Universitext

Rüdiger U. Seydel

Tools for Computational Finance

Sixth Edition

Springer

Universitext

Universitext

Universitext is a series of textbooks that presents material from a wide variety of mathematical disciplines at master's level and beyond. The books, often well class-tested by their author, may have an informal, personal even experimental approach to their subject matter. Some of the most successful and established books in the series have evolved through several editions, always following the evolution of teaching curricula, to very polished texts.

Thus as research topics trickle down into graduate-level teaching, first textbooks written for new, cutting-edge courses may make their way into *Universitext.*

More information about this series at http://www.springer.com/series/223

Rüdiger U. Seydel

Tools for Computational Finance

Sixth Edition

Springer

Rüdiger U. Seydel
Mathematisches Institut
Universität zu Köln
Germany

ISSN 0172-5939 ISSN 2191-6675 (electronic)
Universitext
ISBN 978-1-4471-7337-3 ISBN 978-1-4471-7338-0 (eBook)
DOI 10.1007/978-1-4471-7338-0

Library of Congress Control Number: 2017945224

Mathematics Subject Classification (2010): 91-01, 91-08, 91G20, 91G60, 65-01

Previous editions 1-4 published by Springer-Verlag Berlin Heidelberg

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer-Verlag London Ltd.
The registered company address is: 236 Gray's Inn Road, London WC1X 8HB, United Kingdom

Preface to the Sixth Edition

Five years after the fifth edition came out, there is a need to include additional results and to improve explanations of methods and algorithms further. Besides numerous enhancements of small details, there are new subjects.

New in the sixth edition are, for example, methods for smoothing in Chap. 1 and an introduction into basic aspects of efficiency. In Chap. 2, acceptance–rejection methods for generating random numbers are explained, with application to the ziggurat algorithm for calculating normal variates. Chapter 3 on Monte Carlo methods now includes a subsection on positive solutions and an outline of the antithetic variance reduction. Iterative approaches in Chap. 4 have become less important, in favor of direct methods.

To support the important role tree methods play in practice, an entire new appendix (Appendix D) is devoted to these methods. This appendix includes trinomial methods, multidimensional trees, and implied trees for variable volatility. Further, how to handle discrete dividends is explained.

And the text is enriched by more figures. To facilitate understanding, many of the figures have been recalculated to become colored. Additional formalized exercises are included, and numerous hints at informal exercises are spread throughout the text.

On the technical side, the entire book has been transferred from *plainTeX* to *LaTeX*. This has offered plenty of occasions to work the book over thoroughly. Additional colored figures can be found in the collection *Topics for Computational Finance* (shortly *Topics fCF*) on the Internet platform www.compfin.de.

Köln, Germany
April 2017

Rüdiger U. Seydel

Preface to the Fifth Edition

Financial engineering and numerical computation are genuinely different disciplines. But in finance, many computational methods are used and have become indispensable. This book explains how computational methods work in financial engineering. The main focus is on computational methods; financial engineering is the application. In this context, the numerical methods are *tools*, the tools for computational finance.

Faced with the vast and rapidly growing field of financial engineering, we need to choose a subarea to avoid overloading the textbook. We choose the attractive field of option pricing, a core task of financial engineering and risk analysis. The broad field of option pricing is both ambitious and diverse enough to call for a wide range of computational tools. Confining ourselves to option pricing enables a more coherent textbook and avoids being distracted away from computational issues. We trust that the focus on option-related methods is representative of, or least helpful for, the entire field of computational finance.

The book starts with an introductory Chap. 1, which collects financial and stochastic background. The remaining parts of the book are devoted to computational methods. Organizing computational methods, roughly speaking, leads to distinguish stochastic and deterministic approaches. By "stochastic methods," we mean computations based on random numbers, such as Monte Carlo simulation. Chapters 2 and 3 are devoted to such methods. In contrast, "deterministic methods" are frequently based on solving partial differential equations. This is discussed in Chaps. 4, 5, and 6. In the computer, finally, everything is deterministic. The distinction between "stochastic" and "deterministic" is mainly to motivate and derive different approaches.

All of the computational methods must be adapted to the underlying model of a financial market. Here, we meet different kinds of stochastic processes, from geometric Brownian motion to Lévy processes. Based on the chosen process, an option model is selected. The classical choice is the Black–Scholes model for vanilla options with one underlying asset. This benchmark market model is "complete" in that all claims can be replicated. Established by Black, Merton, Scholes, and others, this model is the main application of methods explained in Chaps. 2–6.

Chapter 7 goes beyond and addresses more general models. Allowing for jump processes, transaction costs, multiasset underlyings, or more complicated payoffs leads to incomplete markets. Computational methods for incomplete markets are briefly discussed in Chap. 7.

This book has been published in several editions. The first German edition (2000) was mainly absorbed by the Black–Scholes equation. Later editions (first English edition 2002) were carefully opened to more general models and a wider selection of methods. The book has grown with the development of the field. Faced with a large variety of possible computational tools, this book attempts to balance the need for a sufficient number of powerful algorithms with the limitations of a textbook. The balance has been gradually shifting over the years and editions. Numerous investigations in our research group have influenced the choice of covered topics. We have implemented and tested many dozens of algorithms and gained insight and experience. A significant part of this knowledge has entered the book.

Readership

This book is written from the perspective of an applied mathematician. The level of mathematics is tailored to advanced undergraduate science and engineering majors. Apart from this basic knowledge, the book is self-contained and can be used for a course on the subject. The intended readership is interdisciplinary and includes professionals in financial engineering, mathematicians, and scientists of many other fields.

An expository style may attract a readership ranging from students to practitioners. Methods are introduced as tools for immediate application. Formulated and summarized as algorithms, a straightforward implementation in computer programs should be possible. In this way, the reader may learn by computational experiment. *Learning by calculating* will be a possible way to explore several aspects of the financial world. In some parts, this book provides an algorithmic introduction to computational finance. To keep the text readable for a wide audience, some aspects of proofs and derivations are exported to exercises at which hints are frequently given.

New in the Fifth Edition

The revisions to this fifth edition are much more extensive than those of previous editions. Compared to the fourth edition, the page count has increased by about 100 pages. The main addition is Chap. 7, which is devoted to incomplete markets. It begins with an introduction to nonlinear Black–Scholes-type partial differential equations, as they arise from considering transaction costs or ranges for a stochastic volatility. Numerical approaches require instruments that converge to viscosity

solutions. These solutions are introduced in an appendix. The role of monotonicity of numerical schemes is outlined. Lévy processes, with a focus on Merton's jump diffusion and a numerical approach to the resulting partial integrodifferential equation, are then addressed. The chapter ends with an exposition on how the Fourier transform can be applied to option pricing. To complete the introduction of more general models and methods, the Dupire equation is outlined in a new appendix.

In addition to the new Chap. 7, several larger extensions and new sections have been written for this edition. The calculation of Greeks is described in more detail, including the method of adjoints for a sensitivity analysis (new Sect. 3.7). Penalty methods are introduced and applied to a two-factor model in the new Sect. 6.7. More material is presented in the field of analytical methods; in particular, Kim's integral representation and its computation have been added to Chap. 4. Tentative guidelines on how to compare different algorithms and judge efficiency are given in the new Sect. 4.9. The chapter on finite elements has been extended with a discussion of two-asset options.

Apart from additional material listed above, the entire book has been thoroughly revised. The clarity of the expository parts has been improved; all sections have been tested in the classroom. Numerous amendments, further figures, exercises, and many references have been added. For example, the principal component analysis and its applications are included, and the role of different boundary conditions is outlined in more detail.

How to Use This Textbook

Exercises are stated at the end of each chapter. They range from easy routine tasks to laborious projects. In addition to these explicitly formulated exercises, plenty of "hidden" exercises are spread throughout the book, with comments such as "the reader may check." Of course, the reader is encouraged to fill in those small intermediate steps that are excluded from the text.

This book explains the basic ideas of several approaches, presenting more material than is accomplishable in one semester. The following guidelines have proved successful in teaching:

First Course:

Chapter 1 without Sect. 1.6.2
Chapter 2
Chapter 3 without Sect. 3.7
Chapter 4, with one analytic method out of Sect. 4.8
and without Sect. 4.9
Chapter 6, or parts of it

Second Course:

The remaining parts, in particular
Chapter 5 and Chap. 7

Depending on the detail of explanation, the first course could be for undergraduate students. The second course may attract graduate students.

Extensions in the Internet

There is an accompanying Internet page:

www.compfin.de

This is intended to serve the needs of the computational finance community and provides complementary material to this book. In particular, the collection *Topics for Computational Finance*, which is under construction, presents several of our findings or figures that would go beyond the limited scope of a textbook. In its final state, *Topics* is anticipated as a companion source to the *Tools*.

Acknowledgments

It is a pleasure to acknowledge helpful discussions with many people including Rainer Int-Veen, Karl Riedel, Sebastian Quecke, and Christian Jonen from the *QuantsCologne* research group. Many talented students have given feedback and challenged the search for better explanations and faster algorithms. Their enthusiasm has greatly inspired the project. Finally, the author would like to give special thanks in particular to Pascal Heider and Roland Seydel for their valuable contributions.

Köln, Germany Rüdiger U. Seydel
December 2011

Contents

Notations

Elements of Options

t	Time
T	Maturity date, time to expiration
S	Price of underlying asset
S_j, S_{ji}	Specific values of the price S
S_t	Price of the asset at time t
K	Strike price, exercise price
Ψ	Payoff function
V	Value of an option (V_{C} value of a call, V_{P} value of a put $^{\mathrm{Am}}$ American, $^{\mathrm{Eur}}$ European)
σ	Volatility
r	Interest rate (Appendix A.1)

General Mathematical Symbols

$\mathbb{R}$	Set of real numbers
$\mathbb{N}$	Set of integers > 0
$\in$	Element in
$\subseteq$	Subset of, $\subset$ strict subset
$[a, b]$	Closed interval $\{x \in \mathbb{R} : a \leq x \leq b\}$
$[a, b)$	Half-open interval $a \leq x < b$ (analogously $(a, b]$, (a, b))
P	Probability
E	Expectation (Appendices B.1, B.2)
Var	Variance
Cov	Covariance
log	Natural logarithm
$:=$	Defined to be

$\doteq$	Equal except for rounding errors
$\equiv$	Identical
$\Longrightarrow$	Implication
$\Longleftrightarrow$	Equivalence
$O(h^k)$	Landau symbol: for $h \to 0$ $f(h) = O(h^k) \iff \frac{f(h)}{h^k}$ is bounded
$\sim \mathcal{N}(\mu, \sigma^2)$	Normal distributed with expectation μ and variance σ^2
$\sim \mathcal{U}[0, 1]$	Uniformly distributed on $[0, 1]$
Δt	Small increment in t
^{tr}	transposed (A^{tr} is the matrix where the rows and columns of A are exchanged.)
$\mathcal{D}$	Set in $\mathbb{R}^n$ or in the complex plane, $\bar{\mathcal{D}}$ closure of $\mathcal{D}$, $\mathcal{D}^\circ$ interior of $\mathcal{D}$
$\partial\mathcal{D}$	Boundary of $\mathcal{D}$
$[0, 1]^2$	Unit square
$\mathcal{C}^0[a, b]$	Set of functions that are continuous on $[a, b]$
$\in \mathcal{C}^k[a, b]$	k-times continuously differentiable
$\mathcal{C}^{2,1}$	Set of functions of two arguments, twice differentiable w.r.t. to the first argument, and differentiable w.r.t. to the second argument
$\mathcal{L}^2$	Set of square-integrable functions
$\mathcal{H}$	Hilbert space, Sobolev space (Appendix C.3)
Ω	Sample space (in Appendix B.1)
$f^+ := \max\{f, 0\}$	
d	Symbol for differentiation
$\dot{u}$	Time derivative $\frac{\mathrm{d}u}{\mathrm{d}t}$ of a function $u(t)$
f'	Derivative of a function f
i	Symbol for imaginary unit
e	Symbol for the basis of the exponential function exp
∂	Symbol for partial differentiation
$\mathbf{1}_{\mathcal{M}}$	$= 1$ on a set $\mathcal{M}$, $= 0$ elsewhere (indicator function)
$\vert$	"such that" in the set-builder notation $\{ \quad \vert \quad \}$ and in conditional expectation
#	Number of

Integers

$i, j, k, l, m, n, M, N, \nu$

Various Variables

$X_t, X, X(t)$	Random variable
W_t	Wiener process, Brownian motion (Definition 1.7)
$y(x, \tau)$	Solution of a partial differential equation for (x, τ)
w	Approximation of y
h	Discretization grid size
φ	Basis function (Chap. 5)
ψ	Test function (Chap. 5)

Abbreviations

BDF	Backward difference formula (see Sect. 4.2.1)
CIR	Cox–Ingersoll–Ross model (see Sect. 1.7.4)
CFL	Courant–Friedrichs–Lewy (see Sect. 6.5.1)
Dow	Dow Jones Industrial Average
FE	Finite element
FFT	Fast Fourier transform
FTBS	Forward time backward space (see Sect. 6.5.1)
FTCS	Forward time centered space (see Sect. 6.4.2)
GBM	Geometric Brownian motion (see (1.47))
LCP	Linear complementarity problem
MC	Monte Carlo
ODE	Ordinary differential equation
OTC	Over the counter
OU	Ornstein–Uhlenbeck
PDE	Partial differential equation
PIDE	Partial integro-differential equation
PSOR	Projected successive overrelaxation
QMC	Quasi-Monte Carlo
SDE	Stochastic differential equation
SOR	Successive overrelaxation
TVD	Total variation diminishing
i.i.d.	Independent and identically distributed
inf	Infimum, largest lower bound of a set of numbers
sup	Supremum, least upper bound of a set of numbers
supp(f)	Support of a function f: $\{x \in \mathcal{D} : f(x) \neq 0\}$
t.h.o.	Terms of higher order
w.r.t.	With respect to

Hints on the Organization

(2.6) Number of Eq. (2.6)
(The first digit in all numberings refers to the chapter.)
(A.10) Equation in Appendix A, similarly B, C, D
⟶ hint (for instance, to an exercise)

Chapter 1
Modeling Tools for Financial Options

1.1 Options

What do we mean by option? An option is the right (but not the obligation) to buy or sell one unit of a risky asset at a prespecified fixed price within a specified period. An option is a financial instrument that allows—amongst other things—to make a bet on rising or falling values of an underlying asset. The **underlying** asset typically is a stock, or a parcel of shares of a company. Other examples of underlyings include stock indices (as the Dow Jones Industrial Average), currencies, or commodities. Since the value of an option depends on the value of the underlying asset, options and other related financial instruments are called *derivatives* (⟶ Appendix A.2). An option is a contract between two parties about trading the asset at a certain future time. One party is the *writer*, often a bank, who fixes the terms of the option contract and sells the option. The other party is the *holder*, who purchases the option, paying the market price, which is called *premium*. How to calculate a fair value of the premium is a central theme of this book. The holder of the option must decide what to do with the rights the option contract grants. The decision will depend on the market situation, and on the type of option. There are numerous different types of options, which are not all of interest to this book. In Chap. 1 we concentrate on standard options, also known as *vanilla options*. This Sect. 1.1 introduces important terms.

Options have a limited life time. The *maturity date* T fixes the time horizon. At this date the rights of the holder expire, and for later times ($t > T$) the option is worthless. There are two basic types of option: The **call** option gives the holder the right to *buy* the underlying for an agreed price K by the date T. The **put** option gives the holder the right to *sell* the underlying for the price K by the date T. The previously agreed price K of the contract is called **strike** or **exercise price**.[1] It is

[1]The price K as well as other prices are meant as the price of one unit of an asset, say, in \$.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_1

important to note that the holder is not obligated to *exercise*—that is, to buy or sell the underlying according to the terms of the contract. The holder may wish to close his position by selling the option. In summary, at time t the holder of the option can choose to

- sell the option at its current market price on some options exchange (at $t < T$),
- retain the option and do nothing,
- exercise the option ($t \leq T$), or
- let the option expire worthless ($t \geq T$).

In contrast, the writer of the option has the obligation to deliver or buy the underlying for the strike price K, in case the holder chooses to exercise. The risk situation of the writer differs strongly from that of the holder. The writer receives the premium when he issues the option and somebody buys it. This up-front premium payment compensates for the writer's potential liabilities in the future. The asymmetry between writing and owning options is evident. This book mostly takes the standpoint of the holder (long position in the option).

Not every option can be exercised at any time $t \leq T$. For **European options**, exercise is only permitted at expiration T. **American options** can be exercised at any time up to and including the expiration date. For options the labels American or European have no geographical meaning; both types are traded in each continent. Options on stocks are mostly American style.

