = aX_t dt + bX_t dW_t, \tag{5.4.5}$$

i.e., drift $a(t, x) = ax$ and diffusion $b(t, x) = bx$. Note that (5.4.5) has the explicit solution

$$X_t = X_0 \exp \left\{ \left(a - \frac{b^2}{2} \right) t + bW_t \right\} \tag{5.4.6}$$

for a given Wiener process W_t which allows for comparisons of exact and numerical solution. In fact, the Euler scheme reads

$$Y_{n+1} = Y_n + aY_n\Delta_n + bY_n\Delta W_n, \quad \Delta W_n \sim \mathcal{N}(0, \Delta_n). \quad (5.4.7)$$

On the other hand, (5.4.6) gives

$$X_{\tau_n} = X_0 \cdot \exp \left\{ \left(a - \frac{b^2}{2} \right) \tau_n + b \sum_{i=1}^n \Delta W_{i-1} \right\} \quad (5.4.8)$$

for the exact solution.

When performing this example with two values for the time step parameter

$$\delta := \max_n \Delta_n \quad (5.4.9)$$

with Δ_n defined by (5.4.1) e.g. for an equivalent subdivision with

$$\delta = \Delta_n \equiv \Delta = \frac{T - t_0}{N}, \quad N \in \mathbb{N}, \quad (5.4.10)$$

one expects that the quality of the solution is improved for finer masks, e.g. by going from $\Delta = 2^{-2}$ to $\Delta = 2^{-4}$. However, the true observation is

- the approximation at the end point T is improved;
- the overall approximation on the interval $[t_0, T]$ is not improved.

Definition 5.4.2 Let X_t be the exact solution of a SDE and Y_n its numerical approximation on $[t_0, T]$, then

$$\varepsilon = \mathbb{E}[|X_T - Y(T)|] \quad (5.4.11)$$

is called absolute error. As an estimator, one often uses the quantity

$$\hat{\varepsilon} := \frac{1}{N} \sum_{k=1}^N |X_{T,k} - Y_{T,k}| \quad (5.4.12)$$

for N different simulations (depending on different sample paths of the Wiener process).

A typical result of a numerical experiment reads as follows

Δ	2^{-4}	2^{-5}	2^{-6}	2^{-7}
$\hat{\varepsilon}_1$	0.5093	0.4446	0.3265	0.2292
$\hat{\varepsilon}_2$	0.4692	0.3788	0.2234	0.1477

where different initial values (seeds) for the random number generator have been used. Here, $N = 25$ was fixed in both cases. From these results, it is clear that one wants to obtain an error indicator that is independent e.g. on the seed and on N . This is done by estimating the variance σ_ε^2 of $\hat{\varepsilon}$.

Idea: arrange the simulations into M batches of N simulations each.

Denote by $Y_{T,k,j}$ the value of the k -th generated Euler trajectory in the j -th batch ($k = 1, \dots, N, j = 1, \dots, M$) at time T and by $X_{T,k,j}$ the corresponding exact solution. Then,

$$\hat{\varepsilon}_j := \frac{1}{N} \sum_{k=1}^N |X_{T,k,j} - Y_{T,k,j}| \quad (5.4.13)$$

are independent and Gaussian for $N \rightarrow \infty$. Now, the Student t -distribution can be used to construct confidence intervals. I.e.,

$$\hat{\varepsilon} := \frac{1}{M} \sum_{j=1}^M \hat{\varepsilon}_j = \frac{1}{MN} \sum_{j=1}^M \sum_{k=1}^N |X_{T,k,j} - Y_{T,k,j}| \quad (5.4.14)$$

is an estimate for the mean of the batch averages and

$$\hat{\sigma}_\varepsilon^2 := \frac{1}{M-1} \sum_{j=1}^M (\hat{\varepsilon}_j - \hat{\varepsilon})^2 \quad (5.4.15)$$

is an estimator for σ_ε^2 . The Student t -distribution with $M - 1$ degrees of freedom gives an $100(1 - \alpha)\%$ confidence interval for ε as

$$(\hat{\varepsilon} - \Delta\hat{\varepsilon}, \hat{\varepsilon} + \Delta\hat{\varepsilon}) \quad (5.4.16)$$

with

$$\Delta\hat{\varepsilon} = t_{1-\alpha, M-1} \sqrt{\frac{\hat{\sigma}_\varepsilon^2}{M}}. \quad (5.4.17)$$

Remark 5.4.3 Since $\Delta\hat{\varepsilon} \sim M^{-1/2}$ the number M of batches may be quite large.

Remark 5.4.4 One can observe numerically that

$$|\varepsilon|, |\Delta\hat{\varepsilon}| \lesssim \Delta^{1/2}.$$

We will come back to this point later.

Definition 5.4.5 The random variable $\hat{\varepsilon}$ is decomposed into

$$\hat{\varepsilon} = \varepsilon_{sys} + \varepsilon_{stat}, \quad (5.4.18)$$

where

$$\varepsilon_{sys} := \mathbb{E}(\hat{\varepsilon}) \quad (5.4.19)$$

is the **systematic error** and ε_{stat} the **statistical error**.

Then, we have

$$\begin{aligned} \varepsilon_{sys} &= \mathbb{E}(\hat{\varepsilon}) \\ &= \mathbb{E} \left[\frac{1}{MN} \sum_{j=1}^M \sum_{k=1}^N |X_{T,k,j} - Y_{T,k,j}| \right] \\ &= \mathbb{E}[|X_T - Y_T|] = \varepsilon, \end{aligned}$$

i.e., the systematic error coincides with the absolute error in (5.4.11). For the statistical error

$$\varepsilon_{stat} = \hat{\varepsilon} - \varepsilon,$$

we have

$$\begin{aligned} \text{Var}(\varepsilon_{stat}) &= \text{Var}(\hat{\varepsilon} - \varepsilon) = \mathbb{E}[(\hat{\varepsilon} - \varepsilon)^2] \\ &= \frac{1}{(MN)^2} \sum_{j=1}^M \sum_{k=1}^N \mathbb{E}[(|X_{T,k,j} - Y_{T,k,j}| - \varepsilon)^2] \\ &= \frac{1}{MN} \mathbb{E}[(|X_T - Y_T| - \varepsilon)^2] \\ &= \frac{1}{MN} \text{Var}(|X_T - Y_T|). \end{aligned}$$

This means that the statistical error depends on the total number MN of simulations and not separately on M or N .

5.5 Approximation of Moments

In practice, one may not be interested on a pathwise solution but only on moments e.g. the expectation or the variance. Again, we consider the SDE

$$dX_t = aX_t dt + bX_t dW_t \quad (5.5.1)$$

and consider

$$\mu := \mathbb{E}[Y_T] - \mathbb{E}[X_T] \quad (5.5.2)$$

as an error quantity. Because of the zero expectation property of the Itô integral, we obtain for

$$m(t) := \mathbb{E}[X_T]$$

the equation

$$\frac{d}{dt}m(t) = a m(t), \quad (5.5.3)$$

which implies

$$m(t) = m(0) \cdot e^{at} \quad (5.5.4)$$

since (5.5.3) is a deterministic ordinary differential equation. Hence,

$$\mathbb{E}[X_T] = \mathbb{E}[X_0] \cdot e^{aT}. \quad (5.5.5)$$

Hence, we estimate the errors as above by

$$\hat{\mu}_j = \frac{1}{N} \sum_{k=1}^N Y_{T,k,j} - \mathbb{E}[X_T], \quad j = 1, \dots, M, \quad (5.5.6)$$

and

$$\hat{\mu} := \frac{1}{M} \sum_{j=1}^M \hat{\mu}_j = \frac{1}{MN} \sum_{j=1}^M \sum_{k=1}^N Y_{T,k,j} - \mathbb{E}[X_T], \quad (5.5.7)$$

as well as

$$\hat{\sigma}_\mu^2 := \frac{1}{M-1} \sum_{j=1}^M (\hat{\mu}_j - \hat{\mu})^2 \quad (5.5.8)$$

so that we obtain the $100(1 - \alpha)\%$ confidence interval

$$(\hat{\mu} - \Delta\hat{\mu}, \hat{\mu} + \Delta\hat{\mu}) \quad (5.5.9)$$

with

$$\Delta\hat{\mu} = t_{1-\alpha, M-1} \sqrt{\frac{\hat{\sigma}_\mu^2}{M}}. \quad (5.5.10)$$

Remark 5.5.1 Numerically, one observes

$$|\hat{\mu}|, |\Delta\hat{\mu}| \lesssim \Delta,$$

compared to $\Delta^{1/2}$ in Remark 5.4.4.

As in Section 5.4, we decompose $\hat{\mu}$ as

$$\hat{\mu} = \mu_{sys} + \mu_{stat} \quad (5.5.11)$$

with the *systematic error*

$$\mu_{sys} = \mathbb{E}[\hat{\mu}] = \mu \quad (5.5.12)$$

and

$$\text{Var}(\mu_{stat}) = \text{Var}(\hat{\mu}) = \frac{1}{MN} \text{Var}(Y_T). \quad (5.5.13)$$

5.6 Strong Convergence and Consistency

Definition 5.6.1 For a given maximum step size $\delta \in (0, \delta_0)$, we call

$$(\tau)_\delta := \{\tau_n : n = 0, 1, 2, \dots\} \quad (5.6.1)$$

a time discretization if $\{\tau_n : n = 0, 1, 2, \dots\}$ are time instants (possibly random) with

$$0 \leq \tau_0 < \tau_1 < \dots < \tau_n < \dots < \infty, \quad (5.6.2)$$

$$\sup_n (\tau_{n+1} - \tau_n) \leq \delta, \quad (5.6.3)$$

$$n_t := \max_n \{\tau_n \leq t\} < \infty \quad w.p.1, t \in \mathbb{R}^+. \quad (5.6.4)$$

Moreover, we assume that τ_{n+1} is \mathcal{A}_{τ_n} -measurable where $\mathcal{A}_t : t \geq 0$ is a preassigned increasing family of σ -algebras (generally associated with the Itô or Wiener process).

Definition 5.6.2 A cadlag (right continuous with left hand limits) process $Y = \{Y(t) : t \geq 0\}$ is called time discrete approximation with maximum step size $\delta \in (0, \delta_0)$ if it is based on $(\tau)_\delta$ as in Definition 5.6.1 so that Y_{τ_n} is \mathcal{A}_{τ_n} -measurable and $Y_{\tau_{n+1}}$ can be defined by $Y_{\tau_0}, \dots, Y_{\tau_n}, \tau_0, \dots, \tau_n, \tau_{n+1}$ and a finite number ℓ of \mathcal{A}_{τ_n} -measurable random variables $Z_{n+1,j}, j = 1, \dots, \ell, n \in \mathbb{N}_0$.

We often use the notation Y^δ in order to show the dependence of the maximum step size δ .

Definition 5.6.3 A time discrete approximation Y^δ converges strongly to X at time T if

$$\lim_{\delta \rightarrow 0^+} \mathbb{E}[|X_T - Y^\delta(T)|] = 0 \quad (5.6.5)$$

and it converges strongly with order $p > 0$ at time T , if

$$\varepsilon(\delta) := \mathbb{E}[|X_T - Y^\delta(T)|] \lesssim \delta^p \quad (5.6.6)$$

for each $\delta \in (0, \delta_0)$ with a constant independent of δ .

Remark 5.6.4 From the experiments in Section 5.4 it seems that the Euler scheme converges strongly with order $p = \frac{1}{2}$.

Definition 5.6.5 A time discrete approximation Y^δ is strongly consistent if there exists a nonnegative function $c = c(\delta), \lim_{\delta \rightarrow 0^+} c(\delta) = 0$, such that

$$\mathbb{E} \left[\left| \mathbb{E} \left[\frac{Y_{n+1}^\delta - Y_n^\delta}{\Delta_n} \middle| \mathcal{A}_{\tau_n} \right] - a(\tau_n, Y_n^\delta) \right|^2 \right] \leq c(\delta), \quad (5.6.7)$$

$$\mathbb{E} \left[\frac{1}{\Delta_n} |Y_{n+1}^\delta - Y_n^\delta - \mathbb{E}[Y_{n+1}^\delta - Y_n^\delta | \mathcal{A}_{\tau_n}] - b(\tau_n, Y_n^\delta) \Delta W_n|^2 \right] \leq c(\delta) \quad (5.6.8)$$

for all fixed $Y_n^\delta = y$ and $n \in \mathbb{N}_0$.

Remark 5.6.6 (a) As in the case of numerical methods for ode's, the local property consistency is often easier to verify than the global convergence.

(b) Condition (5.6.7) implies that the increment of Y^δ converges to that of X_t . Without noise this is equivalent to the consistency for standard ode's.

(c) Condition (5.6.8) states that the variance of the difference between the random parts of Y^δ and X_t converge to zero.

Theorem 5.6.7 *Under the assumptions of Theorem 5.2.1 (b) (existence of pathwise strong solution), a strongly consistent time discrete approximation Y^δ of an Itô process X with $Y^\delta(0) = X_0$ converges strongly to X .*

Proof: (in the 1D case) with Gronwall, [7], 324–326. □

Proposition 5.6.8 *Under the assumptions of Theorem 5.6.7, we have that*

$$\sup_{0 \leq s \leq t} \mathbb{E}[|Y_{n_s}^\delta - X_s|^2] \lesssim \delta + c(\delta).$$

As a consequence, if $c(\delta) \equiv 0$, the strong convergence order p_{conv} is related to the strong consistency order p_{cons} as

$$p_{conv} = \sqrt{p_{cons}}.$$

Proposition 5.6.9 *The Euler scheme is strongly consistent with $c(\delta) \equiv 0$ and is thus strongly convergent with order at least $p = \frac{1}{2}$.*

Proof: Exercise. □

5.7 Weak Convergence and Consistency

Definition 5.7.1 *A time discrete approximation Y^δ converges weakly with order $\beta > 0$ to X at time T if for each function*

$$g \in C_P^{2(\beta+1)}(\mathbb{R}^d) := \{h \in C^{2(\beta+1)}(\mathbb{R}^d) : h^{(i)} \text{ has polynomial growth, } 1 \leq i \leq 2(\beta+1)\}$$

there exists a positive constant $C \neq C(\delta)$ and $0 < \delta_0 < \infty$ such that

$$|\mathbb{E}[g(X_T)] - \mathbb{E}[g(Y^\delta(T))]| \leq C\delta^\beta \tag{5.7.1}$$

for all $\delta \in (0, \delta_0)$.

We will see later that strong and weak convergence criteria lead to the construction of different time discrete approximations which are only efficient for one of the two criteria. Hence, a good understanding of the particular

requirements is absolutely necessary.

The following property (again) is often easier to check than weak convergence.

Definition 5.7.2 *A time discrete approximation Y^δ is weakly consistent if there exists a function $c = c(\delta) \geq 0$ with*

$$\lim_{\delta \rightarrow 0^+} c(\delta) = 0 \quad (5.7.2)$$

such that

$$\mathbb{E} \left[\left| \mathbb{E} \left[\frac{Y_{n+1}^\delta - Y_n^\delta}{\Delta_n} \middle| \mathcal{A}_{\tau_n} \right] - a(\tau_n, Y_n^\delta) \right|^2 \right] \leq c(\delta) \quad (5.7.3)$$

and

$$\mathbb{E} \left[\left| \mathbb{E} \left[\frac{1}{\Delta_n} (Y_{n+1}^\delta - Y_n^\delta)(Y_{n+1}^\delta - Y_n^\delta)^T \middle| \mathcal{A}_{\tau_n} \right] - b(\tau_n, Y_n^\delta)b(\tau_n, Y_n^\delta)^T \right|^2 \right] \leq c(\delta) \quad (5.7.4)$$

for all fixed values Y_n^δ and $n \in \mathbb{N}_0$.

Remark 5.7.3 *Obviously, (5.7.3) coincides with the condition (5.6.7) in the definition of strong consistency. However, (5.7.4) differs from (5.6.8) and is much weaker because it only involves the variance of the increments, whereas (5.6.8) requires that the variance of the difference in the increments must vanish.*

Lemma 5.7.4 *The Euler approximation is weakly consistent.*

Proof: Exercise. □

Theorem 5.7.5 *Suppose that the drift coefficient $a(x)$ and the diffusion coefficient $b(x)$ satisfy*

$$a, b \in C_P^4(\mathbb{R}^d), \quad |a'(x)|, |b'(x)| \lesssim 1. \quad (5.7.5)$$

Let Y^δ be a weakly consistent time discrete approximation with equidistant time steps $\Delta_n \equiv \delta$ and $Y^\delta(0) = X_0$ with moment bounds

$$\mathbb{E}[\max_n |Y_n^\delta|^{2q}] \leq K(1 + \mathbb{E}[|X_0|^{2q}]), q \in \mathbb{N}, \quad (5.7.6)$$

$$\mathbb{E} \left[\frac{1}{\Delta_n} |Y_{n+1}^\delta - Y_n^\delta|^6 \right] \leq c(\delta), n \in \mathbb{N}_0, \quad (5.7.7)$$

where $c(\delta)$ satisfies (5.7.2).

Then Y^δ converges weakly to the given Itô process.

Proof: E.g. [7], 329–331. □

Remark 5.7.6 Under appropriate regularity and decay assumptions on a and b (i.e., a little bit more than C^2), one can show that the Euler scheme is weakly convergent of order $\beta = 1$, see [7], 460–464.

5.8 Stability

Consistency and convergence of a scheme are no guarantee for the efficiency of the method. Recall the serious upper bounds for the stepsize for stiff problems (Numerik II). A similar phenomenon arises also for SDEs.

However, the definition of stiff SDE is technically cumbersome. Roughly speaking, a stiff SDE involves two or more widely differing time scales in the solutions. This is similar to the deterministic case.

Definition 5.8.1 Let Y^δ be a time discrete approximation with $Y^\delta(t_0) = Y_0^\delta$ and \bar{Y}^δ a second with $\bar{Y}^\delta(t_0) = \bar{Y}_0^\delta$. We call Y^δ stochastically numerically stable for a given SDE if for any $[t_0, T]$ there exists a constant $\Delta_0 > 0$ such that

$$\lim_{|Y_0 - \bar{Y}_0| \rightarrow 0} \sup_{t_0 \leq t \leq T} P(|Y_{n_t}^\delta - \bar{Y}_{n_t}^\delta| \geq \varepsilon) = 0 \quad (5.8.1)$$

for each $\varepsilon > 0$ and each $\delta \in (0, \Delta_0)$.

We say that Y^δ is stochastically numerically stable for the class of SDE for which Y^δ converges.

Lemma 5.8.2 Under the assumptions of Theorem 5.2.1 (b) (existence of unique pathwise strong solution) the Euler scheme is numerically stable.

Proof: Exercise. □

5.9 Higher order methods

The Euler scheme usually gives good numerical results when the drift and diffusion coefficients are nearly constant. In general, it is however not satisfactory and higher order schemes are required.

The first idea is to use the stochastic Taylor expansion in Section 5.3 to obtain a higher order scheme. Recall the Euler scheme from (5.4.2)

$$Y_{n+1} = Y_n + a\Delta_n + b\Delta W_n. \quad (5.9.1)$$

Recalling the stochastic Taylor expansion and the Itô formula, we add the term

$$\frac{1}{2}bb'((\Delta W_n)^2 - \Delta_n)$$

so that we obtain the *Milstein scheme*

$$Y_{n+1} = Y_n + (a - \frac{1}{2}bb')\Delta_n + b\Delta W_n + \frac{1}{2}bb'(\Delta W_n)^2. \quad (5.9.2)$$

In fact, recall the Itô formula for

$$dX_t = a(X_t)dt + b(X_t)dW_t$$

that reads

$$\begin{aligned} f(X_t) = f(X_{t_0}) &+ \int_{t_0}^t (f'(X_s)a(X_s) + \frac{1}{2}f''(X_s)b(X_s)^2)ds \\ &+ \int_{t_0}^t f'(X_s)b(X_s)dW_s. \end{aligned} \quad (5.9.3)$$

This can be seen by $Y_t = f(X_t)$, i.e.,

$$\frac{\partial}{\partial t}U \equiv 0, \quad \frac{\partial}{\partial x}U \equiv f', \quad \frac{\partial^2}{\partial x^2}U \equiv f''$$

as well as

$$e \equiv a, \quad f \equiv b$$

(on the left the notation of Theorem 5.3.1), so that

$$e_u = a, \quad f_u = b, \quad f_u^2 = b^2,$$

so that (5.9.3) coincides with (5.3.6). We apply (5.9.3) for $f(X_t) \equiv a(X_t)$ and obtain

$$\begin{aligned}
a(X_s) = a(X_{t_0}) &+ \int_{t_0}^s (a'(X_z)a(X_z) + \frac{1}{2}a''(X_z)b(X_z)^2)dz \\
&+ \int_{t_0}^s a'(X_z)b(X_z)dW_z
\end{aligned} \tag{5.9.4}$$

and similarly for $f(X_t) = b(X_t)$. Next, apply (5.9.3) for the identity, i.e., $f(X_t) = X_t$:

$$X_t = X_{t_0} + \int_{t_0}^t a(X_s)ds + \int_{t_0}^t b(X_s)dW_s.$$

Now insert (5.9.4) and the corresponding formula for b'

$$\begin{aligned}
X_t = X_{t_0} &+ \int_{t_0}^t \left\{ a(X_{t_0}) + \int_{t_0}^s (a'a + \frac{1}{2}a''b^2)dz + \int_{t_0}^s a'b dW_z \right\} ds \\
&+ \int_{t_0}^t \left\{ b(X_{t_0}) + \int_{t_0}^s (b'a + \frac{1}{2}b''b^2)dz + \int_{t_0}^s b'b dW_z \right\} dW_s.
\end{aligned}$$

If we neglect all double integrals, we end up with the Euler–Maruyama method:

$$X_t = X_{t_0} + a(X_{t_0})(t - t_0) + b(X_{t_0})(W_t - W_{t_0}) + R,$$

where $R = \mathcal{O}(h)$, $h = t - t_0$. One rewrites R in the following form

$$R = \int_{t_0}^t \int_{t_0}^s b'b dW_z dW_s + \tilde{R}.$$

One can show that $\tilde{R} = \mathcal{O}(h^{3/2})$ (exercise). The next step is the following approximation:

$$\int_{t_0}^t \int_{t_0}^s b'b dW_z dW_s \approx b'(X_{t_0})b(X_{t_0}) \int_{t_0}^t \int_{t_0}^s dW_z dW_s$$

and the latter double integral can be computed as (exercise)

$$\int_{t_0}^t \int_{t_0}^s dW_z dW_s = \int_{t_0}^t (W_s - W_{t_0}) dW_s$$

and then

$$\begin{aligned} \int_{t_0}^t \int_{t_0}^s dW_z dW_s &= \int_{t_0}^t W_s dW_s - W_{t_0} \int_{t_0}^t dW_s \\ &= \frac{1}{2}(W_t^2 - W_{t_0}^2) - \frac{t - t_0}{2} - W_{t_0}(W_t - W_{t_0}) \\ &= \frac{1}{2}((W_t - W_{t_0})^2 - (t - t_0)). \end{aligned}$$

This leads to the additional term

$$bb' \frac{1}{2}((\Delta W_n)^2 - \Delta_n)$$

as in (5.9.2).

Proposition 5.9.1 *The Milstein scheme is strongly consistent for bounded bb' .*

Proof: Exercise. □

Thus, Theorem 5.6.7 implies that the Milstein scheme is strongly convergent of order $p = 1$ as opposed to the Euler scheme for which we have by Proposition 5.6.9 just $p = \frac{1}{2}$.

Remark 5.9.2 *One can follow similar ideas (namely to detail more terms of the remainder \tilde{R}) in order to obtain higher order Taylor schemes. Examples of an order 1.5 and an order 2.0 Taylor scheme can be found in [7], 351-359. Also several general Taylor schemes can be found there.*

The drawback of such Taylor schemes is obvious, namely their construction involves derivatives of various orders of drift and diffusion coefficients and is thus somewhat complicated.

The situation is basically the same for deterministic ode's. There, *Runge–Kutta methods* were designed to avoid complicated Taylor schemes. There are also *stochastic Runge–Kutta methods* that avoid the use of derivatives in the same way. However, these schemes are **not** generalizations of deterministic Runge–Kutta schemes.

We consider the Milstein scheme (5.9.2) and want to replace the only appearing derivative, namely $b'(X_t)$:

$$\begin{aligned} b(X_t + \Delta X_t) - b(X_t) &= b'(X_t)\Delta X_t + \mathcal{O}(|\Delta X_t|^2) \\ &= b'(X_t)(a(X_t)\Delta_n + b(X_t)\Delta W_t) + \mathcal{O}(\Delta_n) \\ &= b'(X_t)b(X_t)\Delta W_t + \mathcal{O}(\Delta_n). \end{aligned}$$

Since $\mathbb{E}[(\Delta W_t)^2] = \Delta^2$, we replace ΔW_t by $\sqrt{|\Delta_n|}$ and have

$$b'(X_t)b(X_t) = \frac{1}{\sqrt{\Delta_n}} \left[b\left(X_t + a(X_t)\Delta_n + b(X_t)\sqrt{|\Delta_n|}\right) - b(X_t) \right] + \mathcal{O}(\sqrt{\Delta_n})$$

so that we obtain a first order stochastic Runge–Kutta scheme

$$\begin{aligned} \hat{Y}_n &= Y_n + a\Delta_n + b\sqrt{\Delta_n} \\ Y_{n+1} &= Y_n + a\Delta_n + b\Delta W_n + \frac{1}{2\sqrt{\Delta_n}}(b(\hat{Y}_n, t_n) - b(Y_n, t_n))((\Delta W_n)^2 - \Delta_n) \end{aligned} \quad (5.9.5)$$

One can easily show that this scheme is also of first order like Milstein.

Remark 5.9.3 *One could try to define a general class of stochastic Runge–Kutta methods by*

$$Y_{n+1} = Y_n + \Delta_n \sum_{j=1}^s d_j a(\hat{Y}_j) + \Delta W_n \sum_{j=1}^s e_j b(\hat{Y}_j) \quad (5.9.6)$$

with the increments

$$\hat{Y}_j = Y_n + \Delta_n \sum_{k=1}^s D_{jk} a(Y_k) + \Delta W_n \sum_{k=1}^s E_{jk} b(Y_k), j = 1, \dots, s. \quad (5.9.7)$$

However, it is known that such methods have at most a (strong) convergence order 1, i.e., not more than (5.9.3). To avoid this, one would need to introduce further random variables to approximate the multiple integrals.

Chapter 6

Elliptic Partial Differential Equations

In this chapter, we give a brief introduction to numerical methods for solving elliptic partial differential equations. This is wide topic and could easily fill a lecture by its own. We can only focus on some aspects here. We will focus mainly on the 2D case here. Before we proceed, let us give an example from mathematical finance leading to such a pde.

Example 6.0.4 (Multi-Factor Models) *A short rate model $r(t)$ gives the short interest rate $r(t)$ in dependence of t . A standard model for this is*

$$dr(t) = a(t, r(t)) dt + b(t, r(t)) dW_t,$$

where the functions a, b are assumed to fulfill the assumptions for existence and uniqueness. In particular, this is the only source of risk in this model.

One however observes that not all known short rates can be modelled with one single term W_t , thus one considers the model

$$r(t) = R(X_t, t),$$

where X_t is some Itô-process in \mathbb{R}^d . For example in the well-known Vasicek model one obtains then the following equation for the price function F of a bond in dependence of $r(t)$

$$\begin{cases} F_t(x, t) + \sigma(x, t)\Delta F(x, t) + \boldsymbol{\mu}(x, t)\nabla F(x, t) - R(x, t)F(x, t) = h(x, t) \\ \forall (x, t) \in \mathbb{R}^d \times [0, T] \end{cases} \quad (6.0.1)$$

where $h : \mathbb{R}^d \times [0, T]$ is a given function.

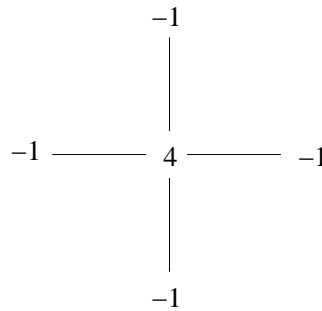


Figure 6.1: Five point stencil

6.1 Finite Difference Methods

In analogy to the 1D-case, we can approximate F_t by finite differences. Again, we do the same in space and obtain

$$(\Delta_2 u)(x, y) := \frac{1}{h^2} [u(x-h, y) + u(x+h, y) + u(x, y-h) + u(x, y+h) - 4u(x, y)]$$

i.e., the well-known *five point stencil* which is visualized in Figure 6.1. In the 1D case, we have seen that the resulting system matrix is tridiagonal and we could use a special version of a direct solver in order to obtain an efficient numerical method. In 2D, we obtain a block-tridiagonal matrix for which there exists no such nice recursion formula. Here, one has to use iterative methods such as Gauß-Seidel, Jacobi, cg or pcg-methods.

As we have already seen in the discussion of 1D problems, finite difference methods are quite simple to derive (and also to implement). But, as we also have seen in 1D, their use poses quite strong regularity assumptions on the solution which might not be satisfied in realistic applications. Moreover, since a finite difference method basically corresponds to a rectangular grid, the treatment of non-trivial geometric domains is a non-trivial task. Even more substantial is the following observation.