The value of the option will be denoted by V. The value V depends on the price per share of the underlying, which is denoted S. This letter S symbolizes stocks, which are the most prominent examples of underlying assets. The variation of the asset price S with time t is expressed by S_t or $S(t)$. The value of the option also depends on the remaining time to expiry $T - t$. That is, V depends on time t. The dependence of V on S and t is written $V(S, t)$. As we shall see later, it is not easy to define and to calculate the fair value V of an option for $t < T$. But it is an easy task to determine the terminal value of V at expiration time $t = T$. In what follows, we shall discuss this topic, and start with European options as seen with the eyes of the holder.

1.1.1 The Payoff Function

At time $t = T$, the holder of a European call option will check the current price $S = S_T$ of the underlying asset. The holder has two alternatives to acquire the underlying asset: either buying the asset on the spot market (costs S), or buying the asset by exercising the call option (costs K). For a rational investor, the decision is easy: the costs are to be minimal. The holder will exercise the call if and only if $S > K$. For then the holder can immediately sell the asset for the spot price S and makes a gain of $S - K$ per share. In this situation the value of the option is $V = S - K$. (This reasoning ignores transaction costs.) In case $S < K$ the holder will not exercise, since then the asset can be purchased on the market for the cheaper

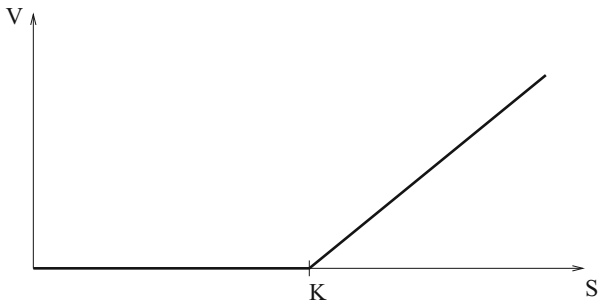

Fig. 1.1 Intrinsic value $V(S,T)$ of a call with exercise price K (payoff function)

price S. In this case the option is worthless, $V = 0$. In summary, the value $V(S,T)$ of a call option at expiration date T is given by

$$V(S_T,T) := \begin{cases} 0 & \text{in case } S_T \le K \quad \text{(option expires worthless)} \\ S_T - K & \text{in case } S_T > K \quad \text{(option is exercised).} \end{cases}$$

Hence

$$V(S_T,T) = \max\{S_T - K,\ 0\}.$$

Considered for all possible prices S_t, $\max\{S_t - K, 0\}$ is a function of S_t, in general for $0 \le t \le T$.[2] This **payoff function** is shown in Fig. 1.1. Using the notation $f^+ := \max\{f,\ 0\}$, this payoff can be written in the compact form $(S_t - K)^+$. Accordingly, the value $V(S_T,T)$ of a call at maturity date T is

$$V(S_T,T) = (S_T - K)^+. \tag{1.1}$$

For a European put, exercising only makes sense in case $S < K$. The payoff $V(S,T)$ of a put at expiration time T is

$$V(S_T,T) := \begin{cases} K - S_T & \text{in case } S_T < K \quad \text{(option is exercised)} \\ 0 & \text{in case } S_T \ge K \quad \text{(option is worthless).} \end{cases}$$

Hence

$$V(S_T,T) = \max\{K - S_T,\ 0\},$$

[2] In this chapter, the payoff evaluated at t only depends on the current value S_t. Payoffs that depend on the *entire path* S_t for all $0 \le t \le T$ occur for exotic options, see Chap. 6.

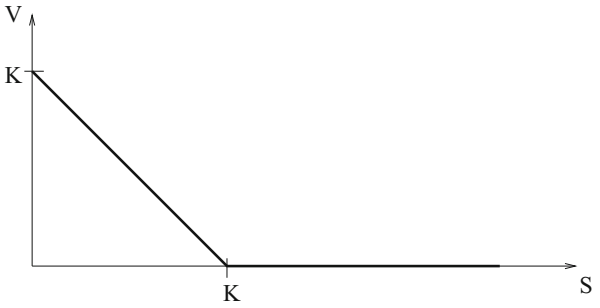

Fig. 1.2 Intrinsic value $V(S,T)$ of a put with exercise price K (payoff function)

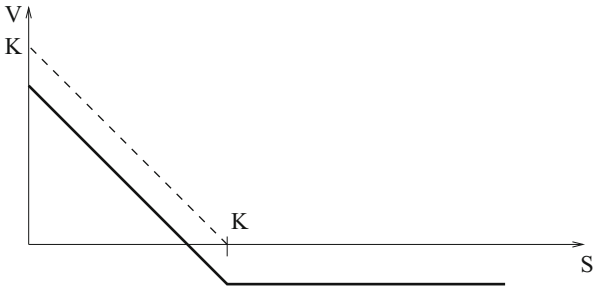

Fig. 1.3 Profit diagram of a put

or

$$V(S_T,T) = (K - S_T)^+ , \tag{1.2}$$

compare Fig. 1.2.

We shall denote both payoff functions in (1.1) and (1.2) by $\Psi(S)$,

$$\Psi(S) := \begin{cases} (S-K)^+ & \text{in case of a call,} \\ (K-S)^+ & \text{in case of a put.} \end{cases} \tag{1.3}$$

The curves in the payoff diagrams of Figs. 1.1 and 1.2 show the option values from the perspective of the holder. The profit is not shown. For an illustration of the profit, the initial costs for buying the option at $t = t_0$ must be subtracted. The initial costs basically consist of the premium and the transaction costs. Since both are paid upfront, they are multiplied by $e^{r(T-t_0)}$ to take account of the time value; r is the continuously compounded interest rate. Subtracting the costs leads to shifting down the curves in Figs. 1.1 and 1.2. The resulting *profit diagram* shows a negative profit for some range of S-values, which of course means a loss (see Fig. 1.3).

The payoff function for an American call is $(S_t - K)^+$ and for an American put $(K - S_t)^+$ for any $t \le T$. The Figs. 1.1 and 1.2 as well as the Eqs. (1.1), (1.2) remain valid for American type options.

The payoff diagrams of Figs. 1.1 and 1.2 and the corresponding profit diagrams show that a potential loss for the purchaser of an option (long position) is limited by the initial costs, no matter how bad things get. The situation for the writer (short position) is reverse. For him the payoff curves of Figs. 1.1 and 1.2 as well as the profit curves must be reflected on the S-axis. The writer's profit or loss is the reverse of that of the holder. Multiplying the payoff of a call in Fig. 1.1 by (-1) illustrates the potentially unlimited risk of a short call. Hence the writer of a call must carefully design a strategy to compensate for his risks. We will come back to this issue in Sect. 1.5.

1.1.2 A Priori Bounds

No matter what the terms of a specific option are and no matter how the market behaves, the values V of the options satisfy certain bounds. These bounds are known a priori. For example, the value $V(S,t)$ of an American option can never fall below the payoff, for all S and all t. These bounds follow from the *no-arbitrage principle* (⟶ Appendices A.2 and A.3).

To illustrate the strength of no-arbitrage arguments, we assume for an American put that its value $V_{\mathrm{P}}^{\mathrm{Am}}$ is below the payoff. $V < 0$ contradicts the definition of the option. Hence $V \geq 0$, and S and V would be in the triangle seen in Fig. 1.2. That is, $S < K$ and $0 \leq V < K - S$. This scenario would allow an arbitrage strategy as follows: Borrow the cash amount of $S + V$, and buy both the underlying and the put. Then immediately exercise the put, selling the underlying for the strike price K. The profit of this arbitrage strategy is $K - S - V > 0$. This is in conflict with the no-arbitrage principle. Hence the assumption that the value of an American put is below the payoff must be wrong. We conclude for the put

$$V_{\mathrm{P}}^{\mathrm{Am}}(S,t) \geq (K-S)^+ \quad \text{for all } S, t\,.$$

Similarly, for the call

$$V_{\mathrm{C}}^{\mathrm{Am}}(S,t) \geq (S-K)^+ \quad \text{for all } S, t\,.$$

(The meaning of the notations $V_{\mathrm{C}}^{\mathrm{Am}}$, $V_{\mathrm{P}}^{\mathrm{Am}}$, $V_{\mathrm{C}}^{\mathrm{Eur}}$, $V_{\mathrm{P}}^{\mathrm{Eur}}$ is evident.)

Other bounds are listed in Appendix E.1. For example, a European put on an asset that pays no dividends until T may also take values below the payoff, but is always above the lower bound $K\mathrm{e}^{-r(T-t)} - S$. The value of an American option should never be smaller than that of a European option because the American type includes the European type exercise at $t = T$ and in addition *early exercise* for $t < T$. That is

$$V^{\mathrm{Am}} \geq V^{\mathrm{Eur}}$$

as long as all other terms of the contract are identical. In summary, with the notation Ψ of (1.3), the inequality

$$V^{\mathrm{Am}}(S,t) \geq \max\{\, V^{\mathrm{Eur}}(S,t),\ \Psi(S)\,\} \tag{1.4}$$

holds for American options. For European options, when no dividends are paid until T, the values of put and call are related by the *put-call parity*

$$S + V_{\mathrm{P}}^{\mathrm{Eur}} - V_{\mathrm{C}}^{\mathrm{Eur}} = K\mathrm{e}^{-r(T-t)}\,,$$

which can be shown by applying arguments of arbitrage (⟶ Exercise 1.1).

1.1.3 Options in the Market

The features of the options imply that an investor purchases puts when the price of the underlying is expected to fall, and buys calls when the prices are about to rise. This mechanism inspires speculators. An important application of options is hedging (⟶ Appendix A.2).

The value of $V(S,t)$ also depends on other factors. Dependence on the strike K and the maturity T is evident. Market parameters affecting the price are the interest rate r, the **volatility** σ of the price S_t, and dividends in case of a dividend-paying asset. The interest rate r is the risk-free rate, which applies to zero bonds or to other investments that are considered free of risks (⟶ Appendices A.1 and A.2). The important volatility parameter σ can be defined as standard deviation of the fluctuations in S_t, for scaling divided by the square root of the observed time period. The larger the fluctuations, represented by large values of σ, the harder is to predict a future value of the asset. Hence the volatility may serve as a measure of risk. The dependence of V on σ is highly sensitive. On occasion we write $V(S,t;\,T,K,r,\sigma)$ when the focus is on the dependence of V on market parameters.

Time is measured in years. The units of r and σ^2 are per year. Writing $\sigma = 0.2$ means a volatility of 20%, and $r = 0.05$ represents an interest rate of 5%. Table 1.1

Table 1.1 List of important variables

t	Current time, $0 \leq t \leq T$
T	Expiration time, maturity
r	Interest rate, return ($r \geq 0$ assumed)
S, $S(t)$, S_t	Current price of underlying asset
σ	Annual volatility of S
K	Strike price, exercise price per share
$V(S,t)$	Value of option
$\Psi(S)$	Payoff of a standard option
δ	Dividend rate

summarizes the key notations of option pricing. The notation is standard except for the strike price K, which is sometimes denoted X, or E.

The time period of interest is $t_0 \le t \le T$. One might think of t_0 denoting the date when the option is issued and t as a symbol for "today." But this book mostly sets $t_0 = 0$ in the role of "today," without loss of generality. Then the interval $0 \le t \le T$ represents the remaining life time of the option. The price S_t is a stochastic process, compare Sect. 1.6. In real markets, the interest rate r and the volatility σ vary with time. To keep the models and the analysis simple, we mostly assume r and σ to be constant on $0 \le t \le T$. Further we suppose that all variables are arbitrarily divisible and consequently can vary continuously—that is, all variables vary in the set $\mathbb{R}$ of real numbers.

1.1.4 The Geometry of Options

As mentioned, our aim is to calculate $V(S,t)$ for fixed values of K, T, r, σ. The values $V(S,t)$ can be interpreted as a surface over the subset

$$S > 0\ ,\ 0 \le t \le T$$

of the (S,t)-plane. Figure 1.4 illustrates the character of such a surface for the case of an American put. For the illustration assume $T = 1$. The figure depicts six curves obtained by cutting the *option surface* with the planes $t = 0,\ 0.2, \ldots, 1.0$. For $t = T$ the payoff function $(K - S)^+$ of Fig. 1.2 is clearly visible.

Shifting this payoff curve parallel for all $0 \le t < T$ creates another surface, which consists of the two planar pieces $V = 0$ (for $S \ge K$) and $V = K - S$ (for

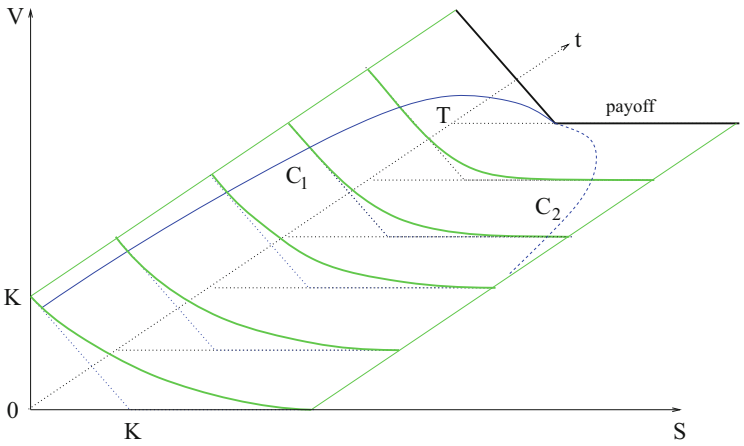

Fig. 1.4 Value $V(S,t)$ of an American put (schematically)

$S < K$). By (1.4), this *payoff surface* $(K-S)^+$ is a lower bound to the option surface, $V(S,t) \geq (K-S)^+$. Figure 1.4 shows two curves C_1 and C_2 on the option surface. The curve C_1 is the *early-exercise curve*, because on the planar part with $V(S,t) = K - S$ holding the option is not optimal. (This will be explained in Sect. 4.5.) The curve C_2 has a technical meaning explained below. Within the area limited by these two curves C_1, C_2, the option surface is clearly above the payoff surface, $V(S,t) > (K-S)^+$. Outside that area, both surfaces coincide. This is strict "above" C_1, where $V(S,t) = K - S$, and holds approximately for S beyond C_2, where $V(S,t) \approx 0$ or $V(S,t) < \varepsilon$ for a small value of $\varepsilon > 0$. The location of C_1 and C_2 is not known, these curves are calculated along with the calculation of $V(S,t)$. Of special interest is $V(S,0)$, the value of the option "today." This curve is seen in Fig. 1.4 for $t = 0$ as the front edge of the option surface. This front curve may be seen as smoothing the corner in the payoff function. The schematic illustration of Fig. 1.4 is completed by a concrete example of a calculated put surface in Fig. 1.5. An approximation of the curve C_1 is shown.

The above was explained for an American put. For other options the bounds are different ($\longrightarrow$ Appendix E.1). As mentioned before, a European put takes values above the lower bound $Ke^{-r(T-t)} - S$, compare Fig. 1.6 and Exercise 1.1b.