Example 6.1.1 *We consider the Poisson problem on the unit square*

$$\begin{aligned} -\Delta u &= 0 & \text{in } \Omega &= (0, 1)^2 \\ u &= x^2 & \text{on } \Gamma &= \partial\Omega \end{aligned}$$

This problem has a unique solution, but

$$u_{xx}(0, 0) = 2 \neq 0 = u_{yy}(0, 0)$$

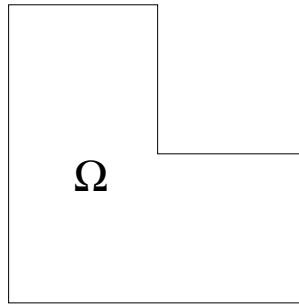


Figure 6.2: L-shaped domain.

due to the boundary conditions which contradicts the differential equation. Thus the solution u cannot be twice continuous differentiable: $u \notin C^2(\bar{\Omega})$.

Example 6.1.2 Again we consider the Poisson problem, but now on the L-shaped domain

$$\Omega := \left(-\frac{1}{2}, \frac{1}{2}\right) \times \left(-\frac{1}{2}, \frac{1}{2}\right) \setminus \left[0, \frac{1}{2}\right] \times \left[0, \frac{1}{2}\right]$$

which is shown in Figure 6.2. Also this problem has a unique solution which can be written in polar coordinates as

$$\begin{aligned} -\Delta u &= 0 && \text{in } \Omega \\ u &= r^{\frac{2}{3}} \sin\left(\frac{2\varphi - \pi}{3}\right) && \text{on } \Gamma \end{aligned}$$

It can easily be seen that the first derivatives of u are not bounded, i.e., we have $u \notin C^1(\bar{\Omega})$.

The above remarks and the two examples clearly show that in many situations the classical (strong) formulation of PDEs is not adequate. Obviously it is not always meaningful to pose a pointwise condition in the partial differential equation. In the sequel, we will hence introduce certain weak formulations of elliptic PDEs. Before doing so, we should explain what is meant by *elliptic*.

This chapter is mainly based upon [1, 2].

6.2 Categories of second order PDEs

In the sequel, we consider the general linear second order differential equation in n variables

$$-\sum_{i,k=1}^n a_{i,k}(x) \frac{\partial}{\partial x_i} \frac{\partial}{\partial x_k} u(x) + \sum_{i=1}^n b_i(x) \frac{\partial}{\partial x_i} u(x) + c(x)u(x) = f(x). \quad (6.2.1)$$

In the case $a_{i,k}(x) \equiv a_{i,k}$, $b_i(x) \equiv b_i$, $c(x) \equiv c$, (6.2.1) is called a PDE with *constant coefficients*. Since $u_{x_i x_k} = u_{x_k x_i}$ if $u \in C^2$, we can assume without loss of generality that

$$A(x) := (a_{i,k}(x))_{i,k=1,\dots,n} \in \mathbb{R}^{n \times n}$$

is symmetric.

Definition 6.2.1 *The equation (6.2.1) is called*

- (i) *elliptic in x , if $A(x)$ is positive definite;*
- (ii) *hyperbolic in x , if $A(x)$ has one negative and $n-1$ positive eigenvalues;*
- (iii) *parabolic in x , if $A(x)$ is positive semidefinite but not definite and the rank of $(A(x), b(x))$ is n .*

If (6.2.1) is elliptic, one often abbreviates (6.2.1) as $Lu = f$.

Example 6.2.2 *Let $Lu = f$ be elliptic.*

- (i) *The equation $u_{tt} + Lu = f$ is hyperbolic.*
- (ii) *The equation $u_t + Lu = f$ is parabolic.*
- (iii) *Let $A(x) \equiv \text{Id}$, then $Lu = -\Delta u + b \cdot \nabla u + cu = f$ is elliptic.*
- (iv) *The wave equation $u_{tt} = u_{xx}$ is hyperbolic.*
- (v) *The heat equation $u_t = u_{xx}$ is parabolic.*

Remark 6.2.3 (i) *The PDEs from finance that we have considered so far are parabolic. If one uses a discretization in time by means of finite differences, the numerical solution of such problems is reduced to the solution of elliptic PDEs.*

- (ii) *The treatment of hyperbolic PDEs needs different techniques due to the presence of shocks, rarefaction waves etc.*

6.3 Variational Formulation of Elliptic PDEs

Introducing suitable discretizations in time, we are left with the treatment of elliptic PDEs. As we have seen before, the classical (i.e., pointwise) formulation of PDEs might not be the appropriate one. Note that the above mentioned examples are in fact elliptic. Hence, we introduce the weak (or variational) formulation of elliptic PDEs in this section.

For $u, v \in L_2(\Omega)$, $\Omega \subset \mathbb{R}^d$ open with piecewise smooth boundary, denote by

$$(u, v)_0 := \int_{\Omega} u(x) v(x) dx$$

the standard inner product in $L_2(\Omega)$ and denote by $\|u\|_0 := \sqrt{(u, u)_0}$ the induced norm.

We start by the definition of weak derivatives.

Definition 6.3.1 *A function $u \in L_2(\Omega)$ has the (weak) derivative $v = \partial^\alpha u$ of order $\alpha \in \mathbb{N}^d$, ($\alpha = (\alpha_1, \dots, \alpha_d)$, $|\alpha| := \alpha_1 + \dots + \alpha_d$) in $L_2(\Omega)$ if $v \in L_2(\Omega)$ and*

$$(\phi, v)_0 = (-1)^{|\alpha|} (\partial^\alpha \phi, u)_0 \quad \forall \phi \in C_0^\infty(\Omega). \quad (6.3.1)$$

Next, similar to the classical smoothness spaces $C^m(\Omega)$ we define spaces of weakly differentiable functions as follows.

Definition 6.3.2 *For $m \geq 0$, $m \in \mathbb{N}$ we denote by $H^m(\Omega)$ the space of all functions $u \in L_2(\Omega)$ such that $\partial^\alpha u \in L_2(\Omega)$ for all $\alpha \in \mathbb{N}^d$ such that $|\alpha| \leq m$. This space is called Sobolev space of order m . The norm in $H^m(\Omega)$ is defined by*

$$\|u\|_m := \sqrt{(u, u)_m}, \quad (u, v)_m := \sum_{|\alpha| \leq m} (\partial^\alpha u, \partial^\alpha v)_0. \quad (6.3.2)$$

The following theorem that we quote from [1, 2] without proof shows that any $H^m(\Omega)$ -function can be approximated by smooth functions to any desired accuracy.

Theorem 6.3.3 *The space $H^m(\Omega) \cap C^\infty(\Omega)$ is dense in $H^m(\Omega)$.*

Definition 6.3.4 *The closure of $C_0^\infty(\Omega)$ (the space of arbitrarily smooth functions with compact support in Ω) with respect to the Sobolev norm $\|\cdot\|_m$ is called $H_0^m(\Omega)$.*

Roughly speaking, $H_0^m(\Omega)$ contains those functions in $H^m(\Omega)$ with generalized homogeneous boundary conditions on the boundary $\partial\Omega$. Note that $H_0^m(\Omega)$ is a subspace of $L_2(\Omega)$, thus point evaluations are not well-defined. Thus one cannot talk about boundary conditions in the classical sense, namely pointwise. Generalized boundary conditions are defined by means of the so-called *trace operator*, for details see [1, 2].

Let us now consider the following **model problem**

$$-\Delta u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega.$$

For a test function $\phi \in C_0^\infty(\Omega)$ we obtain by integration by parts

$$\begin{aligned} (f, \phi)_0 &= (-\Delta u, \phi)_0 = \sum_{i=1}^d \left(-\frac{\partial^2}{\partial x_i^2} u, \phi \right)_0 \\ &= \sum_{i=1}^d \left(\frac{\partial}{\partial x_i} u, \frac{\partial}{\partial x_i} \phi \right)_0 \\ &= (\nabla u, \nabla \phi)_0 =: a(u, \phi). \end{aligned}$$

Then, $u \in H_0^1(\Omega)$ is called a *weak solution* of the model problem, if

$$a(u, v) = (f, v)_0$$

holds for all $v \in H_0^1(\Omega)$.

Remark 6.3.5 (i) The bilinear form $a(\cdot, \cdot)$ is symmetric, i.e., $a(u, v) = a(v, u)$, positive, i.e., $a(u, v) \geq 0$ and $a(u, u) > 0$ if $u \neq 0$, bounded, i.e., $|a(u, v)| \leq C \|u\|_1 \|v\|_1$ for all $u, v \in H_0^1(\Omega)$ and coercive in $H_0^1(\Omega)$, i.e.,

$$a(u, u) \geq \alpha \|u\|_1^2.$$

The latter equation is a consequence of the Poincaré-Friedrichs inequality.

(ii) The existence of a weak solution follows from the following characterization theorem:

The linear functional $J(v) := \frac{1}{2}a(v, v) - (f, v)_0$ has its minimum in u if and only if

$$a(u, v) = (f, v)_0 \quad \forall v \in H_0^1(\Omega).$$

(iii) A bilinear form $a : H \times H \rightarrow \mathbb{R}$ on a Hilbert space H is called continuous if it is bounded. A symmetric and continuous bilinear form is called elliptic if it is coercive.

- (iv) **Lax-Milgram Theorem:** Let $V \subset H$ be a closed and convex subset and let $a(\cdot, \cdot)$ be an elliptic bilinear form. Then, the variational problem

$$J(v) := \frac{1}{2}a(v, v) - \langle \ell, v \rangle \rightarrow \min!$$

has a unique solution in V for any $\ell \in H'$.

- (v) For the special case $H = L_2(\Omega)$ and $V = H_0^1(\Omega)$ we obtain the existence of a weak solution for the general elliptic PDE.
- (vi) If $u \in C^2(\Omega) \cap C^0(\bar{\Omega})$ is a classical solution of $Lu = f$, $u|_{\partial\Omega} = 0^1$, then $u \in H_0^1(\Omega)$ is also a weak solution. On the other hand, if $u \in H_0^1(\Omega)$ is a weak solution and in addition $u \in C^2(\Omega) \cap C^0(\bar{\Omega})$, then it is also a classical solution.
- (vii) **Reduction to homogeneous boundary conditions (homogenization):** Consider the boundary value problem (bvp) $Lu = f$, $u|_{\partial\Omega} = g$. Choose some $u_0 \in H^1(\Omega)$ such that $u_0|_{\partial\Omega} = g$ and consider the homogeneous problem

$$Lw = f_1 := f - Lu_0, \quad w|_{\partial\Omega} = 0.$$

Then $u := w + u_0$ satisfies $u|_{\partial\Omega} = u_0|_{\partial\Omega} = g$ as well as

$$Lu = Lw + Lu_0 = f - Lu_0 + Lu_0 = f,$$

i.e., u is a solution of the original bvp.

6.4 Ritz-Galerkin methods

The introduced variational formulation is a problem posed in a Hilbert space, in our particular case in a function space which is usually of *infinite dimension*. Thus we cannot treat this problem directly on a computer. The idea, which goes back to Ritz (1908), is to replace the infinite-dimensional minimization problem in the Hilbert space by a finite one using finite-dimensional subspaces $S_h \subset V$ as trial- and test spaces.

¹When we write $u|_{\partial\Omega}$ we always mean the boundary values in the sense of the trace operator, [1, 2].

Let $S_h \subset V$ be a finite-dimensional subspace. We consider the problem of determining $u_h \in S_h$ such that

$$a(u_h, v_h) = (f, v_h)_0 \quad \forall v_h \in S_h. \quad (6.4.1)$$

By the above remarks, this problem has a unique solution due to the properties of the bilinear form $a(\cdot, \cdot)$. If $\{\psi_1, \dots, \psi_N\}$ is a basis for S_h , $N = \dim S_h$, (6.4.1) is equivalent to the following problem: Determine $z_h = (z_1, \dots, z_N) \in \mathbb{R}^N$ such that

$$A_h z_h = b_h \quad (6.4.2)$$

where $A_h = \left(a(\psi_k, \psi_i) \right)_{i,k=1,\dots,N} \in \mathbb{R}^{N \times N}$ is called the *stiffness matrix* and $b_h = \left((f, \psi_k)_0 \right)_{k=1,\dots,N} \in \mathbb{R}^N$ is the right-hand side.

Lemma 6.4.1 *The stiffness matrix A_h is symmetric positive definite.*

Proof: Using bilinearity and coercivity yields

$$\begin{aligned} d_h^T A d_h &= \sum_{i,k=1}^N d_i A_{i,k} d_k \\ &= a \left(\sum_{k=1}^N d_k \psi_k, \sum_{i=1}^N d_i \psi_i \right) \\ &= a(v_h, v_h) \geq \alpha \|v_h\|^2. \quad \square \end{aligned}$$

Remark 6.4.2 *If the basis $\{\psi_1, \dots, \psi_N\}$ is local in the sense that*

$$\#\{k = 1, \dots, N : |\text{supp } \psi_k \cap \text{supp } \psi_i| > 0\} \leq c \ll N$$

independent of i , then A_h is also sparse which means that the number of non-zero elements per row and column is $\mathcal{O}(1)$.

We now study the convergence of Ritz-Galerkin methods. The following central statement shows that the Galerkin solution is as good as the best approximation to the (unknown) solution out of the trial space S_h . In that sense, the method is optimal.

Theorem 6.4.3 (Ceá's lemma) *Let the bilinear form $a(\cdot, \cdot)$ be elliptic on V , where $H_0^1(\Omega) \subseteq V \subseteq H^1(\Omega)$ and denote the solutions of the variational*

problem in V and $S_h \subset V$ by u and u_h , respectively. Then, there exists $C > 0$ such that we have the following inequality

$$\|u - u_h\|_1 \leq \frac{C}{\alpha} \inf_{v_h \in S_h} \|u - v_h\|_1 .$$

Proof: We have

$$\begin{aligned} a(u, v) &= (f, v)_0 \quad \forall v \in V \\ a(u_h, v_h) &= (f, v_h)_0 \quad \forall v_h \in S_h , \end{aligned}$$

which implies the so-called *Galerkin orthogonality*

$$a(u - u_h, v_h) = 0 \quad \forall v_h \in S_h .$$

Let $v_h \in S_h$, so that with $w_h := v_h - u_h \in S_h$ we obtain

$$a(u - u_h, v_h - u_h) = 0$$

and by coercivity and boundedness

$$\begin{aligned} \alpha \|u - u_h\|_1^2 &\leq a(u - u_h, u - u_h) \\ &= a(u - u_h, u - v_h) + \underbrace{a(u - u_h, v_h - u_h)}_{=0} \\ &\leq C \|u - u_h\|_1 \|u - v_h\|_1 \end{aligned}$$

which proves the theorem. \square

Sometimes this method is also simply called *Galerkin method* and the solution $u_h \in S_h$ is called the *Galerkin solution* or *Galerkin approximation*. The finite dimensional space S_h is also called *trial space*.

6.5 Some Simple Finite Elements

Again, we consider the homogeneous model problem

$$-\Delta u = f \quad \text{in } \Omega, \quad u = 0 \quad \text{on } \partial\Omega$$

where $\Omega = (0, 1)^2$. The idea is to subdivide Ω into regular triangles of meshsize h as indicated in Figure 6.3. Using this *triangulation* of Ω , we define the trial space

$$S_h := \{v \in C(\bar{\Omega}) : v \text{ is linear in each triangle and } v|_{\partial\Omega} = 0\}.$$

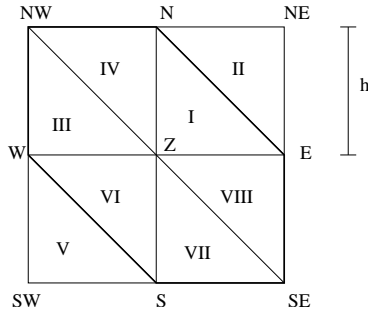


Figure 6.3: Support of the Courant Finite Element.

In each triangle (*element*) a $v_h \in S_h$ takes the form

$$v_h(x, y) = a + bx + cy, \quad a, b, c \in \mathbb{R},$$

i.e., v_h is uniquely determined by the value at three different nodes. From this we can easily see that

$$N = \dim S_h = \text{number of interior grid points}$$

and $v_h \in S_h$ is globally (i.e., on $\bar{\Omega}$) determined by its values on these N grid points. The canonical (so-called *nodal*) basis $\{\psi_i\}_{i=1, \dots, N}$ is defined by

$$\psi_i(x_j, y_j) = \delta_{ij}.$$

In Figure 6.3, the support of ψ_Z is shown. Let h denote the mesh size of the triangles, then we obtain the following values for the derivatives of ψ_Z (see Figure 6.3).

triangle	I	II	III	IV	V	VI	VII	VIII
$\partial_x \psi_Z$	$-h^{-1}$	0	h^{-1}	0	0	h^{-1}	0	$-h^{-1}$
$\partial_y \psi_Z$	$-h^{-1}$	0	0	$-h^{-1}$	0	h^{-1}	h^{-1}	0

Then, straightforward calculations show that by symmetry we have

$$\begin{aligned}
a(\psi_Z, \psi_Z) &= \int_{\text{I-VIII}} (\nabla \psi_Z)^2 dx dy \\
&= 2 \int_{\text{I} \cup \text{III} \cup \text{IV}} [(\partial_x \psi_Z)^2 + (\partial_y \psi_Z)^2] dx dy \\
&= 2h^{-2} \left\{ \underbrace{\int_{\text{I} \cup \text{III}} dx dy}_{=h^2} + \underbrace{\int_{\text{IV}} dx dy}_{=h^2} \right\} \\
&= 4
\end{aligned}$$

for the diagonal entries of the stiffness matrix and

$$\begin{aligned}
a(\psi_Z, \psi_N) &= \int_{\text{I} \cup \text{IV}} \nabla \psi_Z \cdot \nabla \psi_N dx dy \\
&= \int_{\text{I} \cup \text{IV}} (\underbrace{\partial_x \psi_Z \partial_x \psi_N}_{=0} + \partial_y \psi_Z \partial_y \psi_N) dx dy \\
&= -h^{-2} \int_{\text{I} \cup \text{IV}} dx dy \\
&\quad \underbrace{\hspace{10em}}_{=h^2} \\
&= -1 = a(\psi_Z, \psi_O) \\
&= a(\psi_Z, \psi_S) = a(\psi_Z, \psi_W).
\end{aligned}$$

Finally, we obtain

$$\begin{aligned}
a(\psi_Z, \psi_{NW}) &= \int_{\text{III} \cup \text{IV}} (\partial_x \psi_Z \partial_x \psi_{NW} + \partial_y \psi_Z \partial_y \psi_{NW}) dx dy \\
&= \int_{\text{III}} h^{-1} \cdot 0 + 0 \cdot h^{-1} + \int_{\text{IV}} 0 \cdot (h^{-1}) + (-h^{-1})0 = 0
\end{aligned}$$

and again by symmetry

$$a(\psi_Z, \psi_{SO}) = a(\psi_Z, \psi_{SW}) = a(\psi_Z, \psi_{NO}).$$

This means that in this particular case of a uniform triangulation and the nodal basis for piecewise linear finite elements the stiffness matrix coincides with the matrix arising from the 5-point-stencil in finite difference methods. This principle, however, is not true in general, i.e., for a finite element discretization there is in general *not* an equivalent finite difference discretization. Finite elements are much more flexible than finite differences and they allow the treatment of the weak formulation of the bvp.

Some properties of finite elements

- 1.) Subdivision (or *partition*) of Ω in triangular or quadrilateral elements. If all elements are congruent, this is called a *regular* subdivision.
- 2.) In 2D, we denote by

$$\mathcal{P}_t := \left\{ u(x, y) = \sum_{i+k \leq t; i, k \geq 0} c_{i,k} x^i y^k \right\}$$

the set of all algebraic polynomials of degree at most t . The restriction of the trial (or *shape*) functions to an element is a polynomial.

- 3.) Smoothness: A finite element is said to be *of order* k if it is in $C^k(\Omega)$.

For the example of the *Courant Finite Element* which is shown in Figure 6.3, we have $t = 1$ and $k = 0$.

Definition 6.5.1 (i) A partition $\mathcal{T} = \{T_1, \dots, T_M\}$ of Ω in triangular or quadrilateral elements is called *admissible*, if

- a) $\bar{\Omega} = \bigcup_{i=1}^M T_i$
- b) If $T_i \cap T_j$ consists of exactly one point, this point is a corner of both T_i and T_j .
- c) If $T_i \cap T_j$, $i \neq j$ consists of more than one point, then $T_i \cap T_j$ is a common edge of T_i and T_j .

(ii) We write \mathcal{T}_h if each element has a diameter of at most $2h$.

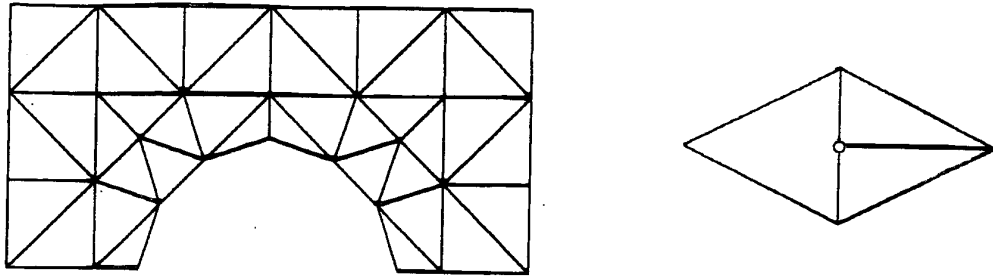


Figure 6.4: An admissible triangulation (left) and a non-admissible triangulation (right) with a hanging node. (Taken from [1].)

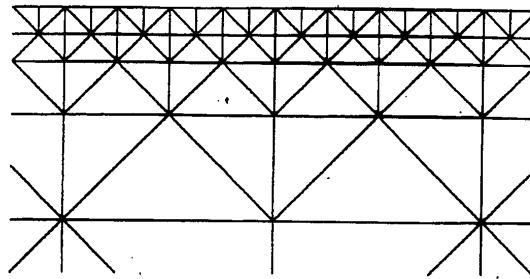


Figure 6.5: A quasi-uniform but non-uniform triangulation. (Taken from [1].)

(iii) \mathcal{T}_h is called quasi-uniform if there exists some $\kappa > 0$ such that each $T \in \mathcal{T}_h$ contains a circle with radius

$$\rho_T \geq \frac{h_T}{\kappa}.$$

Examples of triangulations are shown in Figures 6.4-6.6.

The following theorem shows how to choose the order of the elements in order to be contained in a certain Sobolev space. If the finite elements are contained in the Sobolev space corresponding to the variational formulation, the elements are called *conforming*, otherwise *non-conforming*. For the elliptic second order problem this would mean that the elements are conforming if $S_h \subset H_0^1(\Omega)$.

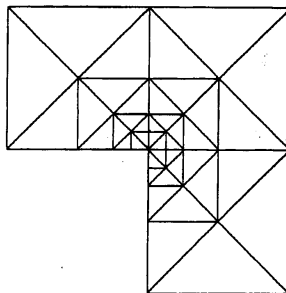


Figure 6.6: A non-uniform triangulation at a reentrant corner. (Taken from [1].)

Theorem 6.5.2 *Let $k \geq 1$ and Ω be bounded. A piecewise C^∞ -function $v : \bar{\Omega} \rightarrow \mathbb{R}$ is in $H^k(\Omega)$ if and only if $v \in C^{k-1}(\bar{\Omega})$. \square*

The latter theorem in particular implies that for the elliptic second order problem we would have $k = 1$ thus $v \in C^0(\bar{\Omega})$ which in particular shows that the Courant Finite Element is conforming.

Definition 6.5.3 *For any finite element space there is a set of points in the sense that the shape functions are uniquely defined by the values at these points. Those functions that take the value 1 at exactly one of these points and 0 on all the others are called nodal basis functions or Lagrange elements.*

Table 6.1 shows a number of standard finite elements. We also show some higher order elements in which the point values are not sufficient to define a shape function uniquely. Also certain derivatives are needed in that case.

Remark 6.5.4 *With the aid of affine mappings, one can usually reduce oneself to one single reference element T_{ref} , i.e., for any $T_j \in \mathcal{T}$ there exists an affine mapping $F_j : T_{\text{ref}} \rightarrow T_j$ such that*

$$v_h(x)|_{T_j} = p(F_j^{-1}x)|_{T_j}, \quad p \in \mathcal{P}_{\text{ref}}, \quad v_h \in S_h.$$

6.6 Approximation Results

In this section, we give some results concerning the approximation properties of the finite element method and also derive some error estimates.

- value of the function
- ⊙ value of the function, 1st and 2nd derivative
- ⊥ normal derivative

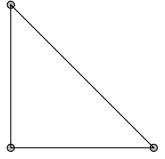
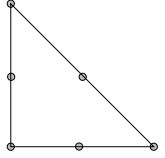
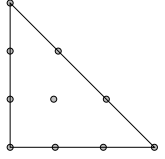
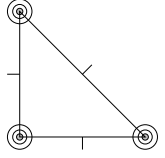
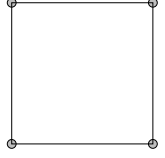
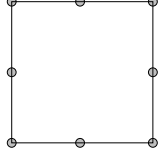
	<p>Linear triangular element</p> <p>$u \in C^0(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_1, \dim \Pi_{\text{ref}} = 3$</p>
	<p>Quadratic triangular element</p> <p>$u \in C^0(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_2, \dim \Pi_{\text{ref}} = 6$</p>
	<p>Cubic triangular element</p> <p>$u \in C^0(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_3, \dim \Pi_{\text{ref}} = 10$</p>
	<p>Agyris element</p> <p>$u \in C^1(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_5, \dim \Pi_{\text{ref}} = 21$</p>
	<p>Linear quadrilateral element</p> <p>$u \in C^0(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_2, u _{\partial T_i} \in \mathcal{P}_1, \dim \Pi_{\text{ref}} = 4$</p>
	<p>Quadrilateral serendipity element</p> <p>$u \in C^1(\Omega)$</p> <p>$\Pi_{\text{ref}} = \mathcal{P}_3, u _{\partial T_i} \in \mathcal{P}_2, \dim \Pi_{\text{ref}} = 8$</p>

Table 6.1: Standard Finite Elements. (Taken from [1].)

Definition 6.6.1 For any partition $\mathcal{T}_h = \{T_1, \dots, T_M\}$ of Ω and $m \geq 1$, we define the grid norm by

$$\|v\|_{m,h} := \left(\sum_{T_j \in \mathcal{T}_h} \|v\|_{m,T_j}^2 \right)^{1/2}.$$

Obviously, we have $\|v\|_{m,h} = \|v\|_{m,\Omega}$ for $v \in H^m(\Omega)$.

The following well-known theorem is a central statement in functional analysis.

Theorem 6.6.2 (Bramble-Hilbert theorem) Let $\Omega \subset \mathbb{R}^2$ be a domain with Lipschitz-continuous boundary, $t \geq 2$ and let $L : H^t(\Omega) \rightarrow Y$ be a linear, bounded operator on a normed space Y . If $\mathcal{P}_{t-1}(\Omega) \subset \text{Ker}(L)$, then we have

$$\|Lv\|_Y \leq c |v|_t$$

for all $v \in H^t(\Omega)$ with a constant $c = c(L, \Omega)$. \square

We now apply this theorem to the interpolation operator

$$I_h : H^t(\Omega) \rightarrow S_h,$$

which is defined by interpolation of the input function with respect to the nodal grid points of the underlying triangulation. Then, we immediately obtain the following error estimate.

Theorem 6.6.3 Let $t \geq 2$ and \mathcal{T}_h be a quasi-uniform triangulation of Ω . Then, we have for the interpolation operator I_h defined by interpolation with piecewise polynomials of degree $t - 1$ that

$$\|u - I_h u\|_{m,h} \leq ch^{t-m} |u|_{t,\Omega}$$

for $u \in H^t(\Omega)$, $0 \leq m \leq t$ and some constant $c = c(\Omega, \mathcal{T}_h, t)$. \square

The principle behind the latter theorem can be roughly described as ‘polynomial exactness implies approximation power’. This is also known as *Bramble-Hilbert type argument*.

Finally, we give an *a priori* error estimate which also shows the continuous dependence of the error on the right-hand side data.