In summary, this Sect. 1.1 has introduced an option with the following features: it depends on *one* underlying, and its payoff is $(K-S)^+$ or $(S-K)^+$, with S evaluated at the current time instant. This is the standard option called vanilla option. All other options are called *exotic*. To clarify the distinction between vanilla options and exotic options, we hint at ways how an option can be "exotic." For example, an option may depend on a basket of several underlying assets, or the payoff may be different, or the option may be *path-dependent* in that V no longer depends solely on the current (S_t, t) but on the entire path S_t for $0 \leq t \leq T$. To give an example of the latter, we mention an *Asian option*, where the payoff depends on the average

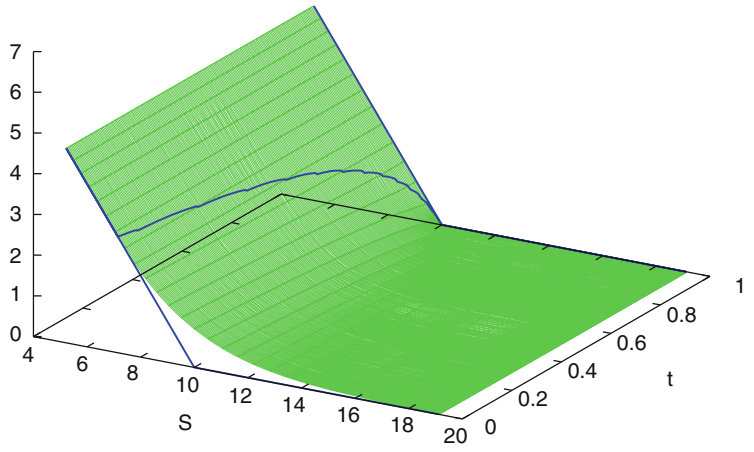

Fig. 1.5 Value $V(S,t)$ of an American put with $r = 0.06$, $\sigma = 0.30$, $K = 10$, $T = 1$

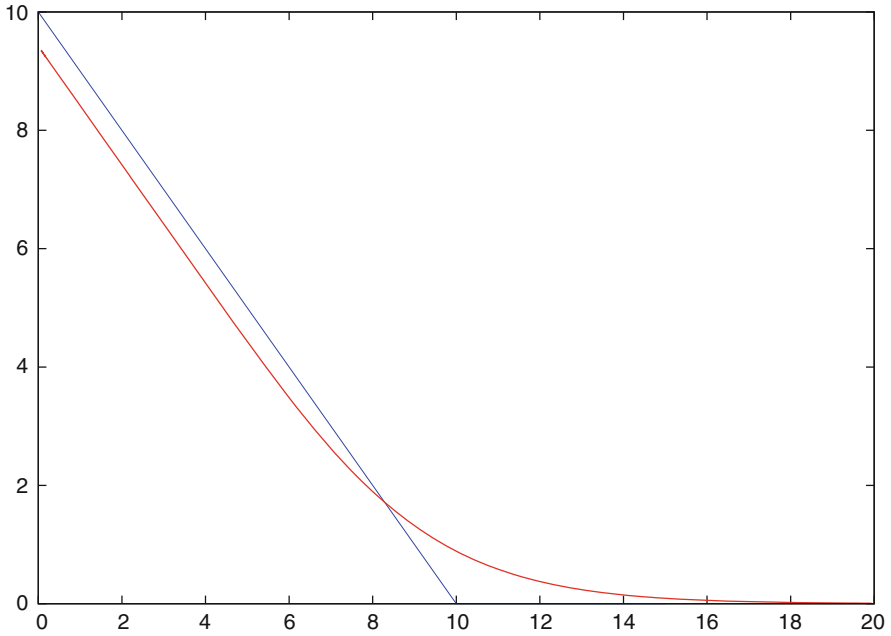

Fig. 1.6 Value $V(S, 0)$ of a European put (in *red*) for $t = 0$ and $0 \leq S \leq 20$. Parameters are $T = 1$, $K = 10$, $r = 0.06$, $\sigma = 0.3$. The payoff $\Psi(S) = V(S, T)$ is drawn with a *blue line*. For small values of S the value V approaches its lower bound, here $9.4 - S$

value of the asset for all times until expiry. Or for a *barrier option* the value also depends on whether the price S_t hits a prescribed barrier during its life time. We come back to exotic options later in the book.

1.2 Model of the Financial Market

Ultimately it is the market that decides on the value of an option. Above, we have been anticipating "surfaces" $V(S, t)$, pretending a value V for any S, t. In the reality of markets, prices V^{mar} of options are only known for a selection of discrete values of the underlying's prices, times, or parameters. Geometrically, the available data form only relatively few points on the anticipated "surfaces" V. If we try to *calculate* a reasonable value of the option, we need a mathematical model of the market. Mathematical models can serve as approximations and idealizations of the complex reality of the financial world. The most prominent example of a model is the model named after the pioneers Black, Merton and Scholes. Their approaches have been both successful and widely accepted. This Sect. 1.2 introduces some key elements of market models. Based on a chosen mathematical model, the value and the potential

of an option is assessed. This includes both the calculation of $V(S,t)$, and an analysis of how sensitive V reacts on changes in S, t, or on variations in the parameters. Of course, the results are subject to the uncertainty of the model.

It is attractive to define the option surfaces $V(S,t)$ on the *half strip* $S > 0$, $0 \le t \le T$ as solutions of suitable equations. Then calculating V amounts to solving the equations on the domain "half strip." In fact, a series of assumptions allows to characterize *value functions* $V(S,t)$ as solutions of certain partial differential equations or partial differential inequalities. The model of Black, Merton and Scholes is represented by the famous Black–Scholes equation, which was suggested in 1973.

Definition 1.1 (Black–Scholes Equation)

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0 \tag{1.5}$$

Equation (1.5) is a partial differential equation (PDE) for the value function $V(S,t)$ of options. This equation may serve as symbol of the classic market model. But what are the assumptions leading to the Black–Scholes equation?

Assumptions 1.2 (Black–Merton–Scholes Model of the Market)

(a) There are no arbitrage opportunities.

(b) The market is frictionless.

This means that there are no transaction costs (fees or taxes), the interest rates for borrowing and lending money are equal, all parties have immediate access to any information, and all securities and credits are available at any time and in any size.[3] *Consequently, all variables are perfectly divisible—that is, can take any real number. Further, individual trading will not influence the price.*

(c) The asset price follows a geometric Brownian motion.

(This stochastic motion will be discussed in Sects. 1.6–1.8.)

(d) r and σ are constant for $0 \le t \le T$. No dividends are paid in that time period. The option is European.

These are the assumptions that lead to the Black–Scholes equation (1.5). The assumptions are rather strong, in particular, the volatility σ being constant. In contrast to the complications of real markets, the Black–Scholes model is a simple model, but most helpful. Some of the assumptions can be weakened. We come to more complex models later in the text. For brevity, we call the restricted model represented by Assumptions 1.2 Black–Scholes model, because Merton has also extended it to include jumps, which are ruled out by (c). A derivation of the Black–Scholes partial differential equation (1.5) is given in Appendix A.4. Admitting all real numbers t within the interval $0 \le t \le T$ leads to characterize the model as *continuous-time model*. In view of allowing also arbitrary values of $S > 0$, $V > 0$, we speak of a continuous model.

[3] In particular, this holds for trading the underlying.

A value function $V(S,t)$ is not fully defined by merely requesting that it solves (1.5) for all S and t out of the domain "half strip." In addition to solving this PDE, the function $V(S,t)$ must satisfy a **terminal condition**. The terminal condition for $t = T$ is

$$V(S,T) = \Psi(S)\,,$$

where Ψ denotes the payoff function (1.3), depending on the type of option. This terminal condition is no artificial requirement. It is a prime statement and naturally represents the definition of an option. In theory, (1.5) with $V(S,T) = \Psi(S)$ is a Cauchy problem and completes one possibility of defining a value function $V(S,t)$.

For computational purposes, the full half strip with $S > 0$ is typically truncated, say, to $S_{\min} \le S \le S_{\max}$. Then **boundary conditions** for $S_{\min}$ and $S_{\max}$ are needed in addition. Sometimes they are given by the financial terms of the option, for example, for barrier options. But often boundary conditions are secondary and artificial, and not immediately provided by the financial construction. Rather, boundary conditions are required to make a solution of the partial differential equation meaningful. In Chap. 4 we will come back to the Black–Scholes equation and to boundary conditions.

For (1.5) an analytic solution is known (Eqs. (1.7)–(1.10), and Appendix A.4). Note that the partial differential equation (1.5) is linear in the value function V.[4] The partial differential equation is no longer linear when Assumptions 1.2(b) are relaxed. For example, for considering trading intervals Δt and transaction costs as k per unit, one could add the nonlinear term

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

to (1.5), see [376], and Sect. 7.1. Also finite liquidity (feedback of trading to the price of the underlying) leads to nonlinear terms in the PDE. In the general case, closed-form solutions do not exist, and a solution is calculated numerically, especially for American options. For the American-style option a further nonlinearity stems from the early-exercise feature (⟶ Chap. 4). For solving (1.5) numerically, a variant with dimensionless variables can be used (⟶ Exercise 1.4).

Of course, the calculated value V of an option depends on the chosen market model. Writing $V(S,t;\,T,K,r,\sigma)$ suggests a focus on the Black–Scholes equation. This could be made definite by writing V^{BS}, for example. Other market models may involve more parameters. Then, in general, the corresponding value of the value function V is different from V^{BS}. Since we mostly stick to the market model of Assumptions 1.2, we drop the superscript. All our prices V are model prices, not market prices. For the relation between our model prices V and market prices V^{mar}, see Sect. 1.10.

[4] The function V is not linear in S or t. Also the payoff is nonlinear; the vanilla functions $\Psi(S) = (K-S)^+$ and $\Psi(S) = (S-K)^+$ are convex.

Based on the chosen mathematical model, a **sensitivity analysis** is possible. We ask, for example, how does the price V change to a value $V+\mathrm{d}V$, when the price S of the underlying changes to $S+\mathrm{d}S$? Similarly, what is the effect of a change $\mathrm{d}\sigma$ in the parameter σ? When the value function $V(S,t;\dots)$ is smooth, the Taylor expansion

$$\mathrm{d}S = \frac{\partial V}{\partial S}\mathrm{d}S + \frac{\partial V}{\partial t}\mathrm{d}t + \frac{\partial V}{\partial \sigma}\mathrm{d}\sigma + \frac{\partial V}{\partial r}\mathrm{d}r + \frac{1}{2}\frac{\partial^2 V}{\partial S^2}(\mathrm{d}S)^2 + \dots . \tag{1.6}$$

suggest an answer. The proper partial derivative of V is an amplification factor. For small enough $\mathrm{d}t$ it provides a first-order guess on how sensitive V may react to changes in the corresponding variable or parameter. In the finance context, these partial derivatives of V are called "greeks." For example, "delta" is the name for

$$\Delta := \frac{\partial V}{\partial S}\,.$$

The second-order derivative "gamma" $\frac{\partial^2 V}{\partial S^2}$ is important too, and is included in the list of first-order terms in (1.6) by reasons that will become clear in Sects. 1.6 and 1.8. Several of these *sensitivity parameters* or *hedge parameters* need to be approximated as well.

At this point, a word on the notation is appropriate. The symbol S for the asset price is used in different roles: First it comes without subscript in the role of an independent real variable $S > 0$ on which the value function $V(S,t)$ depends, say as solution of the partial differential equation (1.5). Second it is used as S_t with subscript t to emphasize its random character as stochastic process. When the subscript t is omitted, the current role of S becomes clear from the context.

1.3 Numerical Methods

Applying numerical methods is indispensable in all fields of technology, including financial engineering. Often this is not apparent. In particular, this holds in case analytic formulas are available. The most important formula in our context is the Black–Scholes formula. Essentially, the standard normal cumulative distribution function F (with density f, compare Exercise 1.5 or Appendix E.2) is evaluated for the following arguments:

$$d_1 := \frac{\log\frac{S}{K} + \left(r - \delta + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}\,, \tag{1.7}$$

$$d_2 := d_1 - \sigma\sqrt{T-t} = \frac{\log\frac{S}{K} + \left(r - \delta - \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}\,. \tag{1.8}$$

With these definitions of d_1 and d_2, the values $V_C(S,t)$ and $V_P(S,t)$ of a European call and put are given by

$$V_C(S,t) = Se^{-\delta(T-t)}F(d_1) - Ke^{-r(T-t)}F(d_2)\,, \tag{1.9}$$

$$V_P(S,t) = -Se^{-\delta(T-t)}F(-d_1) + Ke^{-r(T-t)}F(d_2)\,. \tag{1.10}$$

1.3.1 Algorithms

For the above Black–Scholes formula, the important role of numerical algorithms is often not noticed. The closed-form expressions (1.7)–(1.10) might suggest that no numerical procedure is needed. But such expressions may include evaluating the logarithm or the computation of the distribution function F of the normal distribution. Such elementary tasks are performed using sophisticated numerical algorithms. In pocket calculators one merely presses a button without being aware of the numerics. The robustness of those elementary numerical methods is so reliable and the efficiency so high that underlying algorithms almost appear not to exist. But even for apparently simple tasks the methods are quite demanding (⟶ Exercise 1.5). The methods must be carefully designed because inadequate strategies might produce inaccurate results (⟶ Exercise 1.6). Dangerous subtractions tend to be overlooked. For example, for options deep out of the money, the value $V \approx 0$ is obtained by subtraction of two numbers basically of the same size. Then accuracy is lost; an evaluation of the Black–Scholes formula suffers from cancellation.[5]

Spoiled by generally available black-box software and graphics packages we take the support and the success of numerical workhorses for granted. We make use of the numerical tools with great respect but without further comments, and we just assume an education in elementary numerical methods. An introduction to important methods and hints on the literature are given in Appendix C.1.

Since financial markets undergo apparently stochastic fluctuations, stochastic approaches provide natural tools to simulate prices. These methods are based on formulating and simulating stochastic differential equations. This leads to Monte Carlo methods (⟶ Chap. 3). In computers, related simulations of options are performed in a deterministic manner. It will be decisive how to simulate randomness (⟶ Chap. 2). Chapters 2 and 3 are devoted to tools for simulation. These methods can be applied easily even in case the Assumptions 1.2 are not satisfied.

More efficient methods will be preferred provided their use can be justified by the validity of the underlying models. For example it may be advisable to solve the partial differential equations of the Black–Scholes type. Then one has to choose among several methods. The most elementary ones are finite-difference methods (⟶ Chap. 4). A somewhat higher flexibility concerning error control is possible

[5] But for out-of-the-money options the Black–Scholes model is not recommended anyhow.

with finite-element methods (⟶ Chap. 5). The numerical treatment of exotic options requires a more careful consideration of stability issues (⟶ Chap. 6). Methods based on differential equations will be described in the larger part of this book. And beyond Black and Scholes, even more tools are needed (⟶ Chap. 7).

1.3.2 Discretization

The mathematical formulation benefits from the assumption that all variables take values in the continuum $\mathbb{R}$. This idealization is practical since it avoids initial restrictions of technical nature, and it gives us freedom to impose *artificial* discretizations convenient for the numerical methods. The hypothesis of a continuum applies to the (S,t)-domain of the half strip $0 \le t \le T$, $S > 0$, and to the differential equations. In contrast to the hypothesis of a continuum, the financial reality is rather discrete: Neither the price S nor the trading times t can take any real value. The artificial discretization introduced by numerical methods is at least twofold:

1.) The (S,t)-domain is replaced by a **grid** of a finite number of (S,t)-points, as illustrated in Fig. 1.7.
2.) Differential equations are adapted to the grid and replaced by a finite number of algebraic equations.

The restriction of the differential equations to the grid causes **discretization errors**. The errors depend on the coarseness of the grid. In Fig. 1.7, the distance

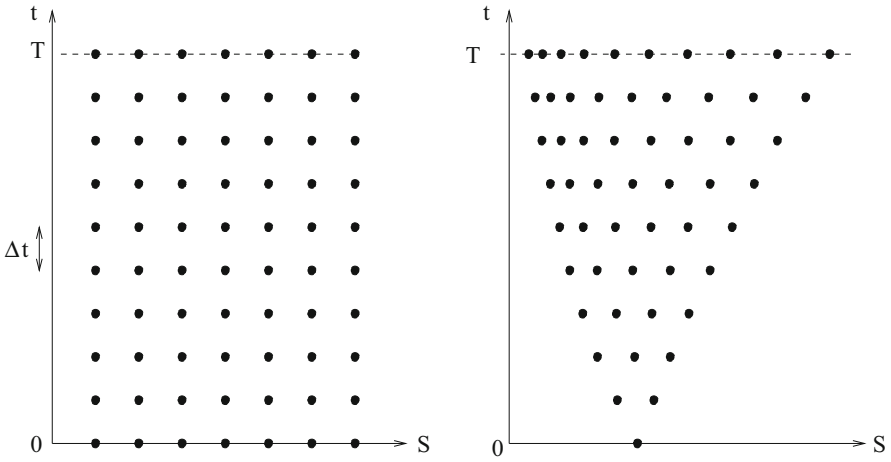

Fig. 1.7 Grid points in the (S,t)-domain

between two consecutive t-values of the grid is denoted Δt.[6] So the errors will depend on Δt and on ΔS. It is one of the aims of numerical algorithms to control the errors. The left-hand figure in Fig. 1.7 shows a simple rectangular grid, whereas the right-hand figure shows a tree-type grid as used in Sect. 1.4. The type of the grid matches the kind of underlying equations. The values of $V(S,t)$ are primarily approximated at the grid points. Intermediate values can be obtained by interpolation.

The continuous model is an idealization of the discrete reality. But the numerical discretization does not reproduce the original discretization. For example, it would be a rare coincidence when Δt represents a day. The derivations that go along with the twofold transition

$$\text{discrete} \longrightarrow \text{continuous} \longrightarrow \text{discrete}$$

do not compensate.

Another kind of discretization is that computers replace the real numbers by a finite number of rational numbers, namely, the floating-point numbers. The resulting rounding error will not be relevant for much of our analysis, except for investigations of stability.