Theorem 6.6.4 *Let \mathcal{T}_h be a family of quasi-uniform triangulations of a convex domain Ω . Then, the piecewise linear finite element approximation $u_h \in S_h$ satisfies the following estimate*

$$\|u - u_h\|_1 \leq ch\|u\|_2 \leq ch\|f\|_0 . \quad \square$$

Remark 6.6.5 (i) *The above regularity assumption $u \in H^2(\Omega)$ can also be weakened.*

(ii) *The computation of an appropriate triangulation is a problem on its own, in particular for complex geometries Ω .*

(iii) *The setup of the linear system (stiffness matrix and right-hand side) is usually done on the reference element.*

6.7 Example: 1D Finite Element Discretization for the Black-Scholes Equation

A variational formulation in 1D reads $a(u, v) = (f, v)_0$ for all $v \in H_0^1(\Omega)$, where $\Omega = (0, 1)$ and (e.g.)

$$a(u, v) = (u', v')_{0,(0,1)} + (u', v)_{0,(0,1)} + (u, v)_{0,(0,1)}.$$

We now subdivide $(0, 1)$ uniformly by equidistant grid points $x_h^k := kh$, where $h = \frac{1}{M}$, $k = 0, \dots, M$, i.e., we have $M - 1$ interior nodes. An ‘element’ in 1D is hence a subinterval (x_h^{k-1}, x_h^k) , $k = 1, \dots, M$. Next, we consider the nodal basis as shown in Figure 6.7. We now apply this for the Black-Scholes equation for european option pricing

$$\frac{\partial}{\partial t} u(t, x) + \frac{1}{2} \sigma^2 x^2 \frac{\partial^2}{\partial x^2} u(t, x) + rx \frac{\partial}{\partial x} u(t, x) - ru(t, x) = 0,$$

where u is the value of the option, x is the asset price, r is the interest rate and σ is the volatility. We consider the call, i.e., the terminal condition

$$u(T, x) = \max(K - x, 0),$$

where again K denotes the strike. We rewrite this in coefficient form

$$\frac{\partial}{\partial t} u(t, x) + \frac{\partial}{\partial x} \left(c(x) \frac{\partial}{\partial x} u(t, x) \right) + b(x) \frac{\partial}{\partial x} u(t, x) - ru(t, x) + ad_a u(t, x) = 0,$$

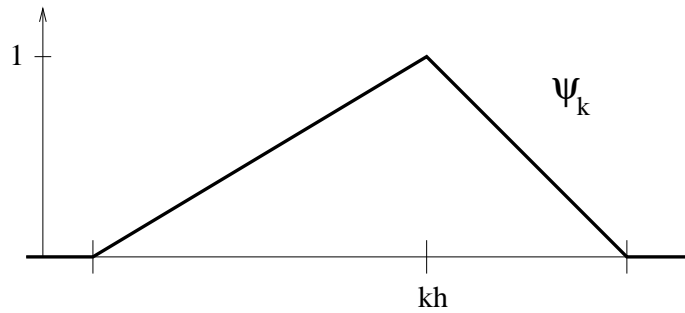


Figure 6.7: 1D nodal piecewise linear shape function.

which is required for FEMLAB

$$\frac{\partial}{\partial t}u(t, x) + \frac{\partial}{\partial x} \left(\frac{1}{2}\sigma^2 x^2 \frac{\partial}{\partial x} u(t, x) \right) + \left(rx - \frac{\partial}{\partial x} \left(\frac{1}{2}\sigma^2 x^2 \right) \right) \frac{\partial}{\partial x} u(t, x) - ru(t, x) = 0,$$

i.e.,

$$\begin{aligned} c &:= \frac{1}{2}\sigma^2 x^2 \\ b &:= (r - \sigma^2)x = rx - \frac{\partial}{\partial x} \left(\frac{1}{2}\sigma^2 x^2 \right) \\ a &:= r, \\ d_a &:= -1. \end{aligned}$$

In this form, we can immediately solve this equation in FEMLAB and we show the computed solution for $K = 40$, $\sigma = 0.3$, $r = 0.12$, $x = 80$ and $T = 12$ in Figure 6.8.

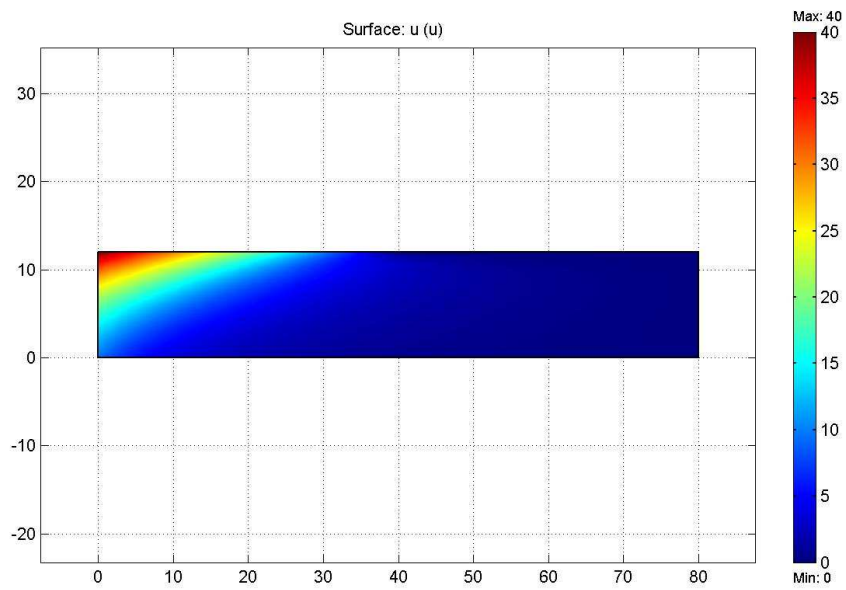


Figure 6.8: Result u of a numerical simulation using FEMLAB. The horizontal axis corresponds to x , the vertical to t .

Chapter 7

American Option Pricing

In contrast to European Options, an American Option can be exercised at **any** time $t \leq T$. This means that the payoff function is (S denotes again the price of the underlying asset, K is the exercise price, i.e. the strike)

$$\begin{aligned} V_C^{\text{am}}(S, t) &= (S_t - K)^+ && \text{for a call and} \\ V_P^{\text{am}}(S, t) &= (K - S_t)^+ && \text{for a put.} \end{aligned} \tag{7.0.1}$$

This means, at expiration time T , the payoff coincides with European Options. Under no-arbitrage assumptions, one can easily show the following a-priori bounds

$$\begin{aligned} V_P^{\text{am}}(S, t) &\geq (K - S)^+ \\ V_C^{\text{am}}(S, t) &\geq (S - K)^+ \end{aligned} \quad \forall S, t$$

and trivially

$$V^{\text{am}} \geq V^{\text{eur}} ,$$

since a European Option is a special case of an American Option.

Again, we do not go into details of the modelling of American options but refer to any lecture on Financial Mathematics and the literature, e.g. [4, 5] which are also the basis for this chapter. Here we concentrate only on those facts that are relevant for the numerical simulation of these financial processes.

The *contact point* S_f is defined as follows

$$V_P^{\text{am}}(S_f, t) = K - S_f \quad (0 < S_f < K) . \tag{7.0.2}$$

Note that S_f depends on the time t , i.e., $S_f = S_f(t)$. The contact point $S_f(t)$ can be characterized by

$$\begin{cases} V_P^{\text{am}}(S, t) > (K - S)^+ & \text{for } S > S_f(t) , \\ V_P^{\text{am}}(S, t) = K - S & \text{for } S \leq S_f(t) . \end{cases} \quad (7.0.3)$$

The location of the manifold $S_f(t)$, $t \in (0, T)$ is *unknown* a priori. Since this graph is the interface between the ‘exercise’ and the ‘no-exercise’ region (see below), we are faced with a so called *free boundary-value problem*. These kind of problems also occur in several other application, e.g., the propagation of waves in a medium (water waves, acoustic waves), plastic deformation processes and so on. Most of what is said in this chapter can also be applied to this kind of ‘industrial’ problems

The interpretation of the contact point for a put is as follows. The holder should exercise as soon as the price of the asset reaches $S_f(t)$. The corresponding time instant t_S is called *stopping time*. For a filtration \mathcal{F}_t , a random variable τ that is \mathcal{F}_t -measurable for all $t \geq 0$ is called *stopping time*.

Boundary conditions: The slope $\frac{\partial V}{\partial S}$ with which $V_P^{\text{am}}(S, t)$ touches the straight line $K - S$ at $S_f(t)$ is used as a boundary condition. Note that $K - S$ has the slope $-1 = \frac{\partial}{\partial S}(K - S)$, hence we require

$$\frac{\partial}{\partial S} V_P^{\text{am}}(S_f(t), t) = -1, \quad (7.0.4)$$

which results in a tangential touching. This is the so-called *high contact condition*.

For the (somewhat hypothetical) case of a *perpetual option* (i.e., for maturity $T = \infty$), we obtain an asymptotic condition (that can be calculated analytically). We will come to this point later.

In general, we obtain two boundary conditions, namely (7.0.2), (7.0.4).

For the American call, we need to include also dividend yields δ (otherwise they coincide with a European call option), i.e., the Black-Scholes equation takes the form

$$\frac{\partial}{\partial t} V(S, t) + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2}{\partial S^2} V(S, t) + (r - \delta) S \frac{\partial}{\partial S} V(S, t) - rV(S, t) = 0 \quad (7.0.5)$$

In fact, one can show that if $\delta = 0$ an early exercise does not pay off for a call. Since $V_C^{\text{am}} \geq S - Ke^{-r(T-t)}$, we have

$$V_C^{\text{am}} > S - K \quad \text{for } t < T \quad \text{and } r > 0,$$

i.e., for $\delta = 0$ American and European calls are identical: $V_C^{\text{am}} = V_C^{\text{eur}}$. In this case, the corresponding boundary conditions read as follows.

$$V_C^{\text{am}}(S_f(t), t) = S_f(t) - K, \quad (7.0.6)$$

$$\frac{\partial}{\partial S} V_C^{\text{am}}(S_f(t), t) = -1. \quad (7.0.7)$$

Black-Scholes Inequality

In the derivation of (7.0.5) early exercise was excluded. Following the lines of argumentation ‘= 0’ is now replaced by ‘ ≤ 0 ’ and we obtain *Black-Scholes-Inequality*, which holds for all (S, t) .

The inequality can be reformulated taking into account that the contact boundary S_f divides the half strip into two disjoint regulars.

$$\begin{aligned} \text{Put: } V_P^{\text{am}} &= K - S && \text{for } S \leq S_f \text{ (stop),} \\ V_P^{\text{am}} &\text{ solves (7.0.5)} && \text{for } S > S_f \text{ (hold).} \end{aligned} \quad (7.0.8)$$

$$\begin{aligned} \text{Call: } V_C^{\text{am}} &= S - K && \text{for } S \geq S_f \text{ (stop),} \\ V_C^{\text{am}} &\text{ solves (7.0.5)} && \text{for } S < S_f \text{ (hold).} \end{aligned} \quad (7.0.9)$$

This shows that also here the Black-Scholes equations have to be solved but with the additional problem of the *free boundary*.

7.1 The Binomial Method

We have already seen the binomial method for European options. Now, we give an easy and straightforward modification to American Options. It just amounts to including one projection step.

In fact, the only difference is the computation of V_{ji} , the approximation of $V(t_i, S_{ji})$ where $S_{ji} = S_0 u^j d^{i-j}$ so that (t_i, S_{ji}) can be seen as grid points. Then, we have

Theorem 7.1.1 Input: r, σ, S_0, K, M , choice of put or call

- $\Delta t = \frac{T}{M}$ u, d, p like in the European case
- $S_{00} = S_0$
- $S_{jM} = S_{00} u^j d^{M-j}$, $j = 0, 1, \dots, M$
 $S_{ji} = S_{00} u^j d^{i-j}$, $i = 1, \dots, M-1$ $j = 0, 1, \dots, y$

- $V_{j,M} = \begin{cases} (S_{jM} - K)^+ & \text{for Call} \\ (K - S_{jM})^+ & \text{for Put} \end{cases}$
- $V_{j,i} = \begin{cases} \max\{(S_{ji} - K)^+, e^{-r\Delta t}(pV_{j+1,i+1} + (1-p)V_{j,i+1})\}, & \text{Call} \\ \max\{(K - S_{ji})^+, e^{-r\Delta t}(pV_{j+1,i+1} + (1-p)V_{j,i+1})\}, & \text{Put} \end{cases}$
 $i < M$

Output: V_{00} is the approximation of $V(S_0, 0)$.

Example: See [14], Exercise 1.6.

7.2 Obstacle Problem

Another numerical method for solving the pricing problem for American Options uses the Black-Scholes inequality (see above). This can be seen as a particular instant of an *obstacle problem*, which is also common in several areas of application.

Example 7.2.1 *Given a membrane on which a force $-f$ acts on some domain $\Omega \subset \mathbb{R}^2$. The membrane is fixed on the boundary $\partial\Omega = \Gamma$ and the displacement of the membrane is bounded in Ω by a given function g (the obstacle).*

Then u is given as the solution of the following system

$$\left. \begin{array}{l} -\Delta u \geq f \\ u \geq g \\ (-\Delta u - f)(u - g) = 0 \end{array} \right\} \text{ in } \Omega, \quad u/\Gamma = 0 \quad (7.2.1)$$

Let us now subdivide Ω into the (unknown) contact zone

$$D_2 := \{x \in \Omega : u(x) = g(x)\}$$

and $D_1 := \Omega \setminus D_2$. Then we have

$$\left\{ \begin{array}{l} \bullet \text{ in } D_2 : u = g \ (\Rightarrow \Delta u = \Delta g < f) \\ \bullet \text{ in } D_1 : u > g \ \Rightarrow -\Delta u = f \end{array} \right. \quad (7.2.2)$$

i.e., the same behavior as for the Black-Scholes inequality:

if $V^{am} > \text{payoff} \Rightarrow \text{Black-Scholes-equation (7.0.1, ..., 7.0.5) holds,}$

if $V^{am} = \text{payoff} \Rightarrow \text{Black-Scholes-inequality.}$

As opposed to (7.2.2), the fomulation (7.2.1) does **not** involve the unknown D_2 , (7.2.1) is also called *linear complementary problem*. We use this for the design of a numerical method. If a solution u of (7.2.1) is determined, we can compute D_2 from it.

Variational Inequalities

It is known from the theory of partial differential equations that the classical (strong) formulation of boundary value problems is often not appropriate. The same also holds for inequalities.

Example 7.2.2 Consider the 1d elliptic PDE:

$$-u'' = f(x), x \in (0, 1), \quad u(0) = u(1) = 0,$$

which leads to the variational formulation of finding $u \in H_0^1(0, 1)$ such that

$$(\nabla u, \nabla v)_0 =: a(u, v) = (f, v)_0 \quad \forall v \in H_0^1(0, 1),$$

which is equivalent to the minimization problem

$$J(v) := \frac{1}{2}a(v, v) - (f, v)_0 \rightarrow \min \quad \text{for } v \in H_0^1(0, 1).$$

If the above minimization has to be constrained, i.e., $v \in H_0^1(0, 1)$ is replaced by a (convex) subset $K \subset H_0^1(0, 1)$, then we obtain a variational inequality (its analysis leads to the field of convex analysis).

The general form (which is also appropriate for American option pricing problems) reads as follows: Let H be a real Hilbert space and $K \subset H$ convex, $K \neq \emptyset$. Further, let $L : K \subset H \rightarrow H'$ be given. Then, one has to determine $u \in K$ such that

$$\langle L(u), v - u \rangle \geq 0 \quad \forall v \in K \tag{7.2.3}$$

where $\langle \cdot, \cdot \rangle$ is the duality pairing of H and its dual H' , i.e., for $w \in H'$, $v \in H$, the duality pairing is defined by $\langle w, v \rangle := w(v) \in \mathbb{R}$.

Remark 7.2.3 If K is a linear subspace of H , i.e., $v := u \pm z \in K$ for all $u, z \in K$. Inserting this in (7.2.3) yields on one hand $\langle L(u), z \rangle \geq 0$ and on the other hand, due to the linearity of $\langle L(u), \cdot \rangle$ that $\langle L(u), -z \rangle = -\langle L(u), z \rangle \geq 0$, i.e., we obtain $\langle L(u), z \rangle = 0$ for all $z \in K$, i.e., a variational equality.

Example 7.2.4 In Example 7.2.1, we obtain $K = \{v \in H_0^1(\Omega) : v \geq g \text{ in } \Omega \text{ a.e.}\}$ and

$$-\int_{\Omega} \nabla u \nabla (v - u) \, dx = a(u, v - u) \geq (f, v - u)_0, \quad \forall v \in K. \quad (7.2.4)$$

i.e., $L : K \rightarrow V'$ is defined on all $V > K$, i.e.,

$$\langle Lu, v \rangle = a(u, v) - \langle f, v \rangle.$$

The relation to the complementarity problem (7.2.1) is as follows. If (in addition to the above assumptions) $u \in H^2(\Omega)$, we have by integration by parts

$$\int_{\Omega} \nabla u \nabla (v - u) \, dx = \int_{\Omega} (-\Delta u)(v - u) \, dx,$$

i.e., (7.2.4) reads

$$(-\Delta u, v - u)_0 \geq (f, v - u)_0 \quad \forall v \in K \quad (7.2.5)$$

and since $u \in K$ we have $u \geq g$ in Ω a.e. Now let $\varphi \in H_0^1(\Omega)$, $\varphi \geq 0$ in Ω a.e. Then, setting $v := \varphi + u \in K$, we obtain

$$(-\Delta u, \varphi)_0 \geq (f, \varphi)_0 \quad \forall \varphi \geq 0, \quad (7.2.6)$$

i.e., $-\Delta u \geq f$ in Ω . Choosing $v := g$ in (7.2.5), we have

$$\begin{aligned} (-\Delta u - f, g - u)_0 &\geq 0 \text{ as well as} \\ \underbrace{(-\Delta u - f, g - u)_0}_{\geq 0} &\leq 0, \text{ so that in total} \end{aligned}$$

$(-\Delta u - f)(g - u) = 0$. Hence, u solves the complementary problem (7.2.1). On the other hand, let $u \in H^2(\Omega)$ solve (7.2.1), then we have from the first and second equation that

$$(-\Delta u - f, v - g)_0 \geq 0 \quad \forall v \in K$$

as well as

$$(-\Delta u - f, u - g)_0 = 0,$$

from the third equation. Next, by subtracting the latter two equations yields

$$\begin{aligned} 0 &\leq (-\Delta u - f, v - g)_0 - (-\Delta u - f, u - g)_0 \\ &= (-\Delta u - f, v - u)_0 \\ &= a(u, v - u) - (f, v - u)_0, \end{aligned}$$

i.e., (7.2.4).

Remark 7.2.5 *The equivalent minimization problem reads*

$$J(v) := \frac{1}{2}a(v, v) + (f, v)_0 \rightarrow \min_{v \in K}!$$

Hence, an obstacle problem may equivalently be formulated as

- free boundary value problem,
- linear complementary problem,
- variational inequality (if $u \in H^2(\Omega)$),
- minimization problem (if $u \in H^2(\Omega)$).

7.3 Finite Difference Methods

For notational convenience, we restrict ourselves to the 1D case and consider the complementary problem analogous to (7.2.1)

$$\left\{ \begin{array}{l} -u''(x) \geq f(x) \\ u(x) \geq g(x) \\ (-u''(x) - f(x))(u(x) - g(x)) = 0 \\ u(-1) = u(1) = 0, \quad u \in C^1[-1, 1] \end{array} \right\} \quad \forall x \in (-1, 1) \quad (7.3.1)$$

Again we introduce a simple equidistant grid

$$\Delta := \{-1 = x_0 < x_1 < \cdots < x_m = 1\}, \quad h := \frac{2}{m}, \quad m \in \mathbb{N} \\ x_i = -1 + ih, \quad 0 \leq i \leq m,$$

and use the central difference approximation

$$\begin{aligned} -u''(x_i) &\approx \frac{1}{h^2}(-u(x_{i-1}) + 2u(x_i) - u(x_{i+1})) , & f_i &:= f(x_i) \\ & & g_i &:= g(x_i) , \end{aligned}$$

i.e., we compute an approximation $u_i \approx u(x_i)$ by

$$\left\{ \begin{array}{l} u_0 = u_m = 0 \\ (-u_{i-1} + 2u_i - u_{i+1} - h^2 f_i)(u_i - g_i) = 0 \\ u_i \geq g_i \\ -u_{i-1} + 2u_i - u_{i+1} \geq h^2 f_i \end{array} \right\} \quad 1 \leq i \leq m-1, \quad (7.3.2)$$

or, in matrix-vector notation

$$\left\{ \begin{array}{l} (u - g)^T (Au - f) = 0, \\ u \geq g, \\ Au \geq f, \end{array} \right. \quad (7.3.3)$$

where $g := (g_1, \dots, g_{m-1})^T$, $f := h^2(f_1, \dots, f_{m-1})^T$, $u := (u_1, \dots, u_{m-1})^T$ and the system matrix

$$A := \begin{bmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{bmatrix} \in \mathbb{R}^{(m-1) \times (m-1)}.$$

Note that

$$\begin{aligned} (u - g)^T (Au - f) &= \sum_{i=1}^{m-1} (-u_{i-1} + 2u_i + u_{i+1} - h^2 f_i)(u_i - g_i) = 0 \\ \iff & (-u_{i-1} + 2u_i + u_{i+1} - h^2 f_i)(u_i - g_i) = 0 \quad \forall i \end{aligned}$$

in the case $-u_{i-1} + 2u_i + u_{i+1} \geq h^2 f_i$ and $u_i \geq g_i$ (i.e., if all signs are equal). Let us now describe a numerical (iterative) method for the solution of (7.3.3). First we note that (7.3.3) is equivalent to

$$\min\{Au - f, g - u\} = 0 \quad (\text{componentwise}). \quad (7.3.4)$$

This means either $u_i = g_u$ or $(Au)_i = f_i$, $1 \leq i \leq m-1$. We now consider the decomposition

$$A = D - L - U$$

where L is the lower left and U the upper right part of A . From now on, we assume that A is s.p.d, then

$$D_{ii} = a_{ii} > 0 \quad \forall 1 \leq i \leq m - 1 \quad (7.3.5)$$

and since

$$Au - f = D(u - D^{-1}(Lu + Uu + f))$$

the equations (7.3.4) is equivalent to

$$\min\{u - D^{-1}(Lu + Uu + f), g - u\} = 0, \quad (7.3.6)$$

or

$$u = \max\{D^{-1}(Lu + Uu + f), g\}. \quad (7.3.7)$$

The general idea is to modify appropriate iterative methods for $Au = f$ in such a way that (7.3.7) is incorporated in the iteration.

7.3.1 Classical Iterative Methods

The idea is to construct a suitable fixpoint iteration that converges towards the solution x^* of a linear system $Ax = b$.

If $A \in \mathbb{R}^{n \times n}$, choose a regular matrix $Q \in \mathbb{R}^{n \times n}$. Then

$$\begin{aligned} Ax = b &\iff Q^{-1}(Ax - b) = 0 \\ &\iff \phi(x) := \underbrace{(I - Q^{-1}A)}_{=:G} x + \underbrace{Q^{-1}b}_{=:c} = x, \end{aligned}$$

i.e., the solution of the linear system of equations is equivalent to the fixpoint problem $\phi(x) = x$. Then, Banach fixpoint theorem yields a corresponding iteration:

$$x_{k+1} = \phi(x_k) = Gx_k + c.$$

Then, we have the following standard result.

Lemma 7.3.1 *The fixpoint iteration converges for any initial value $x_0 \in \mathbb{R}^n$ if $\rho(G) < 1$, where*

$$\rho(G) := \max_{1 \leq j \leq n} |\lambda_j(G)|$$

denotes the spectral radius of G .

Proof: Consider the singular value decomposition of G

$$G = U\Sigma V^T$$

with orthogonal matrices U, V and $\Sigma = \text{diag}(\sigma_i)$, $\sigma_i = \lambda_i(G)^2 < 1$,

$$\Rightarrow \lim_{k \rightarrow \infty} \Sigma^k = 0$$

thus

$$\lim_{k \rightarrow \infty} G^k = u \left(\lim_{k \rightarrow \infty} \Sigma^k \right) V^T = 0 ,$$

which proves the claim. □

We still have the choice of the matrix Q . We describe some standard and well-known examples.

Richardson method: This corresponds to the choice $Q = \alpha I$, $\alpha \in \mathbb{R}$, i.e.,

$$x_{k+1} = x_k + \alpha(b - Ax_k)$$

The particular choice $\alpha \neq 1$ is known as *damped Richardson iteration*.

Jacobi method: With the decomposition $A = L + D + U$, where $D = \text{diag}(A)$ is the diagonal of A , one uses $Q = D^{-1}$. This results in the iteration:

$$x_{k+1} := (I - D^{-1}A)x_k + D^{-1}b = -D^{-1}(L + U)x_k + D^{-1}b.$$

For this method, we have the following convergence result.

Theorem 7.3.2 *If A is strictly diagonal dominant, i.e.,*

$$|a_{ii}| > \sum_{i \neq j} |a_{ij}| \quad \forall 1 \leq i \leq n ,$$

then the Jacobi iteration converges towards $x = A^{-1}b$ for all $x_0 \in \mathbb{R}^n$.

Proof: Follows by Lemma 7.3.1 since $G = I - D^{-1}A = -D^{-1}(L + R)$ and

$$\rho(D^{-1}(L + R)) \leq \|D^{-1}(L + R)\|_{\infty} = \max_i \sum_{i \neq j} \left| \frac{a_{ij}}{a_{ii}} \right| < 1 ,$$

which proves the theorem. □

Let us now consider the number of operations:

- $\mathcal{O}(N^2)$ for dense matrices $A \in \mathbb{R}^{N \times N}$,
- $\mathcal{O}(N)$ for sparse matrices

per step.

Gauß-Seidel method: Again, we use the decomposition $A = L + D + U$ and set $Q = D + L$, which yields the iteration:

$$\begin{aligned} x_{k+1} &= (I - (D + L)^{-1}A)x_k + (D + L)^{-1}b \\ &= -(D + L)^{-1}Ux_k + (D + L)^{-1}b, \end{aligned}$$

since

$$\begin{aligned} I - (D + L)^{-1}A &= (D + L)^{-1}(D + L - A) \\ &= (D + L)^{-1}(D + L - D - L - U). \end{aligned}$$

For this method, the following result is known.

Theorem 7.3.3 *The Gauß-Seidel method converges for all s.p.d. matrices $A \in \mathbb{R}^{n \times n}$.*

For the proof, we need some preparations. For a s.p.d. matrix $A \in \mathbb{R}^{n \times n}$, we consider the following scalar product.

$$(x, y)_A := x^T A y = (x, Ay), \quad x, y \in \mathbb{R}^n.$$

Note that $B^* = A^{-1}B^T A$ is the A -adjoint matrix i.e.,

$$(Bx, y)_A = (x, B^*y)_A$$

for all $y, x \in \mathbb{R}^n$. In fact, we have

$$(Bx, y)_A = x^T B^T A y = x^T A \underbrace{A^{-1}B^T A}_{=B^*} y = (x, B^*y)_A.$$

Any A -selfadjoint matrix B (which means that $B^* = B$) is called A -positiv if

$$(Bx, x)_A > 0 \quad \forall x \neq 0.$$

Lemma 7.3.4 *If $B := I - G^*G$ with $G \in \mathbb{R}^{n \times n}$, is A -positiv, then $\rho(G) < 1$.*

Proof: By assumption, B is A -positiv, i.e.

$$0 < (Bx, x)_A = (x, x)_A - (G^*Gx, x)_A = (x, x)_A - (Gx, Gx)_A$$

which means that $(x, x)_A > (Gx, Gx)_A$. Thus, we have for the A -norm

$$\|x\|_A := \sqrt{(x, x)_A}$$

that $\|x\|_A > \|Gx\|_A$. Finally

$$\rho(G) \leq \|G\|_A := \sup_{\|x\|_A=1} \frac{\|Gx\|_A}{\|x\|_A} < 1$$

since the supremum is attained due to the compactness of the unit ball with respect to $\|\cdot\|_A$, i.e. $\partial B_{1,A}(0) := \{x \in \mathbb{R}^n : \|x\|_A = 1\}$. \square

Proof of Theorem 7.3.3: Show that $B := I - G^*G$ is A -positive for $G = I - (D + L)^{-1}A$. Since $U^T = L$, we have

$$\begin{aligned} G^* &= I - A^{-1}A^T(D + L)^{-T}A \\ &= I - (D^T + L^T)^{-1}A = I - (D + U)^{-1}A, \end{aligned}$$

and thus by standard calculations

$$\begin{aligned} B &= I - G^*G = I - (I - (D + U)^{-1}A)(I - (D + L)^{-1}A) \\ &= I - I + (D + U)^{-1}A + (D + L)^{-1}A - \underbrace{(D + U)^{-1}A}_{\substack{=(D+U)^{-1}(D+U+L) \\ =I+(D+U)^{-1}L}} (D + L)^{-1}A \\ &= (D + U)^{-1}A - (D + U)^{-1}L(D + L)^{-1}A \\ &= (D + U)^{-1}D \underbrace{(D^{-1}(D + L) - D^{-1}L)}_{=I+D^{-1}L-D^{-1}L} (D + L)^{-1}A \\ &= (D + U)^{-1}D(D + L)^{-1}A. \end{aligned}$$

Hence, we obtain

$$\begin{aligned} (Bx, x)_A &= ((D + U)^{-1}D(D + L)^{-1}Ax, Ax) \\ &= (D(D + L)^{-1}Ax, \underbrace{(D + U)^{-T}Ax}_{(D+L)^{-1}}) \\ &= (D^{1/2}(D + L)^{-1}Ax, (D^{1/2}(D + L)^{-1}Ax)) \\ &= \underbrace{\|D^{1/2}(D + L)^{-1}Ax\|}_{\text{regular}} > 0 \end{aligned}$$

for $x \neq 0$ which shows that B is s.p.d. Thus, the claim follows from Lemma 7.3.4. \square

Relaxation Methods

For Gauß-Seidel, we have $G = I - (D + L)^{-1}A$. In order to speed up the convergence, one can introduce an additional parameter $\omega > 0$ and obtain the iteration matrix

$$G_\omega := I - \left(\frac{1}{\omega}D + L \right)^{-1} A ,$$

which means in particular, that for $\omega = 1$, we obtain the above mentioned Gauß-Seidel method. Thus, we obtain the iteration

$$x^{(k+1)} = \left(I - \left(\frac{1}{\omega}D + L \right)^{-1} A \right) x_k + \left(\frac{1}{\omega}D + L \right)^{-1} b .$$

The method is applied in practice as follows for a given iterate $x^{(k)}$

$$\left\{ \begin{array}{l} \text{For } i = 1, \dots, N \\ z_i^{(k+1)} = \frac{1}{a_{ii}} \left[\underbrace{- \sum_{m < i} a_{im} x_m^{(k+1)} - \sum_{m > i} a_{im} x_m^{(k)} + b_i}_{= (-Lx^{(k+1)} - Ux^{(k)} + b)_i} \right] \\ x_i^{(k+1)} = x_i^{(k)} + \omega (z_i^{(k+1)} - x_i^{(k)}) . \end{array} \right. \quad (7.3.8)$$

This can be seen as follows:

$$\begin{aligned} a_{ii}x_i^{(k+1)} &= a_{ii}x_i^{(k)} + \omega \left[- \sum_{m < i} a_{im}x_m^{(k+1)} - \sum_{m > i} a_{im}x_m^{(k)} + b_i - a_{ii}x_i^{(k)} \right] \\ \Leftrightarrow Dx^{(k+1)} &= Dx^{(k)} + \omega(-Lx^{(k+1)} - Ux^{(k)} + b - Dx^{(k)}) \\ \Leftrightarrow (D + \omega L)x^{(k+1)} &= (D - \omega \overbrace{(U + D)}^{=A-L})x^{(k)} + \omega b \\ \Leftrightarrow \omega \left(\frac{1}{\omega}D + L \right) x^{(k+1)} &= \omega \left[\left(\frac{1}{\omega}D + L \right) - \omega A \right] x^{(k)} + \omega b \end{aligned}$$

\Leftrightarrow

$$x^{(k+1)} = \left(I - \left(\frac{1}{\omega} D + L \right)^{-1} A \right) x^{(k)} + \left(\frac{1}{\omega} D + L \right)^{-1} b.$$

For $\omega < 1$, this is called a *damped iteration*, for $1 < \omega < 2$ it is called *over-relaxed*, the method is also known as SOR (*successive over relaxation*). Details can be found e.g. in [15] 2, §8.