1.3.3 Efficiency

The various methods are discussed in terms of accuracy and speed. Ultimately the methods must give quick and accurate answers to real-time problems posed in financial markets. Efficiency and reliability are key demands. Internally the numerical methods must deal with diverse problems such as convergence order or stability. So the numerical analyst is concerned in error estimates and error bounds. Technical criteria such as complexity or storage requirements are relevant for the implementation.

But it is not easy to judge efficiency. We sketch how to approach the problem. Denote the grid size by h (Δt in Fig. 1.7). Then the number of grid points in one discretized variable is of the order $O(\frac{1}{h})$. For n such variables ($n = 2$ in Fig. 1.7) one has $O(h^{-n})$ grid points, and we expect costs

$$\text{costs} = \alpha h^{-n} .$$

The performance is characterized by the achieved error, for example,

$$\text{error} = \beta h^{p} ,$$

[6] The symbol Δt denotes a small increment in t (analogously $\Delta S, \Delta W$). In case Δ would be a number, the product with u would be denoted $\Delta \cdot u$ or $u\Delta$.

with constants α, β, and an order p of the error. Eliminating the grid size h gives

$$\text{costs} \sim \text{error}^{-n/p} .$$

This is a potential law. Testing algorithms, measuring costs and errors, and entering the results in a double-logarithmic form allows to compare the efficiency of various algorithms. This will be done in Sect. 4.9. For the task of evaluating F, this can be found in Topic 10 of the *Topics for Compuational Finance* (Topics fCF). Notice that in general every accuracy level has a different answer to the question, which method is most efficient.

1.4 The Binomial Method

The major part of the book is devoted to continuous models and their discretizations. With much less effort a discrete approach provides us with a short way to establish a first algorithm for calculating options. The resulting *binomial method* is robust and widely applicable.

In practice one is often interested in the one value $V(S_0, 0)$ of an option at the current spot price S_0. Then it can be unnecessarily costly to calculate the surface $V(S, t)$ for the entire domain to extract the required information $V(S_0, 0)$. The relatively small task of calculating $V(S_0, 0)$ can be comfortably solved using the binomial method. This method is based on a tree-type grid applying appropriate binary rules at each grid point. The grid is not predefined but is constructed by the method. For illustration see the right-hand side in Fig. 1.7, and Fig. 1.10.

1.4.1 A Discrete Model

We begin with discretizing the continuous time t, replacing t by equidistant time instances t_i. Let us use the notations

M: number of time steps
$\Delta t := \frac{T}{M}$
$t_i := i \cdot \Delta t, \quad i = 0, \ldots, M .$
$S_i := S(t_i)$

So far the domain of the (S, t) half strip is *semidiscretized* in that it is replaced by parallel straight lines with distance Δt apart, leading to a discrete-time model. The next step of discretization replaces the continuous values S_i along the parallel $t = t_i$ by discrete values $S_{j,i}$, for all i and appropriate j. For a better understanding of the S-discretization compare Fig. 1.8. This figure shows a mesh of the grid, namely, the transition from t to $t + \Delta t$, or from t_i to t_{i+1}.

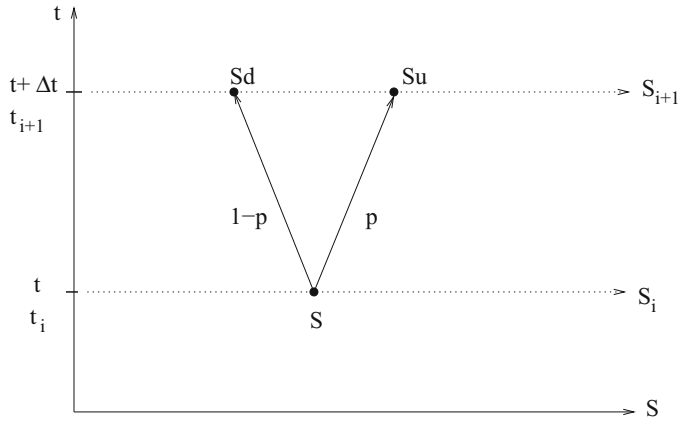

Fig. 1.8 The principle setup of the binomial method

Assumptions 1.3 (Binomial Method) *Assume (Bi1), (Bi2), and (Bi3).*

(Bi1)

The price S over each period of time Δt can only have two possible outcomes: An initial value S either evolves "up" to Su, or "down" to Sd, with $0 < d < u$.

(Bi2)

The probability of an up movement is p, $\mathsf{P}(\text{up}) = p$.

The factor u is the factor of an upward movement, and d is the factor of a downward movement. The rules (Bi1) and (Bi2) represent the framework of a binomial process. Such a process behaves like tossing a biased coin where the outcome "head" (up) occurs with probability p. At this stage of the modeling, the values of the three parameters u, d and p are undetermined. They are fixed in a way such that the model is consistent with the continuous model in case $\Delta t \to 0$. This aim leads to further assumptions. The basic idea of the approach is to equate the expectation and the variance of the discrete model with the corresponding values of the continuous model. This amounts to require

(Bi3)

Expectation and variance of S refer to their continuous counterparts, evaluated for the risk-free interest rate r.

This assumption (Bi3) leads to equations for the parameters u, d, p. The resulting probability P of (Bi2) does not reflect the expectations of an individual in the market. Rather P is an artificial risk-neutral probability that matches (Bi3).[7] The expectation E below in (1.11) refers to this probability; this is sometimes written

[7]To distinguish this specific "money market measure" P from other probabilities, one gives it a specific notation. In later sections we shall use the symbol Q.

E_P. (We shall return to the assumptions (Bi1), (Bi2), and (Bi3) in the subsequent Sect. 1.5.) Let us further assume that no dividend is paid within the time period of interest. This assumption simplifies the derivation of the method and can be removed later.

1.4.2 Derivation of Equations

Recall the definition of the expectation for the discrete case, Appendix B.1, Eq. (B.13), and conclude

$$\mathsf{E}(S_{i+1}) = pS_iu + (1-p)S_id\,.$$

Here S_i is an arbitrary value, which develops randomly to S_{i+1}, when t_i proceeds to t_{i+1}, following the assumptions (Bi1) and (Bi2). In this sense, E is a conditional expectation. As will be seen in Sect. 1.7.2, the expectation of the continuous model is

$$\mathsf{E}(S_{i+1}) = S_i\,\mathrm{e}^{r\Delta t}\,. \tag{1.11}$$

Equating gives

$$\mathrm{e}^{r\Delta t} = pu + (1-p)d\,. \tag{1.12}$$

This is the first of three equations required to fix u, d, p. Solved for the risk-neutral probability p we obtain

$$p = \frac{\mathrm{e}^{r\Delta t} - d}{u - d}\,. \tag{1.13}$$

To be a valid model of probability, $0 \le p \le 1$ must hold. Leaving aside degenerate cases, we require $0 < p < 1$. This is equivalent to

$$d < \mathrm{e}^{r\Delta t} < u\,. \tag{1.14}$$

These inequalities relate the upward and downward movements of the asset price to the riskless interest rate r. The inequalities (1.14) are no new assumptions but follow from the no-arbitrage principle.

Next we equate variances. Via the variance the volatility σ enters the model. From the continuous model we apply the relation

$$\mathsf{E}(S_{i+1}^2) = S_i^2\,\mathrm{e}^{(2r+\sigma^2)\Delta t}\,. \tag{1.15}$$

For the relations (1.11) and (1.15) we refer to Sect. 1.8 (⟶ Exercise 1.22). Recall that the variance satisfies $\mathsf{Var}(S) = \mathsf{E}(S^2) - (\mathsf{E}(S))^2$ (⟶ Appendix B.1). Equations (1.11) and (1.15) combine to

$$\mathsf{Var}(S_{i+1}) = S_i^2 \mathrm{e}^{2r\Delta t}(\mathrm{e}^{\sigma^2\Delta t} - 1)\,.$$

On the other hand the discrete model satisfies

$$\begin{aligned}\mathsf{Var}(S_{i+1}) &= \mathsf{E}(S_{i+1}^2) - (\mathsf{E}(S_{i+1}))^2 \\ &= p(S_i u)^2 + (1-p)(S_i d)^2 - S_i^2(pu + (1-p)d)^2\,.\end{aligned}$$

Equating variances of the continuous and the discrete model, and applying (1.12) leads to

$$\begin{aligned}\mathrm{e}^{2r\Delta t}(\mathrm{e}^{\sigma^2\Delta t} - 1) &= pu^2 + (1-p)d^2 - (\mathrm{e}^{r\Delta t})^2 \\ \mathrm{e}^{2r\Delta t + \sigma^2\Delta t} &= pu^2 + (1-p)d^2\,. \qquad (1.16)\end{aligned}$$

The Eqs. (1.12), (1.16) constitute two relations for the three unknowns u, d, p. We are free to impose an arbitrary third equation, and choose

$$u \cdot d = \gamma \qquad (1.17)$$

for a suitably chosen value of γ. One example is the plausible assumption $\gamma = 1$, or $ud = 1$, which reflects a symmetry between upward and downward movement of the asset price. Later, we shall use (1.17) with a specifically designed γ. With the third equation (1.17), the parameters u, d and p are fixed. First we discuss the classic choice $\gamma = 1$. Then the parameters depend on r, σ and Δt. So does the grid, which is analyzed next (Fig. 1.9).

The above rules are applied to each grid line $i = 0, \ldots, M$, starting at $t_0 = 0$ with the specific value $S = S_0$. Attaching meshes of the kind depicted in Fig. 1.8 for subsequent values of t_i builds a tree with node values $Su^j d^k$ and $j + k = i$.

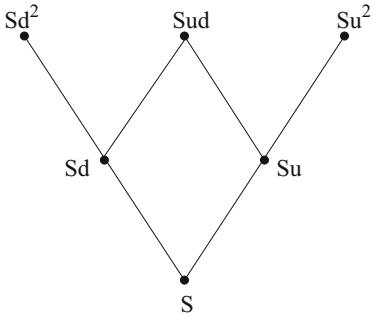

Fig. 1.9 Sequence of several meshes (schematically)

In this way, specific discrete values $S_{j,i}$ of S_i and the nodes of the tree are defined. Since the same constant factors u and d underlie all meshes and since $Sud = Sdu$ holds, after the time period $2\Delta t$ the asset price can only take three values rather than four: The tree is recombining. It does not matter which of the two possible paths we take to reach Sud. This property extends to more than two time periods. Consequently the binomial process defined by Assumptions 1.3 is *path independent*. Accordingly at expiration time $T = M\Delta t$ the price S can take only the $(M + 1)$ discrete values $S_{j,M} := Su^j d^{M-j}$, $j = 0, 1, \ldots, M$. The number of nodes in the tree grows quadratically in M. (Why?)

The symmetry of the choice $ud = 1$ becomes apparent in that after two time steps the asset value S repeats. (Compare also Fig. 1.10.) For $ud = 1$, the central line of the tree grows vertically. The vertical arrangement is advantageous for matching a tree to barriers. But to smooth the convergence, it may be advisable to bend the tree such that its central line ends up at the strike. (We return to such improvements below.) In a (t, S)-plane the tree can be interpreted as a grid of exponential-like curves. The binomial approach defined by (Bi1) with the proportionality between S_i and S_{i+1} reflects exponential growth or decay of S. Since the tree extends from $S_0 d^M$ to $S_0 u^M$, all grid points have the desirable property $S > 0$, but for large M the tree becomes unrealistically wide.

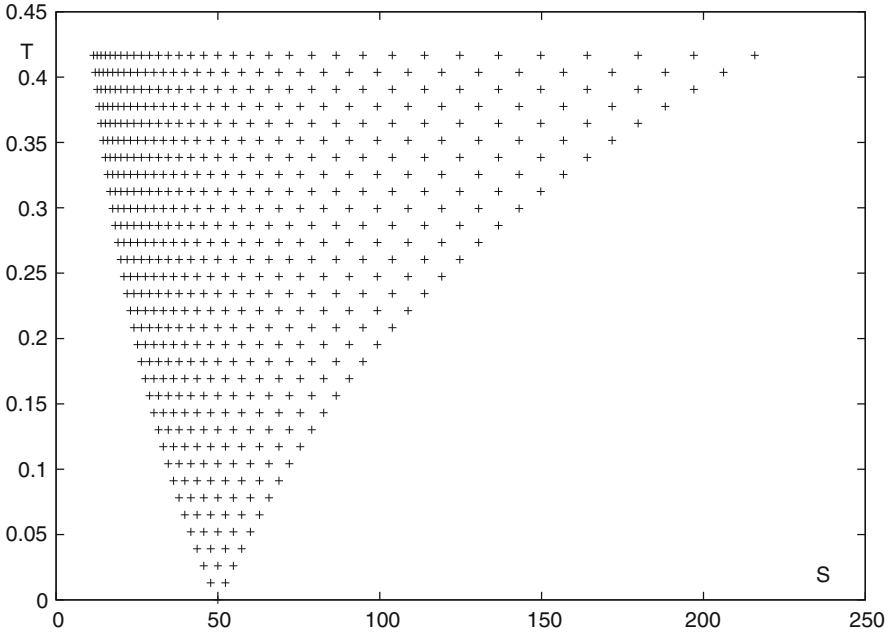

Fig. 1.10 Tree in the (S, t)-plane for $M = 32$ (data of Example 1.6)

1.4.3 Solution of the Equations

Using the abbreviation $\alpha := \mathrm{e}^{r\Delta t}$ we obtain by elimination (which the reader may check in more generality in Exercise 1.7b) the quadratic equation

$$0 = u^2 - u(\underbrace{\gamma\alpha^{-1} + \alpha\mathrm{e}^{\sigma^2\Delta t}}_{=:2\beta}) + \gamma\,,$$

with solutions $u = \beta \pm \sqrt{\beta^2 - \gamma}$. By virtue of $ud = \gamma$ and Vieta's Theorem, d is the solution with the minus sign. In summary, the three parameters u, d, p are given by

$$\begin{aligned} \beta &:= \tfrac{1}{2}(\gamma\mathrm{e}^{-r\Delta t} + \mathrm{e}^{(r+\sigma^2)\Delta t}) \\ u &= \beta + \sqrt{\beta^2 - \gamma} \\ d &= \gamma/u = \beta - \sqrt{\beta^2 - \gamma} \\ p &= \frac{\mathrm{e}^{r\Delta t} - d}{u - d}\,. \end{aligned} \tag{1.18}$$

A consequence of this approach is that up to terms of higher order the relation $u = \mathrm{e}^{\sigma\sqrt{\Delta t}}$ holds ($\longrightarrow$ Exercise 1.8). Therefore the extension of the tree in S-direction matches the volatility of the asset. So the tree is scaled well and will cover a relevant range of S-values.

1.4.4 A Basic Algorithm

Next we transform the binomial method into an algorithm.

1.4.4.1 Forward Phase: Initializing the Tree

Now the factors u and d can be considered as known, and the node values of S for each t_i until $t_M = T$ can be calculated. The current spot price $S = S_0$ for $t_0 = 0$ is the root of the tree.

(To adapt the matrix-like notation to the two-dimensional grid of the tree, this initial price will be also denoted $S_{0,0}$.) Each initial price S_0 leads to another tree of node values $S_{j,i}$.

$$\begin{aligned} &\textit{For } i = 1, 2, \ldots, M \textit{ calculate}: \\ &S_{j,i} := S_0 u^j d^{i-j}, \quad j = 0, 1, \ldots, i \end{aligned}$$

Now the grid points $(S_{j,i}, t_i)$ are fixed, on which approximations to the option values $V_{j,i} := V(S_{j,i}, t_i)$ are to be calculated.