Theorem 7.3.5 (Ostrowski, Reich) *For any s.p.d. matrix $A \in \mathbb{R}^{n \times n}$, the SOR method converges for all $0 < \omega < 2$.*

The proof can be found in any standard textbook on Numerical Analysis. The question naturally arises what might be an optimal choice for the parameter ω .

Definition 7.3.6 *A matrix $A \in \mathbb{R}^{n \times n}$ is called consistently ordered if the eigenvalues of the matrices*

$$J(\alpha) := D^{-1}(\alpha L + \alpha^{-1}U) \quad (\alpha \neq 0) \tag{7.3.9}$$

are independent of α , if $A = L + D + U$.

The following theorem can be found e.g. in [15].

Theorem 7.3.7 *If A is consistently ordered, then*

$$\rho(G_1) = \rho(J)^2, \quad J = J(1),$$

where $G_1 = I - (D + L)^{-1}A = -(D + L)^{-1}U$ is the iteration matrix of the Gauß-Seidel method. \square

Note that $-J(1) = -D^{-1}(L + U)$ is the iteration method of Jacobi, thus Theorem 7.3.7 says that the Jacobi method roughly needs the double number of iterations than Gauß-Seidel (if A is consistently ordered).

Now the optimal parameter is characterized by

$$\rho(G_{\omega_{\text{opt}}}) = \min_{\omega \in \mathbb{R}} \rho(G_\omega) = \min_{0 < \omega < 2} \rho(G_\omega)$$

and the following result is known (see again [15]).

Theorem 7.3.8 (Young, Varga) *Let A be consistently ordered and assume that $J = J(1)$ has only real eigenvalues such that $\rho(J) < 1$ (see Lemma 7.3.1). Then*

$$\omega_{opt} = \frac{2}{1 + \sqrt{1 - \rho(J)^2}} , \quad \rho(G_{\omega_{opt}}) = \omega_{opt} - 1 .$$

□

Remark 7.3.9 *Note that tridiagonal- and block-tridiagonal-matrices are consistently ordered which can easily be verified.*

7.3.2 Projected SOR-method for Complementary Problems

Now, we modify the above described SOR-method for solving the complementary problem

$$(Au - f)^T(u - g) = 0 , \quad u \geq g , \quad Au \geq f \quad (7.3.10)$$

which we have seen to be equivalent to (7.3.7), i.e.,

$$u = \max\{D^{-1}(Lu + Uu + f), g\} ,$$

if the matrix $A = D - L - U$ is s.p.d. We add a projection step in the SOR-method(7.3.8) (note: there we have used the decomposition $A = D + L + U$):

$$\left\{ \begin{array}{l} \text{For } i = 1, \dots, N \text{ do} \\ z_i^{(k+1)} = \frac{1}{a_{ii}}(Lu^{(k+1)} + Uu^{(k)} + f)_i \\ u_i^{(k+1)} = \max\{u_i^{(k)} + \omega(z_i^{(k+1)} - u_i^{(k)}), g_i\} \end{array} \right. \quad (7.3.11)$$

which is called *projected SOR-method*.

We aim to prove that (7.3.11) converges towards the solution u of (7.3.10). We need some preparations. The following proofs are taken from [5].

Lemma 7.3.10 *The problem (7.3.10) is equivalent to*

$$u \geq g , \quad J(u) = \min_{v \geq g} J(v) \quad (7.3.12)$$

where $J(v) := \frac{1}{2}v^T Av = f^T v$, if A is s.p.d.

Proof: Let u be a solution of (7.3.10) and let $v \geq g$, then

$$\begin{aligned}
J(v) - J(u) &= \frac{1}{2}v^T Av - \frac{1}{2}u^T Au + f^T(u - v) \\
&= \frac{1}{2}(v - u)^T A(v - u) + u^T Av - u^T Au + u^T f - v^T f \\
&= \underbrace{\frac{1}{2}(v - u)^T A(v - u)}_{\geq 0 \text{ since } A \text{ is s.p.d.}} + v^T(Au - f) - u^T(Au - f) \\
&\geq (v - u)^T(Au - f) \\
&= \underbrace{(v - g)^T}_{\geq 0} \underbrace{(Au - f)}_{\geq 0 \text{ by (7.3.10)}} - \underbrace{(u - g)^T(Au - f)}_{=0 \text{ by (7.3.10)}} \geq 0,
\end{aligned}$$

i.e., $J(u) \leq J(v)$ for all $v \geq g$, i.e., u solves (7.3.12).

On the other hand, let $u \geq g$ solve (7.3.12). Let $v^{(k)} := u + \varepsilon \delta_k$, $\varepsilon > 0$ and denote by $\delta_k = (\delta_{1,k}, \dots, \delta_{n,k})^T$ the k -th canonical vector. This means that $v^{(k)} \geq u \geq g$ and

$$\begin{aligned}
0 \leq J(v^{(k)}) - J(u) &= \frac{\varepsilon^2}{2} \delta_k^T A \delta_k + \varepsilon \delta_k^T (Au - f) \\
&= \frac{\varepsilon^2}{2} a_{kk} + \varepsilon (Au - f)_k, \quad \forall \varepsilon > 0,
\end{aligned}$$

which implies $0 \leq (Au - f)_k + \frac{\varepsilon}{2} a_{kk} \rightarrow (Au - f)_k$ as $\varepsilon \rightarrow 0$ for all k , i.e. $Au \geq f$.

Now suppose $(Au - f)_k > 0$ and $u_k \geq g_k$ for some k . Choose $\varepsilon > 0$ small enough such that $w^{(k)} := u - \varepsilon \delta^k \geq g_k$. This implies $0 \leq J(w^{(k)}) - J(u) = \frac{\varepsilon^2}{2} a_{kk} - \varepsilon (Au - f)_k < 0$ for ε small enough, which finally gives $(Au - f)^T(u - g) = 0$. \square

Theorem 7.3.11 (Cryer) *Let $A \in \mathbb{R}^{n \times n}$ s.p.d., $b, f \in \mathbb{R}^n$, $1 < \omega < 2$, then $\{u^{(k)}\}_{k \in \mathbb{N}}$ defined by (7.3.11) converges towards the unique solution u of (7.3.10).*

Remark 7.3.12 *The latter theorem also states that the complementary problem (7.3.10) has a unique solution.*

Proof of Theorem 7.3.11:

We split the proof in two parts.

1.) **Uniqueness of solution:**

Let w_1, w_2 be two solutions of (7.3.10) and (7.3.12), respectively. Thus, we have by Lemma 7.3.10

$$\begin{aligned}
0 &= J(w_1) - J(w_2) \\
&= \frac{1}{2} \underbrace{(w_1 - w_2)^T A (w_1 - w_2)}_{\geq 0} + \underbrace{w_1^T (Aw_2 - f) - w_2^T (Aw_2 - f)}_{=0} \\
&= (w_1 - w_2)^T (Aw_2 - f) \\
&= \underbrace{(w_1 - g)^T (Aw_2 - f)}_{\geq 0} - \underbrace{(w_2 - g)^T (Aw_2 - f)}_{=0} \\
&\geq \frac{1}{2} (w_1 - w_2)^T A (w_1 - w_2) \geq 0,
\end{aligned}$$

i.e., $0 = (w_1 - w_2)^T A (w_1 - w_2)$, which means that $w_1 - w_2 = 0$.

2.) **Existence of a solution:**

The idea is to show convergence of $u^{(k)}$ in (7.3.11)

a) For all i, k there exists some $\omega_k \in [0, \omega]$ such that for $u_i^{(k+1)} := u_i^{(k)} + \omega_{i,k} (z_i^{(k+1)} - u_i^{(k)})$ we have

- If $g_i \leq u_i^{(k)} + \omega (z_i^{(k+1)} - u_i^{(k)})$ and thus $\omega_{i,k} = \omega$.
- If $g_i > u_i^{(k)} + \omega (z_i^{(k+1)} - u_i^{(k)})$ and thus we have $u_i^{(k+1)} = g_i$ since $u_i^{(k)} \geq g_i$
(because $u_i^{(k)} = \max\{\dots, g_i\}$), we have $z_i^{(k+1)} - u_i^{(k)} < 0$ and hence

$$\omega_{i,k} := \frac{u_i^{(k)} - g_i}{u_i^{(k)} - z_i^{(k+1)}} \geq 0$$

and $\omega_{i,k} < \omega$. This implies that $u_i^{(k)} + \omega_{i,k} (z_i^{(k+1)} - u_i^{(k)}) = u_i^{(k)} - u_i^{(k)} + g_i = g_i = u_i^{(k+1)}$.

b) Set $u^{(k,i)} = (u_1^{(k+1)}, \dots, u_i^{(k+1)}, u_{i+1}^{(k)}, \dots, u_N^{(k)})^T$ and

$$J_j := J(u^{(k,i)}), \quad j := N(k-1) + i$$

Note that

$$u^{(k,i,0)} = (u_1^{(k+1)}, \dots, u_N^{(k+1)})^T := u^{(k,N+1)}$$

and

$$(u^{(k,i)} - u^{(k,i-1)}) = (u_i^{(k+1)} - u_i^{(k)}) \delta_i \quad (7.3.13)$$

for $\delta_i = (\delta_{1,i}, \dots, \delta_{N,i})^T$. By (7.3.11)

$$\begin{aligned}
a_{ii}(z_i^{(k+1)} - u_i^{(k)}) &= (Lu^{(k+1)} + Uu^{(k)} + f)_i - a_{ii}u_i^{(k)} \\
&= -(Au^{(k,i)})_i + f_i \\
&= -(Au^{(k,i)} - f)_i
\end{aligned} \tag{7.3.14}$$

Thus, we obtain

$$\begin{aligned}
\Rightarrow J_j - J_{j-1} &= J(u^{(k,i)}) - J(u^{(k,i-1)}) \\
&= \frac{1}{2}(u^{(k,i)} - u^{(k,i-1)})^T A(u^{(k,i)} - u^{(k,i-1)}) \\
&\quad + (u^{(k,i)} - u^{(k,i-1)})^T (Au^{(k,i)} - f) \\
&= \frac{1}{2}a_{ii}(u_i^{(k+1)} - u_i^{(k)})^2 - a_{ii}(u_i^{(k+1)} - u_i^{(k)}) \underbrace{(z_i^{(k+1)} - u_i^{(k)})}_{= \frac{1}{\omega_{i,k}}(u_i^{(k+1)} - u_i^{(k)})} \\
&= -\frac{a_{ii}}{2} \left(\frac{2}{\omega_{i,k}} - 1 \right) (u_i^{(k+1)} - u_i^{(k)})^2 \\
&\leq -\underbrace{\frac{a_{ii}}{2}}_{>0} \underbrace{\left(\frac{2}{\omega} - 1 \right)}_{>0} \underbrace{(u_i^{(k+1)} - u_i^{(k)})^2}_{\geq 0} \quad \text{if } \omega_{i,k} > 0 \\
&\leq 0.
\end{aligned}$$

If $\omega_{i,k} = 0$ we have $u_i^{(k+1)} = u_i^{(k)}$, hence $J_j = J_{j-1}$, i.e., $J_j \searrow$ ($j \rightarrow \infty$).

On the other hand $J_j = \frac{1}{2} \underbrace{u^{(k,i)T} Au^{(k,i)}}_{\geq 0} - \underbrace{f^T u^{(k,i)}}_{\leq c}$ which implies

the existence of the limit $J = \lim_{j \rightarrow \infty} J_j$.

c) A standard estimate yields for any component index i

$$\begin{aligned}
|u_i^{(k+1)} - u_i^{(k)}| &= \left(\frac{2}{a_{ii}(\frac{2}{\omega_{i,k}} - 1)} (J_{j-1} - J_j) \right)^{1/2} \\
&\leq \left(\frac{2}{\min_i a_{ii}(\frac{2}{\omega_{i,k}} - 1)} (J_{j-1} - J_j) \right)^{1/2} \rightarrow 0,
\end{aligned}$$

which means that $\{u_i^{(k)}\}_k$ is a Cauchy sequence. Thus, there exists some u_i such that $\lim_{k \rightarrow \infty} u_i^{(k)} = u_i$.

d) If we define

$$z_i := \lim_{k \rightarrow \infty} z_i^{(k+1)} = \frac{1}{a_{ii}}(Lu + Uu + f)_i = u_i - \frac{1}{a_{ii}}(Au - f)_i$$

we get

$$u_i = \max\{u_i + \omega(z_i - u_i), g_i\} = \max\{u_i - \omega \frac{1}{a_{ii}}(Au - f)_i, g_i\}.$$

Thus, we have $\min\{\frac{\omega}{a_{ii}}(Au - f)_i, u_i - g_i\} = 0$, which is equivalent to (7.3.10), which proves the theorem. \square

Chapter 8

Exotic Options

All nonstandard options (i.e., non American and European) are called exotic options. Nowadays there is a whole variety of such tailor-made options. The general observation for such options is that transformation methods to simple pde's like the heat equations do not work. The corresponding pde has to be solved directly which causes several numerical difficulties. Let us just collect some main types of exotic options without going into details and without claiming completeness:

- compound options
- chooser options
- binary options
- path-dependent options: barrier options, lookback options, Asian options.

Some of these options can be reduced to the Black–Scholes equation. Here, we focus on some aspects for those cases where this reduction is not possible.

8.1 Asian Options

As an example, let us consider Asian options whose characteristics is the dependency of its price from an average of values of the underlying S_t at previous times. Asian options can be of European and American style. The

average can be taken in an arithmetic sense

$$\frac{1}{m} \sum_{i=1}^n S_{t_i}, \quad \frac{1}{T} \int_0^T S_t dt$$

or a geometric sense

$$\left(\prod_{i=1}^n S_{t_i} \right)^{1/n}, \quad \exp \left(\frac{1}{T} \int_0^T \log(S_t) dt \right).$$

Let us denote by \bar{S} one of these averages.

Definition 8.1.1 *With the average \bar{S} of S_t , the payoff function of an Asian option is defined as*

$$\begin{array}{ll} (\bar{S} - K)^+, (K - \bar{S})^+ & \text{average strike call / put} \\ (S_T - \bar{S})^+, (\bar{S} - S_T)^+ & \text{average price call / put.} \end{array}$$

For the modelling, the average \bar{S} is exposed as

$$A_t := \int_0^t f(S_\theta, \theta) d\theta, \quad (8.1.1)$$

where $f(S, t)$ models the type of average. The stochastic behaviour of A_t is described in the Black–Scholes model as

$$dA_t = a_A(t)dt + b_A dW_t. \quad (8.1.2)$$

In (8.1.1) we would just have

$$a_A(t) = f(S_t, t), \quad b_A \equiv 0.$$

The advantage of (8.1.2) is that Itô's lemma can be used to derive the following SDE for the price $V = V(S, A, t)$

$$dV_t = \left(\frac{\partial}{\partial t} V + \mu S \frac{\partial}{\partial S} V + \frac{1}{2} \sigma^2 S^2 \frac{\partial^2}{\partial S^2} V + f(S, t) \frac{\partial}{\partial A} V \right) dt + \sigma S \frac{\partial}{\partial S} V dW_t, \quad (8.1.3)$$

or as a pde

$$\frac{\partial}{\partial t}V + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2}{\partial S^2}V + rS \frac{\partial}{\partial S}V + f(S, t) \frac{\partial}{\partial A}V - rV = 0. \quad (8.1.4)$$

Hence, we obtain the additional term

$$f(S, t) \frac{\partial}{\partial A}V.$$

Note that no second derivative of V with respect to A is present which makes the problem convection dominated or even hyperbolic.

First one can reduce the problem to a 1D problem (details see [14], 216-219, exercise) by introducing the auxiliary variable

$$R_t := \frac{1}{S_t} \int_0^t S_\theta d\theta, \quad (8.1.5)$$

where we restrict ourselves for simplicity to the arithmetic average, i.e., $f(S, t) = S$. Then, one assumes a separation of the form

$$V(S, A, t) = S \cdot H(R, t) \quad (8.1.6)$$

for some function H . Here, R is an independent variable. Then, one gets the following problem

$$\frac{\partial}{\partial t}H + \frac{1}{2}\sigma^2 R^2 \frac{\partial^2}{\partial R^2}H + (1 - rR) \frac{\partial}{\partial R}H = 0 \quad (8.1.7)$$

$$H = 0 \quad \text{for } R \rightarrow \infty \quad (8.1.8)$$

$$\frac{\partial}{\partial t}H + \frac{\partial}{\partial R}H = 0 \quad \text{for } R = 0 \quad (8.1.9)$$

$$H(R_T, T) = \left(1 - \frac{1}{T}R_T\right)^+. \quad (8.1.10)$$

The obvious advantage is the reduction of the dimension of the problem.

8.2 Convection–Diffusion Problems

We can view (8.1.7) as a particular example of a convection-diffusion problem of the form

$$\partial_t u - (au' - cu)' = f \quad (8.2.1)$$

for $u = u(t, x)$. In fact

$$\begin{aligned} (au' - cu)' &= au'' + a'u' - cu' - c'u \\ &= au'' + (a' - c)u' - c'u. \end{aligned}$$

Such equations appear in many applications e.g. transport, reaction in porous media, semiconductors and so on. The behaviour of such equations can be characterised by the **global Peclét number**

$$Pe := \frac{\|c\|_\infty \text{diam}(\Omega)}{\|a\|_\infty}, \quad (8.2.2)$$

where Ω is the spatial domain, $x \in \Omega$. The global Peclét number measures the proportion of reaction/convection to diffusion. For example, one has

- $Pe \sim 25$ (groundwater transport)
- $Pe \sim 10^7$ (semiconductor)

A little bit vague a problem (8.2.1) is called **convection dominated** if

$$Pe \gg 1.$$

Example 8.2.1 For $k > 0$ consider

$$\begin{cases} -(ku' + u)' = 0 & \text{in } \Omega := (0, 1) \\ u(0) = u(1) = 0 \end{cases} \quad (8.2.3)$$

which has the exact solution

$$u(x) = \frac{1 - \exp(x/k)}{1 - \exp(1/k)}.$$

The global Peclét number is given as $Pe = \frac{1}{k}$. Even for moderate values of Pe (e.g. $Pe \sim 100$), we observe a strong **boundary layer**:



Let us describe what happens for the numerical solution. Let us consider a FDM w.r.t. an equidistant grid with step size

$$h = \frac{1}{M+1}$$

and use central differences for both terms

$$\left(-\frac{2k}{h} - 1\right) u_{i-1} + \frac{4k}{h} u_i + \left(-\frac{2k}{h} + 1\right) u_{i+1} = 0, \quad 1 \leq i \leq M. \quad (8.2.4)$$

Using the ansatz $u_i = \lambda^i$ gives an exact solution

$$u_i = \frac{1 - \left(\frac{2k+h}{2k-h}\right)^i}{1 - \left(\frac{2k+h}{2k-h}\right)^{M+1}}.$$

For $2k < h$ (which is realistic for $k \sim 10^{-7}$) we obtain heavy oscillations in the numerical solution. These oscillations do **not** occur in the exact solution. For $2k > h$, the oscillations disappear but one may lack convergence.

Let us describe some of the technical problems that occur here. Let us consider the pde

$$-\varepsilon u'' + cu' + ru \quad (8.2.5)$$

which is of the form (8.2.1). The bilinear form associated to (8.2.5) reads

$$a(u, v) = (\varepsilon u', v') + (cu', v) + (ru, v), \quad u, v \in H_0^1(\Omega) =:?? \quad (8.2.6)$$

If we assume that

$$r - \frac{1}{2}c' \geq r_0$$

for some constant $r_0 > 0$ one can easily show (exercise)

$$a(v, v) \geq \tilde{\alpha} \|v\|_\varepsilon^2, \quad \tilde{\alpha} := \min\{1, r_0\}$$

where

$$\|v\|_\varepsilon := (\varepsilon|v|_1^2 + \|v\|_0^2)^{1/2} \quad (8.2.7)$$

is the ε -weighted H^1 -norm. Performing the complete error analysis gives

$$\|u - u_h\|_\varepsilon \leq Ch\varepsilon^{-3/2}, \quad \varepsilon \rightarrow 0^+. \quad (8.2.8)$$

Note that the factor $\varepsilon^{-3/2}$ can be extremely large.

8.3 SUPG-method

The SUPG (streamline upwind Petrov-Galerkin) method is maybe the most commonly used method for the numerical solution of convection-dominated problems. The main idea was introduced by Hughes and Brooks in 1979. In order to describe the method, we consider the model problem in 2D, namely

$$\begin{aligned} L_\varepsilon u &:= -\varepsilon\Delta u + c \cdot \nabla u + ru = f && \text{in } \Omega, \\ u &= 0 && \text{on } \Gamma = \partial\Omega, \end{aligned} \quad (8.3.1)$$

with $\varepsilon > 0$, coefficient functions $c \in C^1(\bar{\Omega}, \mathbb{R}^n)$, $r \in C(\bar{\Omega})$ and a given right-hand side $f \in L_2(\Omega)$. As above, we assume

$$r - \frac{1}{2}\nabla \cdot c \geq r_0 > 0$$

for some constant $r_0 > 0$. The bilinear form reads

$$a(u, v) := \int_{\Omega} \left[\varepsilon \nabla u \cdot \nabla v + c \cdot \nabla u v + r uv \right] dx$$

for $u, v \in V := H_0^1(\Omega)$, so that the variational formulation reads

$$u \in V : \quad a(u, v) = (f, v)_{0;\Omega}, \quad v \in V. \quad (8.3.2)$$

Using a standard finite element discretization with test and trial spaces

$$V_h := \{v_h \in V : v_h|_K \in \mathcal{P}_k(K), \quad K \in \mathcal{T}_h\},$$

we obtain a standard error estimate

$$\inf_{v_h \in V_h} \|u - v_h\|_{l,K} \leq c_{\text{err}} h_K^{k+1-l} |u|_{k+1,K}$$

if $u \in H^{k+1}(\Omega)$ for $l \in \{0, 1, 2\}$, all $K \in \mathcal{T}_h$ and a constant c_{err} . This is a standard finite element error estimate (Jackson inequality). Moreover, there is also an *inverse inequality*

$$\|\Delta v_h\|_{0,K} \leq \frac{c_{\text{in}}}{h_K} |v_h|_{1,K}, \quad v_h \in V_h$$

for all $K \in \mathcal{T}_h$. Such an estimate is based upon the finite dimensionality of V_h . Note that the two constants $c_{\text{err}}, c_{\text{in}} > 0$ do not depend on u and v_h , nor on K .

The basic idea to overcome the above described stability problems is to add suitable (local) weighted residuals to the variational formulation (8.3.2). Interpreting the original problem in L_2 and restricting it to each element K gives

$$L_\varepsilon u = f \quad \text{a.e. in } K, \quad K \in \mathcal{T}_h.$$

Now we multiply this equation with test functions

$$\tau(v_h)|_K,$$

where $\tau : L_2(\Omega) \rightarrow L_2(\Omega)$ is a suitable function to be detailed later. Moreover, we introduce scaling factors

$$\delta_K \in \mathbb{R}, \quad K \in \mathcal{T}_h,$$

and obtain

$$\sum_{K \in \mathcal{T}_h} \delta_K (-\varepsilon \Delta u + c \cdot \nabla u + ru, \tau(v_h))_{0,K} = \sum_{K \in \mathcal{T}_h} \delta_K (f, \tau(v_h))_{0,K}.$$

This equation is added to the discrete variational problem and we obtain

$$\begin{aligned} a_h(u, v_h) &:= a(u, v_h) + \sum_{K \in \mathcal{T}_h} \delta_K (-\varepsilon \Delta u + c \cdot \nabla u + ru, \tau(v_h))_{0,K} \\ (f, v_h)_h &:= (f, v_h)_{0,\Omega} + \sum_{K \in \mathcal{T}_h} \delta_K (f, \tau(v_h))_{0,K} \end{aligned}$$

and the following new discrete problem results

$$u_h \in V_h : \quad a_h(u_h, v_h) = (f, v_h)_h, \quad v_h \in V_h.$$

If the original and the new discrete problem have unique solutions, we obtain the error equation (Galerkin orthogonality)

$$a_h(u - u_h, v_h) = 0, \quad v_h \in V_h.$$

Remark 8.3.1 (a) *Well-known choices for τ are*

- $\tau(v_h) := c \cdot \nabla v_h$ (*streamline-diffusion method*)
- $\tau(v_h) := -\varepsilon \Delta v_h + c \cdot \nabla v_h + r v_h$ (*Galerkin/least Squares*).

(b) *One can show that the additional term adds some artificial diffusion which is the reason for the stabilization.*

With quite some technicalities, one can prove the following error estimate.

Theorem 8.3.2 *Let the parameters be chosen as*

$$\delta_K = \begin{cases} \delta_1 \frac{h_K^2}{\varepsilon}, & \text{if } Pe_K \leq 1, \\ \delta_2 h_K, & \text{if } Pe_K > 1, \end{cases}$$

with $\delta_1, \delta_2 > 0$ are independent of K and ε is chosen in such a way that

$$0 < \delta_K \leq \frac{1}{2} \left\{ \frac{h_K^2}{\varepsilon c_{\text{inv}}^2}, \frac{r_0}{\|r\|_{0,\infty,K}} \right\}.$$

If the weak solution is in $K^{k+1}(\Omega)$, then

$$\|u - u_h\|_{\text{sd}} \leq C(\sqrt{\varepsilon} + \sqrt{h})h^k |u|_{k+1,\Omega},$$

where $\|\cdot\|_{\text{sd}}$ denotes the streamline diffusion norm

$$\|u\|_{\text{sd}}^2 := \varepsilon |v|_1^2 + r_0 \|v\|_0^2 + \sum_{K \in \mathcal{T}_h} \delta_K \|c \cdot \nabla u\|_{0,K}^2.$$

□

Remark 8.3.3 (a) *The reason for the name Petrov-Galerkin comes from the fact that the above streamline diffusion method can also be interpreted as a variational problem where one uses different test and trial spaces. This is called a Petrov-Galerkin method.*

(b) *The major drawback is that still there is a dependence on negative powers of ε . Alternatives are Finite Volume methods or Discontinuous Galerkin methods.*

Bibliography

- [1] D. Braess, *Finite Elemente*, Springer, 1997.
- [2] D. Braess, *Finite Elements*, Cambridge University Press, Cambridge, 2001.
- [3] C. W. Clenshaw and A. R. Curtis, *Numer. Math.* 2, 197–205.
- [4] C. Geiger, C. Kanzow, *Numerische Verfahren zur Lösung unrestringierter Optimierungsaufgaben*, Springer 1999.
- [5] M. Günther, A. Jüngel, *Finanzderivate mit MATLAB*, Vieweg 2003.
- [6] C. Großmann and H.-G. Roos, *Numerik partieller Differentialgleichungen*, Second edition, Teubner, Stuttgart, 1994.
- [7] P. E. Kloeden and E. Platen, *Stochastic Differential Equations*, Third edition, Springer 1999.
- [8] R. Mallier, G. Alobaidi, Laplace transforms and American options, *Appl. Math. Finance* 7 (2000), 241-256.
- [9] M. Matsumoto and T. Nishimura, Mersenne Twister: A 623-dimensionally equidistributed uniform pseudorandom number generator, *ACM Transactions on Modeling and Computer Simulations*, 1998.
- [10] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*, SIAM 1992.
- [11] H. Niederreiter and K. Petras, *Numerische Methoden in der Finanzmathematik*, Manuskript, TU Braunschweig, <http://www-public.tu-bs.de:8080/petras/lva/finanz/vorl.html>

- [12] A. Quateroni, R. Sacco, F. Saleri, *Numerische Mathematik 2*, Springer 2002.
- [13] Ch. Schwab., R.A. Todor, *Sparse Finite Elements for Elliptic Problems with Stochastic Loading*, erscheint in *Numer. Mathematik*, 2003.
- [14] R. Seydel, *Tools for Computational Finance*, Springer 2002.
- [15] J. Stoer, R. Bulirsch, *Numerische Mathematik 2*, Springer, Berlin, Heidelberg 2000.
- [16] S. Stojanovic, *Computational Financial Mathematics Using MATHEMATICA*, Birkhauser, 2003.