1.4.4.2 Calculating the Option Value, Valuation on the Tree

For t_M and vanilla options, the payoff $V(S, t_M)$ is known from (1.1), (1.2). The payoff is valid for each S, including $S_{j,M} = Su^j d^{M-j}$, $j = 0, \ldots, M$. This defines the values $V_{j,M}$:

Call: $V(S(t_M), t_M) = \max\{S(t_M) - K, 0\}$, hence:

$$V_{j,M} := (S_{j,M} - K)^+ \tag{1.19}$$

Put: $V(S(t_M), t_M) = \max\{K - S(t_M), 0\}$, hence:

$$V_{j,M} := (K - S_{j,M})^+ \tag{1.20}$$

The **backward phase** recursively calculates for t_{M-1}, $t_{M-2}, \ldots$ the option values V for all t_i, starting from $V_{j,M}$. The recursion is based on Assumption 1.3, (Bi3). Repeating the equation that corresponds to (1.12) with double index leads to

$$S_{j,i}\mathrm{e}^{r\Delta t} = pS_{j,i}u + (1-p)S_{j,i}d\,,$$

and

$$S_{j,i}\mathrm{e}^{r\Delta t} = pS_{j+1,i+1} + (1-p)S_{j,i+1}\,.$$

As will be explained in Sect. 1.5, we relate the Assumption 1.3, (Bi3), of risk neutrality to V,

$$V_i = \mathrm{e}^{-r\Delta t}\,\mathsf{E}(V_{i+1})\,.$$

In double-index notation the resulting recursion is

$$V_{j,i} = \mathrm{e}^{-r\Delta t}\left(pV_{j+1,i+1} + (1-p)V_{j,i+1}\right)\,. \tag{1.21}$$

So far, this recursion for $V_{j,i}$ is merely an analogy, which might be seen as a further assumption. But the following Sect. 1.5 will give a justification for (1.21), which turns out to be a consequence of the no-arbitrage principle and the risk-neutral valuation.

For **European options**, (1.21) is a recursion for $i = M-1, \ldots, 0$, starting from (1.20), and terminating with $V_{0,0}$. (For an illustration see Fig. 1.11.) The obtained value $V_{0,0}$ is an approximation to the value $V(S_0, 0)$ of the continuous model, which results in the limit $M \to \infty$ ($\Delta t \to 0$). The accuracy of the approximation $V_{0,0}$ depends on M. This is reflected by writing $V_0^{(M)}$ ($\longrightarrow$ Exercise 1.9). The basic idea of the approach implies that the limit of $V_0^{(M)}$ for $M \to \infty$ is the Black–Scholes value $V(S_0, 0)$ ($\longrightarrow$ Exercises 1.10, 1.27, and Appendix D.1).

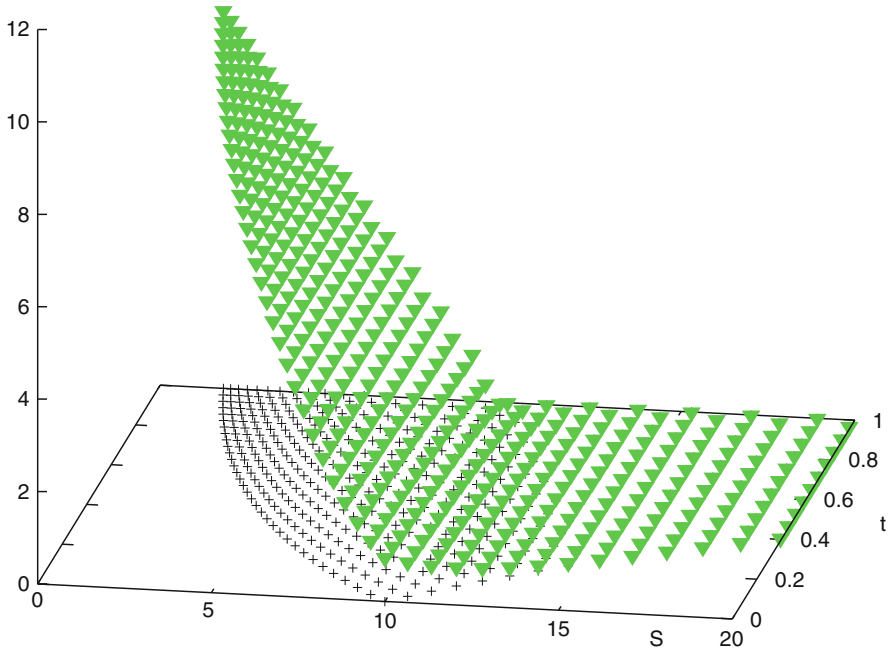

Fig. 1.11 In (S, t, V)-space: tree in the (S, t)-plane (*black* +) with (S, t, V)-points (*green triangles*) for $M = 32$ (data as in Fig. 1.5)

For **American options**, the above recursion must be modified by adding a test whether early exercise is to be preferred. To this end the value of (1.21) is compared with the value of the payoff $\Psi(S)$. In this context, the value (1.21) is the *continuation value*, denoted $V_{j,i}^{\text{cont}}$. And at any time t_i the holder optimizes the position and decides which of the two choices

$$\{\,\text{exercise},\ \text{continue to hold}\,\}$$

is preferable.[8] So the holder chooses the maximum

$$\max\{\,\Psi(S_{j,i}),\ V_{j,i}^{\text{cont}}\,\}\,.$$

This amounts to the *dynamic programming* principle: The optimality of the decision policy must be optimal also for any remaining time period. In summary, the

[8] Of course, the holder may wish to sell the option.

dynamic-programming procedure, based on the Eqs. (1.20) for i rather than M, combined with (1.21), reads as follows:

$$\begin{aligned} V_{j,i}^{\text{cont}} &:= \mathrm{e}^{-r\Delta t} \cdot (pV_{j+1,i+1} + (1-p)V_{j,i+1}) \\ V_{j,i} &= \max\left\{(S_{j,i}-K)^+,\ V_{j,i}^{\text{cont}}\right\} \text{ for a call}, \\ V_{j,i} &= \max\left\{(K-S_{j,i})^+,\ V_{j,i}^{\text{cont}}\right\} \text{ for a put}. \end{aligned} \tag{1.22}$$

The resulting algorithm is

Algorithm 1.4 (Binomial Method, Basic Version)

input:
r, σ, $S = S_0$, T, K, choice of put or call,
European or American, M.
Set the parameter γ, for example, according to (1.23).
calculate:
$\Delta t := T/M$, u, d, p from (1.18)
$S_{0,0} := S_0$
$S_{j,M} = S_{0,0}u^j d^{M-j}$, $j = 0, 1, \ldots, M$
(for American options, also $S_{j,i} = S_{0,0}u^j d^{i-j}$
for $0 < i < M$, $j = 0, 1, \ldots, i$)
valuation:
$V_{j,M}$ from (1.20)
$V_{j,i}$ for $i < M$ from (1.21) for European options,
and from (1.22) for American options.
output:
$V_{0,0}$ is the approximation $V_0^{(M)}$ to $V(S_0, 0)$.

Imagine that we color those nodes of the tree where early exercise is chosen. At these points the holding of the option is stopped. Accordingly, these points mark a part of the (S,t)-strip called the *stopping region*. The complementary part is the *continuation region*; we come back to this issue in Sect. 4.5.

1.4.5 Improving the Convergence

The convergence order of the binomial method should be one. Then, ideally, extrapolation would make sense (⟶ Exercise 1.11). But with $\gamma = 1$ in (1.17), the basic version of Algorithm 1.4 suffers from the fact that the payoff is not smooth at the strike K. The tree with $\gamma = 1$ is rigid, not depending on the location of the strike K. This affects the accuracy at nodes near the kink $(S,t,V) = (K,T,0)$. For $S_0 \neq K$ the accuracy of the $(\gamma = 1)$-version of Algorithm 1.4 depends on how the strike K is grasped by the tree and its grid points. The error depending on M may oscillate, which is mainly caused by the erratic way how the point $(S,t) = (K,T)$ takes its place among the nodes $S_{j,M}$.

Table 1.2 Results of Example 1.5; for γ see Exercise 1.7

M	$V^{(M)}(5,0)$ for $ud = 1$	$V^{(M)}(5,0)$ for $ud = \gamma$	With order
8	4.42507	4.43542	
16	4.42925	4.43325	0.833
32	4.429855	4.431933	0.923
64	4.429923	4.431218	0.963
128	4.430047	4.430846	0.982
256	4.430390	4.430657	0.991
2048	4.430451	4.430489	0.999
Black–Scholes	4.43046477621		

This flaw of the ($\gamma = 1$)-version can be cured in an easy way. Following an idea of [244], the tree can be bent such that for $i = M$ the medium grid point falls on the strike value K, no matter what (even) value of M is chosen. To this end, choose in (1.17) γ as

$$\gamma := \exp\left[\frac{2}{M}\log\frac{K}{S_0}\right] \tag{1.23}$$

($\longrightarrow$ Exercise 1.7). Corresponding values of u and d, given by (1.18), smooth the error significantly. We call the tree defined by (1.23) and (1.18) the *tilted tree*. With (1.23) all parameters depend on K. Also, the tilted-tree method leads to approximations V varying smoothly with S_0. The Algorithm 1.4 with (1.23) is straightforward to implement. With the tilted-tree version and an even value of $M/2$, extrapolation

$$V^{(M,\mathrm{extr})} := 2V^{(M)} - V^{(M/2)} \tag{1.24}$$

does make sense. For further means of smoothing, see Appendix D.1 and Exercise 1.12.

Example 1.5 (European Put) Choose $K = 10$, $S = S_0 = 5$, $r = 0.06$, $\sigma = 0.3$, $T = 1$.

Table 1.2 lists approximations $V^{(M)}$ to $V(5,0)$, both for $ud = 1$ and for $ud = \gamma$ with γ from (1.23). The two main columns of Table 1.2 are graphed in the two illustrations of Fig. 1.12. The convergence towards the Black–Scholes value $V(5,0)$ is visible; the latter was calculated by evaluating the analytic solution (1.7)–(1.10).[9] (In this book the number of printed decimals illustrates at best the attainable accuracy and does not reflect economic practice.)

[9] Recall that for European-style vanilla options an analytic solution exists, and Algorithm 1.4 is not needed. Hence, applying Algorithm 1.4 to Example 1.5 is only to create an ideal setting for the purpose of investigating accuracy and convergence.

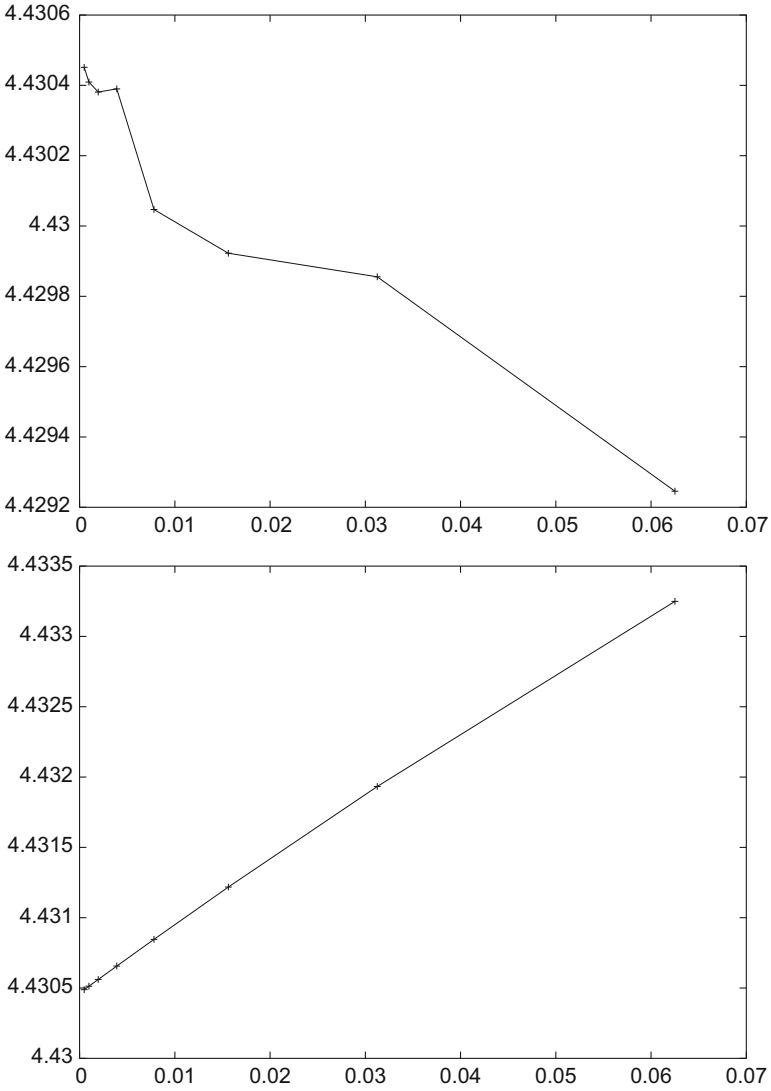

Fig. 1.12 Example 1.5: European-style option. Approximations $V^{(M)}$ over $\Delta t = 1/M$. *Top figure*: the basic Algorithm 1.4 with $ud = 1$, linear convergence is hardly visible; *bottom figure*: the improved algorithm of the tilted tree with $ud = \gamma$ and γ from (1.23), linear convergence is clearly visible

The convergence rate of Algorithm 1.4 is visible in the results of Table 1.2, and in Fig. 1.12. The rate is linear, $O(\Delta t) = O(M^{-1})$. For $S_0 \neq K$ and $ud = 1$ this rate is corrupted and hard to observe. The reader may wish to investigate more closely how the error of the basic version with $ud = 1$ decays with M ($\longrightarrow$ Exercise 1.9). It turns out that for the ($\gamma = 1$)-version of the binomial method the convergence in

M is not monotonic. It will not be recommendable to extrapolate these $V^{(M)}$-data to the limit $M \to \infty$, at least not the data of Table 1.2 ($ud = 1$).

But the linear convergence rate can be seen well from the much better results obtained for the tilted tree $ud = \gamma$ with γ from (1.23). The linear rate is reflected by the plots $V^{(M)}$ over M^{-1}, where the values of $V^{(M)}$ lie close to a straight line, which in this figure represents the linear error decay. Here extrapolation works well (illustrations in Fig. 1.13). The convergence rate can also be calculated from the data ($\longrightarrow$ Exercise 1.11). This can be seen from Table 1.2 in a perfect way.

In case the function $V(S, 0)$ is to be approximated for several S out of an interval of S-values, other methods should be applied. Figure 1.6 shows related results obtained by using the methods of Chap. 4.

Example 1.6 (American Put) Choose $K = 50,\ S = 50,\ r = 0.1,\ \sigma = 0.4,\ T = 0.41666\ldots$ ($\frac{5}{12}$ for 5 months).

Here the pricing is at the money, so $\gamma = 1$. Figure 1.10 shows the tree for $M = 32$. The corresponding approximation to V_0 is $V^{(32)} = 4.2719$, calculated with Algorithm 1.4; almost three digits are correct. With $M = 2048$ and extrapolation we obtain 4.2842. At the early-exercise curve the surface $V(S, t)$ is not C^2-smooth. As a consequence the convergence order is not as close to $q = 1$ as in Example 1.5. Note again that the function $V(S, 0)$ can be approximated with the methods of Chap. 4, compare Fig. 4.11.

1.4.6 Sensitivities

The sensitivity parameters at $(S, t) = (S_{0,0}, 0)$

$$\text{delta} = \frac{\partial V}{\partial S}, \quad \text{gamma} = \frac{\partial^2 V}{\partial S^2}, \quad \text{theta} = \frac{\partial V}{\partial t}, \tag{1.25}$$

can be approximated by difference quotients. The variations of V with S and t are expressed by the tree, and therefore information on derivatives can be obtained as by-product. For example, $\frac{V_{1,1}-V_{0,1}}{S_{1,1}-S_{0,1}}$ serves as a rough approximation for delta. But this quotient is evaluated at $t_1 = \Delta t$ rather than at $t = 0$. And a corresponding approximation of gamma requires three node values, which are available for t_2. These approximations are available at zero cost, but the accuracy may be poor.

To improve the accuracy, the difference quotients should be evaluated at the root node $(S, t) = (S_{0,0}, 0)$. This can be accomplished with a nice idea [297]. The tree can be extended by starting it with a root at $t_{-2} := -2\Delta t$ rather than at $t = 0$, with S-value $S_{-1,-2} := S_{0,0}/\gamma$. The extended tree of $M + 2$ slices follows the rules of Assumptions 1.3 and embeds the core tree. Now, $j = -1, \ldots, i + 1$. In this way, two additional lines of nodes are created, one at each side of the core tree. In particular, this creates two additional nodes at $t = 0$, with S-values $S_{-1,0}$ and $S_{1,0}$,

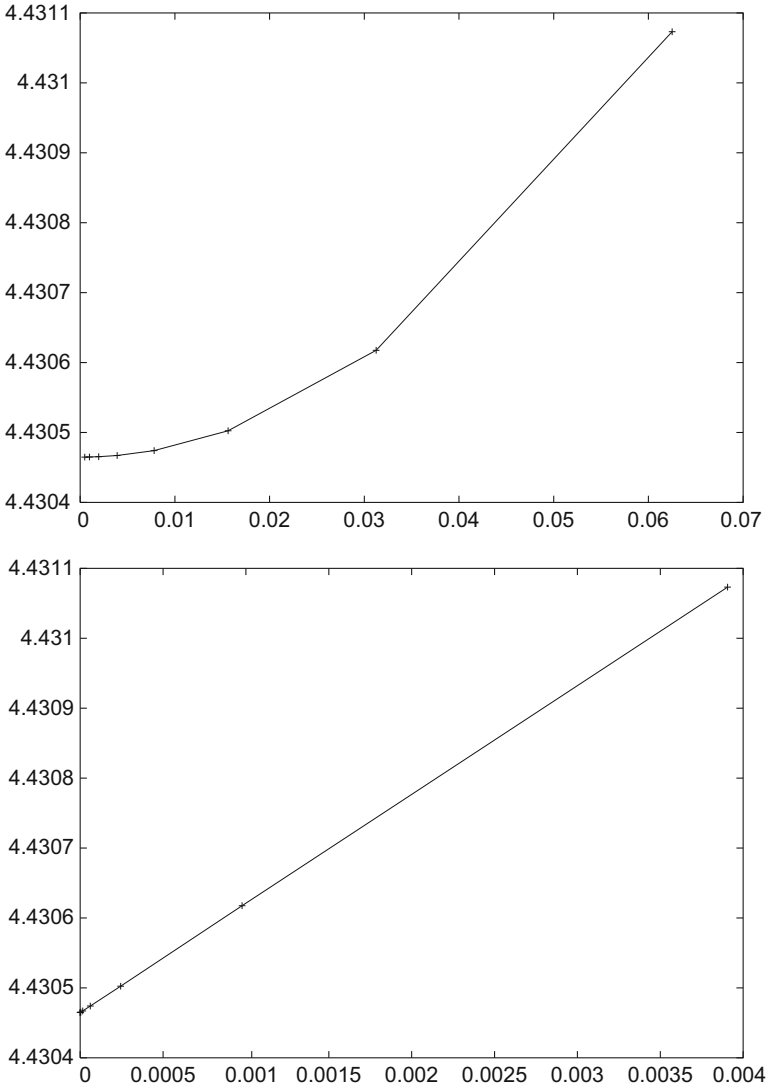

Fig. 1.13 Example 1.5: European-style option. Extrapolated approximations of the tilted tree showing quadratic convergence. *Top figure*: extrapolated values $V^{(M,\text{extr})}$ over $\Delta t = 1/M$, based on two approximations with M and $M/2$, $M/2$ even, $V^{(M,\text{extr})} := 2V^{(M)} - V^{(M/2)}$; *bottom figure*: $V^{(M,\text{extr})}$ over Δt^2

and corresponding V-values $V_{-1,0}$ and $V_{1,0}$. Figure 1.9 may serve as illustration, when *Sud* stands for $S_{0,0}$. The approximations are

$$\text{delta}: \quad \frac{V_{1,0} - V_{-1,0}}{S_{1,0} - S_{-1,0}}$$

$$\text{gamma}: \quad \frac{\frac{V_{1,0}-V_{0,0}}{S_{1,0}-S_{0,0}} - \frac{V_{0,0}-V_{-1,0}}{S_{0,0}-S_{-1,0}}}{(S_{1,0} - S_{-1,0})/2}$$

$$\text{theta}: \quad \frac{V_{0,0} - V_{-1,-2}}{2\Delta t} \text{ (for example, when } ud = 1)$$

The costs of calculating these difference quotients can be neglected, because essentially the tree is not recalculated. Compared with the overall costs of $O(M^2)$, the costs of the $2M + 5$ additional nodes of the improved version are relatively small as long as M is large. Algorithm 1.4 needs to be adapted (⟶ Exercise 1.13).

Since the above sensitivities with respect to S and t are revealed by one calculated tree, they can be considered as bargain greeks. In contrast, the sensitivities with respect to the parameters σ and r are more costly to approximate; these are the expensive greeks because the entire tree must be recalculated. For example, to set up a difference quotient for the greek vega$= \frac{\partial V}{\partial \sigma}$ requires to recalculate the tree for a parameter value σ_1 close to σ. If the corresponding value of the option obtained by the σ_1-tree is denoted V_1, then we have a difference-quotient approximation

$$\text{vega} \approx \frac{V - V_1}{\sigma - \sigma_1}\,.$$

For an improved accuracy at higher costs, one applies a symmetric difference quotient, for which the tree is recalculated on the other side, for $\sigma_2 := 2\sigma - \sigma_1$.

1.4.7 Extensions

The paying of dividends can be incorporated into the binomial algorithm. For a continuous dividend flow with constant rate δ, replace r in (1.18) by $r - \delta$, but not in the discounting in (1.21), (1.22). In case a discrete dividend D is paid at time instant t_D, the price of the asset drops by the same amount D. In practice, the input asset price S_0 is adjusted to $\tilde{S}_0$, by subtracting the current value of dividend payments. In the simplest case, with a single dividend payment at time t_D in $0 < t_D < T$ and constant rate r, start the algorithm from

$$\tilde{S}_0 := S_0 - D\mathrm{e}^{-rt_D}\,.$$

See also Appendix D.2 (⟶ Exercise 1.14)

An extension of the binomial model is the *trinomial model*. Here each mesh offers three outcomes, with probabilities p_1, p_2, p_3 and $p_1 + p_2 + p_3 = 1$. The reader may wish to derive the trinomial method (⟶ Exercise 1.15). For further hints, see Notes and Comments at the end of Chap. 1, and Appendix D.3.

1.5 Risk-Neutral Valuation

In the previous Sect. 1.4 we have used the Assumptions 1.3 to derive an algorithm for valuation of options. This Sect. 1.5 discusses the assumptions again, leading to a different interpretation.

The situation of a path-independent binomial process with the two factors u and d continues to be the basis of the argumentation. The scenario is illustrated in Fig. 1.14. Here the time period is the time to expiration T, which replaces Δt in the local mesh of Fig. 1.8. Accordingly, this global model is called *one-period model*. The one-period model with only two possible values of S_T has two clearly defined values of the payoff, namely, $V^{(d)}$ (corresponds to $S_T = S_0 d$) and $V^{(u)}$ (corresponds to $S_T = S_0 u$). In contrast to the Assumptions 1.3 we neither assume the risk-neutral world (Bi3) nor the corresponding probability $\mathsf{P}(\text{up}) = p$ from (Bi2). Instead we derive the probability using the no-arbitrage argument. In this section the factors u and d are assumed to be given.

Let us construct a portfolio of an investor with a short position in one option and a long position consisting of Δ shares of an asset, where the asset is the underlying of the option. The portfolio manager must **choose the number Δ of shares such that the portfolio is riskless.** That is, a hedging strategy is needed. To discuss the hedging properly assume that no funds are added or withdrawn.

By Π_t we denote the wealth of this portfolio at time t. Initially the value is

$$\Pi_0 = S_0 \cdot \Delta - V_0 \,, \tag{1.26}$$

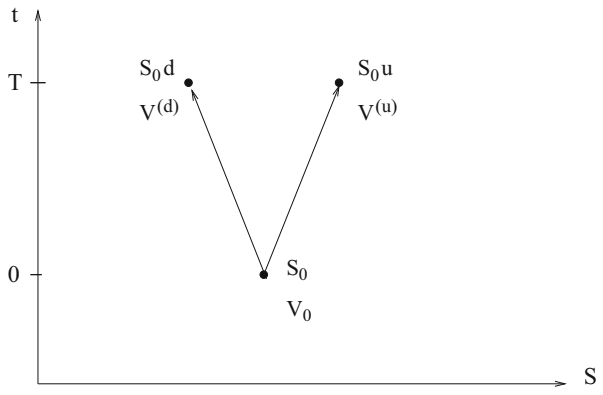

Fig. 1.14 One-period binomial model

where the value V_0 of the written option is not yet determined. At the end of the period the value V_T either takes the value $V^{(u)}$ or the value $V^{(d)}$. So the value of the portfolio Π_T at the end of the life of the option is either

$$\Pi^{(u)} = S_0 u \cdot \Delta - V^{(u)}$$

or

$$\Pi^{(d)} = S_0 d \cdot \Delta - V^{(d)} .$$

In the no-arbitrage world, Δ is chosen such that the value Π_T is riskless. Then all uncertainty is removed and $\Pi^{(u)} = \Pi^{(d)}$ must hold. This is equivalent to

$$(S_0 u - S_0 d) \cdot \Delta = V^{(u)} - V^{(d)} ,$$

which defines the strategy

$$\Delta = \frac{V^{(u)} - V^{(d)}}{S_0(u - d)} . \tag{1.27}$$

With this value of Δ the portfolio with initial value Π_0 evolves to the final value $\Pi_T = \Pi^{(u)} = \Pi^{(d)}$, regardless of whether the stock price moves up or down. Consequently the portfolio is riskless.

If we rule out early exercise, the final value Π_T is reached with certainty. The value Π_T must be compared to the alternative risk-free investment of an amount of money that equals the initial wealth Π_0, which after the time period T reaches the value $\mathrm{e}^{rT}\Pi_0$. Both the assumptions $\Pi_0 \mathrm{e}^{rT} < \Pi_T$ and $\Pi_0 \mathrm{e}^{rT} > \Pi_T$ would allow a strategy of earning a risk-free profit. This is in contrast to the assumed arbitrage-free world. Hence both $\Pi_0 \mathrm{e}^{rT} \geq \Pi_T$ and $\Pi_0 \mathrm{e}^{rT} \leq \Pi_T$ and equality must hold.[10] Accordingly the initial value Π_0 of the portfolio equals the discounted final value Π_T, discounted at the interest rate r,

$$\Pi_0 = \mathrm{e}^{-rT}\Pi_T .$$

This means

$$S_0 \cdot \Delta - V_0 = \mathrm{e}^{-rT}(S_0 u \cdot \Delta - V^{(u)}) ,$$

[10] For an American option it is not certain that Π_T can be reached because the holder may choose early exercise. In this situation we have only the inequality $\Pi_0 \mathrm{e}^{rT} \leq \Pi_T$.

which upon substituting (1.27) leads to the value V_0 of the option:

$$\begin{aligned} V_0 &= S_0 \cdot \Delta - \mathrm{e}^{-rT}(S_0 u \Delta - V^{(u)}) \\ &= \mathrm{e}^{-rT}\{\Delta \cdot [S_0 \mathrm{e}^{rT} - S_0 u] + V^{(u)}\} \\ &= \frac{\mathrm{e}^{-rT}}{u-d}\{(V^{(u)} - V^{(d)})(\mathrm{e}^{rT} - u) + V^{(u)}(u-d)\} \\ &= \frac{\mathrm{e}^{-rT}}{u-d}\{V^{(u)}(\mathrm{e}^{rT} - d) + V^{(d)}(u - \mathrm{e}^{rT})\} \\ &= \mathrm{e}^{-rT}\{V^{(u)}\frac{\mathrm{e}^{rT}-d}{u-d} + V^{(d)}\frac{u-\mathrm{e}^{rT}}{u-d}\} \\ &= \mathrm{e}^{-rT}\{V^{(u)}q + V^{(d)} \cdot (1-q)\} \end{aligned}$$

with

$$q := \frac{\mathrm{e}^{rT} - d}{u - d}\,. \tag{1.28}$$

We have shown that with q from (1.28) the value of the option is given by

$$V_0 = \mathrm{e}^{-rT}\{V^{(u)}q + V^{(d)} \cdot (1-q)\}\,. \tag{1.29}$$

The expression for q in (1.28) is identical to the formula for p in (1.13), which was derived in the previous section. Again we have

$$0 < q < 1 \quad \Longleftrightarrow \quad d < \mathrm{e}^{rT} < u\,.$$

Presuming these bounds for u and d, q can be interpreted as a probability Q. Then $qV^{(u)} + (1-q)V^{(d)}$ is the expected value of the payoff with respect to this probability (1.28),

$$\mathsf{E}_\mathsf{Q}(V_T) = qV^{(u)} + (1-q)V^{(d)}\,.$$

Now (1.29) can be written

$$V_0 = \mathrm{e}^{-rT}\mathsf{E}_\mathsf{Q}(V_T)\,. \tag{1.30}$$

That is, the value of the option is obtained by discounting the expected payoff [with respect to q from (1.28)] at the risk-free interest rate r. An analogous calculation shows

$$\mathsf{E}_\mathsf{Q}(S_T) = qS_0u + (1-q)S_0d = S_0\mathrm{e}^{rT}\,.$$

The probabilities p of Sect. 1.4 and q from (1.28) are defined by identical formulas (with T corresponding to Δt). Hence $p = q$, and $\mathsf{E}_\mathsf{P} = \mathsf{E}_\mathsf{Q}$. But the underlying arguments are different. Recall that in Sect. 1.4 we showed the implication

$$\mathsf{E}(S_T) = S_0 \mathrm{e}^{rT} \quad \Longrightarrow \quad p = \mathsf{P}(\text{up}) = \frac{\mathrm{e}^{rT} - d}{u - d} ,$$

whereas in this section we arrive at the implication

$$p = \mathsf{P}(\text{up}) = \frac{\mathrm{e}^{rT} - d}{u - d} \quad \Longrightarrow \quad \mathsf{E}(S_T) = S_0 \mathrm{e}^{rT} .$$

So both statements must be equivalent. Setting the probability of the up movement equal to p is equivalent to assuming that the expected return on the asset equals the risk-free rate. This can be rewritten as

$$\mathrm{e}^{-rT} \mathsf{E}_\mathsf{P}(S_T) = S_0 . \tag{1.31}$$

The important property expressed by Eq. (1.31) is that of a *martingale*: The random variable $\mathrm{e}^{-rT} S_T$ of the left-hand side has the tendency to remain at the same level. That is why a martingale is also called "fair game." A martingale displays no trend, where the trend is measured with respect to E_P. In the martingale property of (1.31) the discounting at the risk-free interest rate r exactly matches the risk-neutral probability P of (1.13)/(1.28). The specific probability for which (1.31) holds is also called *martingale measure*.

Summary of results for the one-period model: Under the Assumptions 1.2 of the market model, the choice Δ of (1.27) eliminates the random-dependence of the payoff and makes the portfolio riskless. There is a specific probability Q (P in Sect. 1.4) with $\mathsf{Q}(\text{up}) = q$, q from (1.27), such that the value V_0 satisfies (1.30), and S_0 the analogous property (1.31). These properties involve the risk-neutral interest rate r. That is, the option is valued in a risk-neutral world, and the corresponding Assumption 1.3 (Bi3) is meaningful.

In the real-world economy, growth rates in general are different from r, and individual subjective probabilities differ from our Q. But the assumption of a risk-neutral world leads to a fair valuation of options. The obtained value V_0 can be seen as a *rational* price. In this sense the resulting value V_0 applies to the real world. The risk-neutral valuation can be seen as a technical tool. The assumption of risk neutrality is just required to define and calculate a rational price or fair value of V_0. For this specific purpose we do not need actual growth rates of prices, and individual probabilities are not relevant. But note that we do not really assume that financial markets are actually free of risk.

The general principle outlined for the one-period model is also valid for the multiperiod binomial model and for the continuous model of Black and Scholes (⟶ Exercise 1.10).

The Δ of (1.27) is the hedge parameter *delta*, which eliminates the risk exposure of our portfolio caused by the written option. In multiperiod models and continuous models Δ must be adapted dynamically. The expression (1.27) can be seen as a discretized version of the continuous-case definition of delta,

$$\Delta = \Delta(S,t) = \frac{\partial V(S,t)}{\partial S}\,.$$

1.6 Stochastic Processes

Brownian motion originally meant the erratic motion of a particle (pollen) on the surface of a fluid, caused by tiny impulses of molecules. Wiener suggested a mathematical model for this motion, the *Wiener process*. But earlier Bachelier had applied Brownian motion to model the motion of stock prices, which instantly respond to the numerous upcoming information similar as pollen react to the impacts of molecules (Fig. 1.15). To model such behavior, we use stochastic processes.

A *stochastic process* is a family of random variables X_t, which are defined for a set of parameters t ($\longrightarrow$ Appendix B.1). Here we consider the continuous-time situation. That is, $t \in \mathbb{R}$ varies continuously in a time interval I, which typically represents $0 \le t \le T$. A more complete notation for a stochastic process is

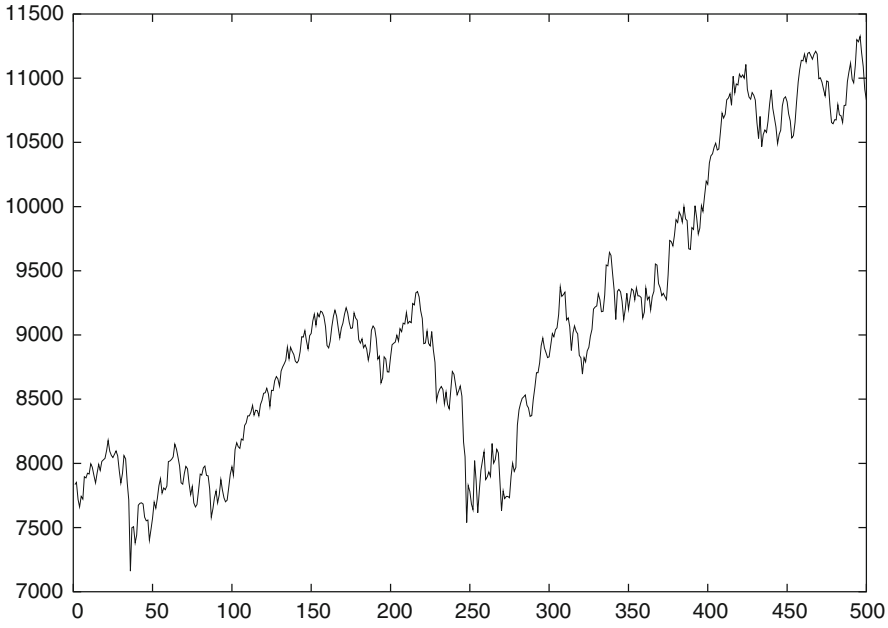

Fig. 1.15 The Dow at 500 trading days from September 8, 1997 through August 31, 1999

$\{X_t, t \in I\}$, or $(X_t)_{0 \le t \le T}$. Let the chance "play," then the resulting function X_t is called *realization* or *path* of the stochastic process.

Special properties of stochastic processes have lead to the following names:

- *Gaussian process:* All finite-dimensional distributions $(X_{t_1}, \ldots, X_{t_k})$ are Gaussian. Hence specifically X_t is distributed normally for all t.
- *Markov process:* Only the present value of X_t is relevant for its future motion. That is, the past history is fully reflected in the present value.[11]

An example of a process that is both Gaussian and Markov, is the Wiener process. The Wiener process is an important building block for models of financial markets, and is the main theme of this section.

1.6.1 Wiener Process

Definition 1.7 (Wiener Process, Standard Brownian Motion) A Wiener process (or standard Brownian motion; notation W_t or W) is a stochastic process for $t \ge 0$ with the properties

(a) $W_0 = 0$.
(b) $W_t \sim \mathcal{N}(0, t)$ for all $t \ge 0$.
That is, for each t the random variable W_t is *distributed normally*, with mean $\mathsf{E}(W_t) = 0$ and variance $\mathsf{Var}(W_t) = \mathsf{E}(W_t^2) = t$.
(c) All increments $\Delta W_t := W_{t+\Delta t} - W_t$ on non overlapping time intervals are *independent*.
That is, the displacements $W_{t_2} - W_{t_1}$ and $W_{t_4} - W_{t_3}$ are independent for all $0 \le t_1 < t_2 \le t_3 < t_4$.
(d) W_t varies continuously with t.

Generally for $0 \le s < t$ the property $W_t - W_s \sim \mathcal{N}(0, t-s)$ holds, in particular

$$\mathsf{E}(W_t - W_s) = 0\,, \tag{1.32}$$

$$\mathsf{Var}(W_t - W_s) = \mathsf{E}((W_t - W_s)^2) = t - s\,. \tag{1.33}$$

These relations can be derived from Definition 1.7 ($\longrightarrow$ Exercise 1.16). The relation (1.33) is also known as

$$\mathsf{E}((\Delta W_t)^2) = \Delta t\,, \tag{1.34}$$

[11] This assumption together with the assumption of an immediate reaction of the market to arriving information are called *hypothesis of the efficient market* [44].

where $\Delta W_t := W_t - W_{t-\Delta t}$. The independence of the increments according to Definition 1.7(c) implies for $t_{j+1} > t_j$ the independence of W_{t_j} and $(W_{t_{j+1}} - W_{t_j})$, but not of $W_{t_{j+1}}$ and $(W_{t_{j+1}} - W_{t_j})$. Wiener processes are examples of martingales—there is no drift. This process is an integral element of more involved models. For example, $X_t := \alpha + \mu t + W_t$ is a general Brownian motion with drift μ.

Discrete-Time Model
Let $\Delta t > 0$ be a constant time increment. For the discrete instances $t_j := j\Delta t$ the value W_t can be written as a sum of increments ΔW_k,

$$W_{j\Delta t} = \sum_{k=1}^{j} \underbrace{\left(W_{k\Delta t} - W_{(k-1)\Delta t}\right)}_{=:\Delta W_k} .$$

By the properties of the Wiener process, the ΔW_k are independent and normally distributed with $\mathsf{Var}(\Delta W_k) = \Delta t$. Increments ΔW with such a distribution can be calculated from standard normally distributed random numbers Z. The implication

$$Z \sim \mathcal{N}(0,1) \implies Z \cdot \sqrt{\Delta t} \sim \mathcal{N}(0, \Delta t)$$

leads to the discrete model of a Wiener process, with

$$\Delta W_k = Z\sqrt{\Delta t} \text{ for } Z \sim \mathcal{N}(0,1) \text{ for each } k\,. \tag{1.35}$$

We summarize the numerical simulation of a Wiener process as follows:

Algorithm 1.8 (Simulation of a Wiener Process)

Start: $t_0 = 0$, $W_0 = 0$; *choose* Δt.
loop for $j = 1, 2, \ldots$:
$t_j = t_{j-1} + \Delta t$
draw $Z \sim \mathcal{N}(0,1)$
$W_j = W_{j-1} + Z\sqrt{\Delta t}$.

The drawing of Z—that is, the calculation of $Z \sim \mathcal{N}(0,1)$—will be explained in Chap. 2. The values W_j are realizations of W_t at the discrete points t_j. The Fig. 1.16 shows a realization of a Wiener process; 5000 calculated points (t_j, W_j) are joined by linear interpolation.

Almost all realizations of Wiener processes are nowhere differentiable. This becomes intuitively clear when the difference quotient

$$\frac{\Delta W_t}{\Delta t} = \frac{W_{t+\Delta t} - W_t}{\Delta t}$$

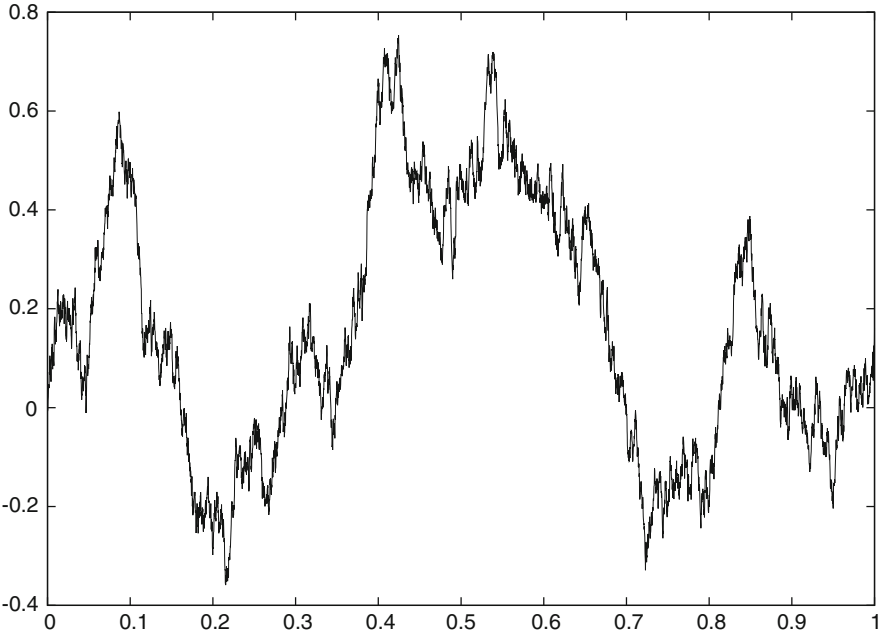

Fig. 1.16 Realization of a Wiener process, with $\Delta t = 0.0002$

is considered. Because of relation (1.33) the standard deviation of the numerator is $\sqrt{\Delta t}$. Hence for $\Delta t \to 0$ the normal distribution of the difference quotient disperses and no convergence can be expected.

1.6.2 Stochastic Integral

For motivation, let us suppose that the price development of an asset is described by a Wiener process W_t. Let $b(t)$ be the number of units of the asset held in a portfolio at time t. We start with the simplifying assumption that trading is only possible at discrete time instances t_j, which define a partition of the interval $0 \le t \le T$. Then the trading strategy b is piecewise constant,

$$\begin{aligned} &b(t) = b(t_{j-1}) \quad \text{for} \quad t_{j-1} \le t < t_j \\ &\text{and} \quad 0 = t_0 < t_1 < \ldots < t_N = T\,. \end{aligned} \tag{1.36}$$

Such a function $b(t)$ is called *step function*. The trading gain for the subinterval $t_{j-1} \le t < t_j$ is given by $b(t_{j-1})(W_{t_j} - W_{t_{j-1}})$, and

$$\sum_{j=1}^{N} b(t_{j-1})(W_{t_j} - W_{t_{j-1}}) \tag{1.37}$$

represents the trading gain over the time period $0 \le t \le T$. The trading gain (possibly < 0) is determined by the strategy $b(t)$ and the price process W_t.

We now drop the assumption of fixed trading times t_j and allow b to be arbitrary continuous functions. This leads to the question whether (1.37) has a limit when with $N \to \infty$ the size of all subintervals tends to 0. If W_t would be of bounded variation than the limit exists and is called *Riemann–Stieltjes integral*

$$\int_0^T b(t)\,\mathrm{d}W_t\,.$$

In our situation this integral generally does not exist because almost all Wiener processes are not of bounded variation. That is, the *first variation* of W_t, which is the limit of

$$\sum_{j=1}^{N} |W_{t_j} - W_{t_{j-1}}|\,,$$

is unbounded even in case the lengths of the subintervals vanish for $N \to \infty$.

Although this statement is not of primary concern for the theme of this book,[12] we digress for a discussion because it introduces the important rule $(\mathrm{d}W_t)^2 = \mathrm{d}t$. For an arbitrary partition of the interval $[0, T]$ into N subintervals the inequality

$$\sum_{j=1}^{N} |W_{t_j} - W_{t_{j-1}}|^2 \le \max_j(|W_{t_j} - W_{t_{j-1}}|) \sum_{j=1}^{N} |W_{t_j} - W_{t_{j-1}}| \tag{1.38}$$

holds. The left-hand sum in (1.38) is the *second variation* and the right-hand sum the first variation of W for a given partition into subintervals. The expectation of the left-hand sum can be calculated using (1.33),

$$\sum_{j=1}^{N} \mathsf{E}(W_{t_j} - W_{t_{j-1}})^2 = \sum_{j=1}^{N} (t_j - t_{j-1}) = t_N - t_0 = T\,.$$

But even convergence in the mean holds:

Lemma 1.9 (Second Variation: Convergence in the Mean) *Let* $t_0 = t_0^{(N)} < t_1^{(N)} < \ldots < t_N^{(N)} = T$ *be a sequence of partitions of the interval* $t_0 \le t \le T$ *with*

$$\delta_N := \max_j (t_j^{(N)} - t_{j-1}^{(N)})\,. \tag{1.39}$$

[12] The less mathematically oriented reader may like to skip the rest of this subsection.

Then (dropping the $^{(N)}$)

$$\underset{\delta_N \to 0}{\text{l.i.m.}} \sum_{j=1}^{N} (W_{t_j} - W_{t_{j-1}})^2 = T - t_0 \,. \tag{1.40}$$

Proof The statement (1.40) means convergence in the mean (⟶ Appendix B.1). Because of $\sum \Delta t_j = T - t_0$ we must show

$$\mathsf{E}\left(\sum_j ((\Delta W_j)^2 - \Delta t_j) \right)^2 \to 0 \quad \text{for} \quad \delta_N \to 0$$

for $\Delta t_j := t_j - t_{j-1}$ and $\Delta W_j := W_{t_j} - W_{t_{j-1}}$. Carrying out the multiplications and taking the mean gives

$$2 \sum_j (\Delta t_j)^2$$

(⟶ Exercise 1.17). This can be bounded by $2(T - t_0)\delta_N$, which completes the proof.

With $\Delta W_t = W_t - W_{t-\Delta t}$, part of the derivation can be summarized to

$$\mathsf{E}((\Delta W_t)^2 - \Delta t) = 0 \,, \quad \mathsf{Var}((\Delta W_t)^2 - \Delta t) = 2(\Delta t)^2 \,.$$

Symbolically, this probabilistic property of the Wiener process is written

$$(\mathrm{d}W_t)^2 = \mathrm{d}t \,. \tag{1.41}$$

It will be needed in subsequent sections.

Now we know enough about the convergence of the left-hand sum of (1.38) and turn to the right-hand side of this inequality. The continuity of W_t implies

$$\max_j |W_{t_j} - W_{t_{j-1}}| \to 0 \quad \text{for } \delta_N \to 0 \,.$$

Convergence in the mean applied to (1.38) shows that the vanishing of this factor must be compensated by an unbounded growth of the other factor, to make (1.40) happen. So

$$\sum_{j=1}^{N} |W_{t_j} - W_{t_{j-1}}| \to \infty \quad \text{for } \delta_N \to 0 \,.$$

In summary, Wiener processes are not of bounded variation, and the integration with respect to W_t can not be defined as an elementary limit of (1.37).

The aim is to construct a stochastic integral

$$\int_{t_0}^{t} f(s)\,\mathrm{d}W_s$$

for general stochastic integrands $f(t)$. For our purposes it suffices to briefly sketch the Itô integral, which is the prototype of a stochastic integral.

For a partition of the interval $t_0 \le s \le t$ into n parts with step function b_n from (1.36), an integral can be defined via the sum (1.37),

$$\int_{t_0}^{t} b_n(s)\,\mathrm{d}W_s := \sum_{j=1}^{n} b_n(t_{j-1})(W_{t_j} - W_{t_{j-1}})\,. \tag{1.42}$$

This is the Itô integral over a step function b_n. In case the $b_n(t_{j-1})$ are random variables, the b_n are called *simple processes*. Then the Itô integral is again defined by (1.42). Stochastically integrable functions f can be obtained as limits of simple processes b_n in the sense

$$\mathsf{E}\Big[\int_{t_0}^{t} (f(s) - b_n(s))^2\,\mathrm{d}s\Big] \to 0 \quad \text{for } \delta_n \to 0\,. \tag{1.43}$$

Convergence in terms of integrals $\int \mathrm{d}s$ carries over to integrals $\int \mathrm{d}W_t$. This is achieved by applying Cauchy convergence $\mathsf{E}\int (b_n - b_m)^2 \mathrm{d}s \to 0$ and the *isometry*

$$\mathsf{E}\Big[\Big(\int_{t_0}^{t} b(s)\,\mathrm{d}W_s\Big)^2\Big] = \mathsf{E}\Big[\int_{t_0}^{t} b(s)^2\,\mathrm{d}s\Big]\,.$$

Hence the integrals $\int b_n(s)\mathrm{d}W_s$ form a Cauchy sequence with respect to convergence in the mean. Accordingly the Itô integral of f is defined as

$$\int_{t_0}^{t} f(s)\,\mathrm{d}W_s := \text{l.i.m.}_{\delta_n \to 0} \int_{t_0}^{t} b_n(s)\,\mathrm{d}W_s\,,$$

for simple processes b_n defined by (1.43). The value of the integral is independent of the choice of the b_n in (1.43). The Itô integral as function in t is a stochastic process with the martingale property.

If an integrand $a(x,t)$ depends on a stochastic process X_t, the function f is given by $f(t) = a(X_t, t)$. For the simplest case of a constant integrand $a(X_t, t) = a_0$ the Itô integral can be reduced via (1.42) to

$$\int_{t_0}^{t} \mathrm{d}W_s = W_t - W_{t_0}\,.$$

For the "first" nontrivial Itô integral consider $X_t = W_t$ and $a(W_t, t) = W_t$. Its solution will be presented in Sect. 3.2.

Wiener processes are the driving machines for diffusion models (next section). There are other stochastic processes that can be used for modeling financial markets. For several models jump processes are considered. We turn to jump processes in Sect. 1.9.

1.7 Diffusion Models

Many fundamental models of financial markets use Wiener processes as driving process. These are the diffusion models discussed in this section. We discuss the main representative geometric Brownian motion, and explain the risk-neutral valuation in this context. Then we turn to more general processes, such as mean reversion.

1.7.1 Itô Process

Phenomena in nature, technology and economy are often modeled by means of deterministic differential equations $\dot{x} = \frac{\mathrm{d}}{\mathrm{d}t}x = a(x,t)$. This kind of modeling neglects stochastic fluctuations and is not appropriate for stock prices. If processes x are to include Wiener processes as special case, the derivative $\frac{\mathrm{d}}{\mathrm{d}t}x$ is meaningless. To circumvent non-differentiability, *integral equations* are used to define a general class of stochastic processes. Randomness is inserted additively,

$$x(t) = x_0 + \int_{t_0}^{t} a(x(s), s)\,\mathrm{d}s \quad + \text{randomness}\,,$$

with an Itô integral with respect to the Wiener process W_t. The first integral in the resulting integral equation is an ordinary (Lebesgue- or Riemann-) integral. The final integral equation is symbolically written as a "stochastic differential equation" (SDE) and named after Itô.

Definition 1.10 (Itô Stochastic Differential Equation) An Itô stochastic differential equation is

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}W_t\,; \tag{1.44}$$

this together with $X_{t_0} = X_0$ is a symbolic short form of the integral equation

$$X_t = X_{t_0} + \int_{t_0}^{t} a(X_s, s)\,\mathrm{d}s + \int_{t_0}^{t} b(X_s, s)\,\mathrm{d}W_s\,. \tag{1.45}$$

The terms in (1.44)/(1.45) are named as follows:

- $a(X_t, t)$: drift term or drift coefficient
- $b(X_t, t)$: diffusion coefficient

The integral equation (1.45) defines a large class of stochastic processes X_t; solutions X_t of (1.44) are called Itô process, or stochastic diffusion. In our context, X_0 can be assumed as a deterministic constant, namely, the observed initial state "today."

As intended, the Wiener process is a special case of an Itô process, because from $X_t = W_t$ the trivial SDE $\mathrm{d}X_t = \mathrm{d}W_t$ follows, hence the drift vanishes, $a = 0$, and $b = 1$. For $b \equiv 0$ the SDE is deterministic.

An experimental approach may help to develop an intuitive understanding of Itô processes. The simplest numerical method combines the discretized version of the Itô SDE

$$\Delta X_t = a(X_t, t)\Delta t + b(X_t, t)\Delta W_t \tag{1.46}$$

with the Algorithm 1.8 for approximating a Wiener process, using the same Δt for both discretizations. The result is

Algorithm 1.11 (Euler Discretization of an SDE) *Approximations y_j to X_{t_j} are calculated by*

Start: t_0, $y_0 = X_0$ *; choose* Δt.
loop $j = 0, 1, 2, \ldots$:
$t_{j+1} = t_j + \Delta t$
$\Delta W = Z\sqrt{\Delta t}$ *with* $Z \sim \mathcal{N}(0, 1)$
$y_{j+1} = y_j + a(y_j, t_j)\Delta t + b(y_j, t_j)\Delta W$.

In the simplest setting, the *step length* Δt is chosen equidistant, $\Delta t = T/M$ for a suitable integer M. Of course the accuracy of the approximation depends on the choice of Δt ($\longrightarrow$ Chap. 3). In case the functions a and b are easy to evaluate, the greatest effort may be to calculate random numbers $Z \sim \mathcal{N}(0, 1)$ ($\longrightarrow$ Sect. 2.3). Solutions to the SDE or to its discretized version for a given realization of the Wiener process are called *trajectories* or paths. By *simulation* of the SDE we understand the calculation of one or more trajectories. For the purpose of visualization, the discrete data are mostly joined by straight lines.

Example 1.12 $\mathrm{d}X_t = 0.05X_t\,\mathrm{d}t + 0.3X_t\,\mathrm{d}W_t$

Without the diffusion term the exact solution would be $X_t = X_0\mathrm{e}^{0.05t}$. For $X_0 = 50$, $t_0 = 0$ and a time increment $\Delta t = 1/250$ the Fig. 1.17 depicts a trajectory X_t of the SDE for $0 \le t \le 1$. For another realization of a Wiener process W_t the solution looks different. This is demonstrated for a similar SDE in Fig. 1.18.

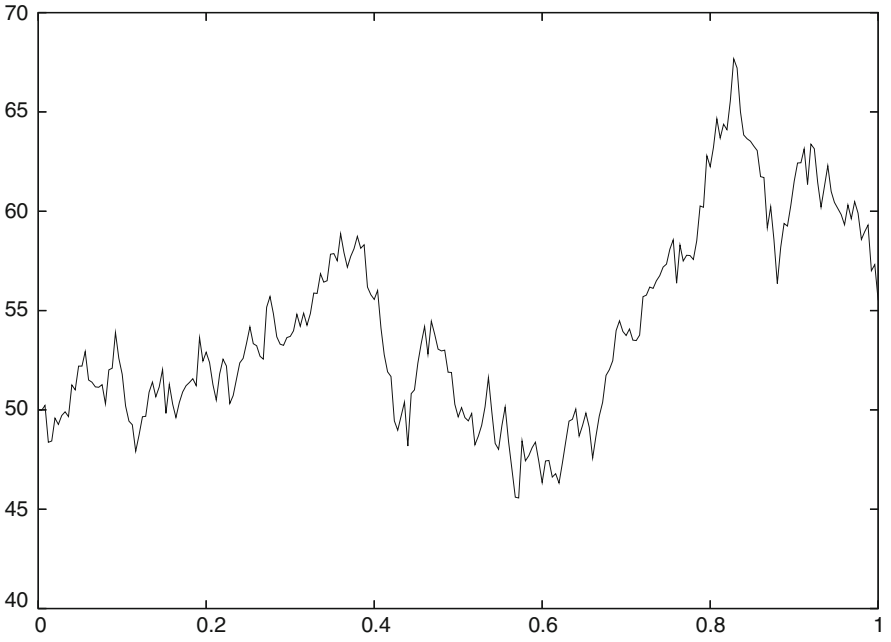

Fig. 1.17 Numerically approximated trajectory X_t of Example 1.12 with $a = 0.05X_t$, $b = 0.3X_t$, $\Delta t = 1/250$, $X_0 = 50$

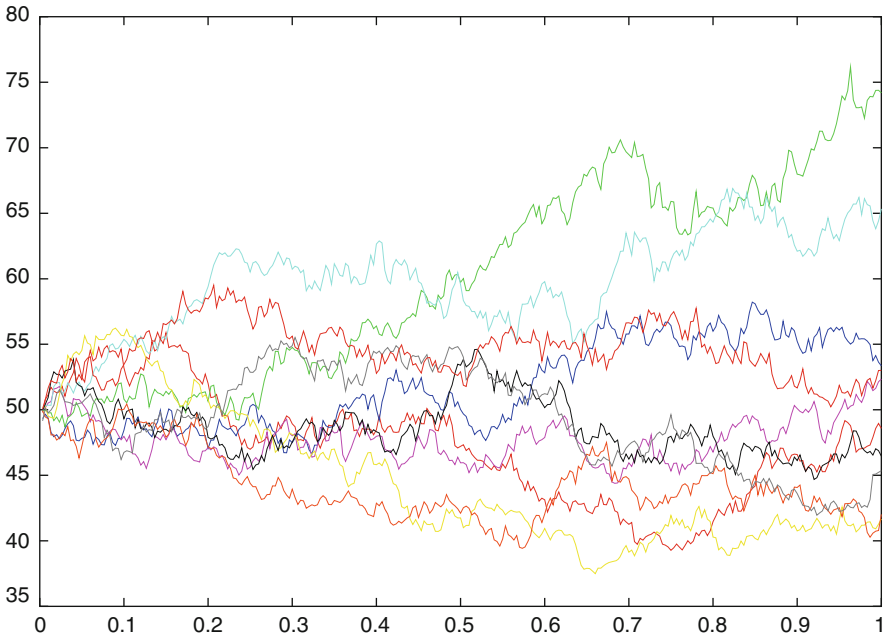

Fig. 1.18 Ten paths of SDE (1.47) with $S_0 = 50$, $\mu = 0.1$ and $\sigma = 0.2$; plot of S against t

1.7.2 Geometric Brownian Motion

Next we discuss one of the most important continuous models for the motion of asset prices, which will be denoted S_t. This standard model for $S > 0$ assumes that the relative change (return) $\mathrm{d}S/S$ of a security in the time interval $\mathrm{d}t$ is composed of a deterministic drift $\mu\,\mathrm{d}t$ plus stochastic fluctuations in the form $\sigma\mathrm{d}W_t$:

Model 1.13 (Geometric Brownian Motion, GBM) *Any solution S_t of the SDE*

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t \tag{1.47}$$

is called geometric Brownian motion, GBM.
This SDE (1.47) is linear in $X_t = S_t$, and $a(S_t, t) = \mu S_t$ is the drift rate with the expected *rate of return* μ, $b(S_t, t) = \sigma S_t$, σ is the volatility. (Compare Example 1.12 and Fig. 1.17.) The geometric Brownian motion of (1.47) is the reference model on which, for example, the Black–Scholes model is based. To match Assumptions 1.2 assume that μ and σ are constant.

A theoretical solution of (1.47) will be given in (1.71). The deterministic part of (1.47) is the ordinary differential equation

$$\dot{S} = \mu S$$

with solution $S_t = S_0 \mathrm{e}^{\mu(t-t_0)}$. For the linear SDE of (1.47) the expectation $\mathsf{E}(S_t)$ solves $\dot{S} = \mu S$. Hence

$$S_0 \mathrm{e}^{\mu(t-t_0)} = \mathsf{E}(S_t \,|\, S_{t_0} = S_0)$$

is the expectation of the stochastic process and μ is the expected continuously compounded return earned by an investor per year, conditional on starting at S_0. The rate of return μ is also called *growth rate*. The function $S_0 \mathrm{e}^{\mu(t-t_0)}$ can be seen as a core about which the process fluctuates. Accordingly the simulated values S_1 of the ten trajectories in Fig. 1.18 group around the value $50 \cdot \mathrm{e}^{0.1} \approx 55.26$.

Let us test empirically how the values S_1 distribute about their expected value. To this end calculate, for example, 10,000 trajectories and count how many of the terminal values S_1 fall into the subintervals $k5 \le S < (k+1)5$, for $k = 0, 1, 2 \ldots$. Figure 1.19 shows the resulting histogram. Apparently the distribution is skewed. We revisit this distribution in the next section.

A discrete version of (1.47) is

$$\frac{\Delta S}{S} = \mu \Delta t + \sigma Z \sqrt{\Delta t}\,, \tag{1.48}$$

known from Algorithm 1.11. This approximation is valid as long as Δt is small and $S > 0$ ($\longrightarrow$ Exercise 1.18). The relative return reflected by the ratio $\frac{\Delta S}{S}$ is called

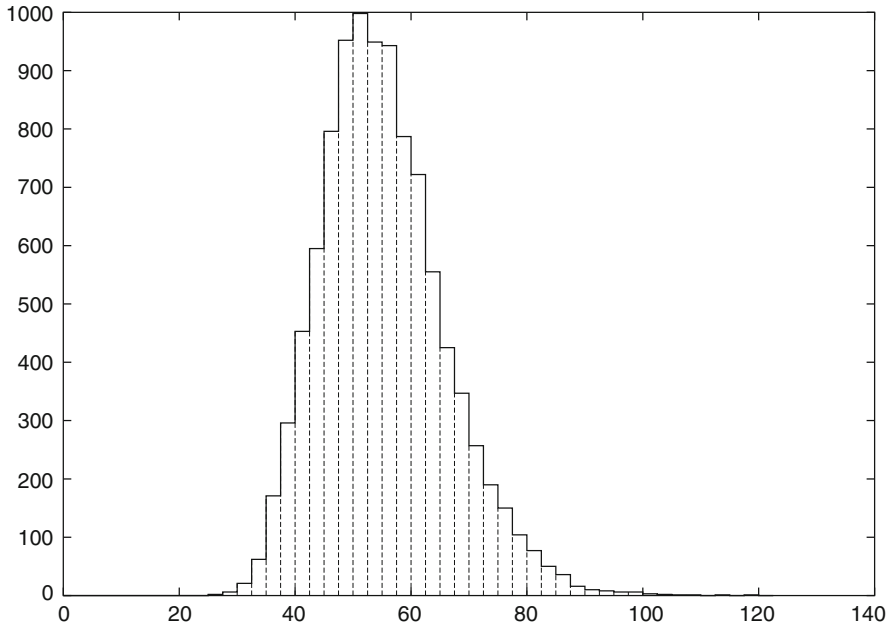

Fig. 1.19 Histogram of 10,000 calculated values S_1 corresponding to (1.47), with $S_0 = 50$, $\mu = 0.1$, $\sigma = 0.2$

one-period *simple return*, where we interpret Δt as one period. According to (1.48) this return satisfies

$$\frac{\Delta S}{S} \sim \mathcal{N}(\mu \Delta t, \sigma^2 \Delta t)\,. \tag{1.49}$$

The distribution of the simple return matches actual market data in a crude approximation, see for instance Fig. 1.23. This allows to calculate estimates of historical values of the volatility σ.[13] Of course this assumes the market data to be correctly described by GBM. We will return to this in Sect. 1.8.

1.7.3 Risk-Neutral Valuation

We digress for the length of this subsection and again turn to the topic of a risk-neutral valuation, now for the continuous-time setting. In Sect. 1.5 we have shown

$$V_0 = \mathrm{e}^{-rT} \mathsf{E}_{\mathsf{Q}}(V_T)$$

[13] For the *implied volatility* see Exercise 1.19.

for the one-period model. Formally, the same holds true for the market model based on GBM. But now the understanding of the risk-neutral probability Q is more involved. This subsection sketches the framework for GBM.

Let us rewrite GBM from (1.47) to get

$$\begin{aligned} \mathrm{d}S_t &= rS_t\,\mathrm{d}t + (\mu - r)S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t \\ &= rS_t\,\mathrm{d}t + \sigma S_t\left[\frac{\mu - r}{\sigma}\,\mathrm{d}t + \mathrm{d}W_t\right], \end{aligned} \tag{1.50}$$

where W_t is Wiener process under the probability measure P. In the reality of the market, an investor expects $\mu > r$ as compensation for the risk that is higher for stocks than for bonds. In this sense, the quotient γ of the *excess return* $\mu - r$ to the risk σ,

$$\gamma := \frac{\mu - r}{\sigma}\,, \tag{1.51}$$

is called *market price of risk*. With this variable γ, (1.50) is written

$$\mathrm{d}S_t = rS_t\,\mathrm{d}t + \sigma S_t\left[\gamma\,\mathrm{d}t + \mathrm{d}W_t\right]. \tag{1.52}$$

For $\gamma \neq 0$ the drifted Brownian motion W_t^γ defined by

$$\mathrm{d}W_t^\gamma = \gamma\,\mathrm{d}t + \mathrm{d}W_t \tag{1.53}$$

is no Wiener process under P. But under certain assumptions on γ there is another probability measure Q such that the process W_t^γ is a (standard) Wiener process under Q.[14] Equation (1.52) becomes

$$\mathrm{d}S_t = rS_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t^\gamma\,. \tag{1.54}$$

Comparing this SDE to (1.47), notice that the growth rate μ is replaced by the risk-free rate r. Together, the transition consists of

$$\begin{aligned} \mu &\to r \\ \mathsf{P} &\to \mathsf{Q} \\ W &\to W^\gamma \end{aligned}$$

which is called **risk-neutral valuation principle** for GBM. To simulate (1.54) under Q, just apply the standard Algorithm 1.8 for the Wiener process W_t^γ. Then the rate r in (1.54) and W_t^γ correspond to the "risk-neutral measure" Q.

[14] Girsanov's theorem, see Appendix B.2. Q and P are equivalent.

What is the reason for adjusting the probability measure $\mathsf{P} \to \mathsf{Q}$? The advantage of the risk-neutral measure Q is that the r-discounted process $\mathrm{e}^{-rt}S_t$ is a martingale under Q,

$$\mathrm{d}(\mathrm{e}^{-rt}S_t) = \sigma \mathrm{e}^{-rt}S_t \,\mathrm{d}W_t^{\gamma} \,.$$

The **fundamental theorem of asset pricing** states that a market model is free of arbitrage if and only if there exists a probability measure Q such that the discounted asset prices are martingales with respect to Q [170]. Hence the property of $\mathrm{e}^{-rt}S_t$ having no drift is an essential ingredient of a no-arbitrage market and a prerequisite to modeling options. For a thorough discussion of the continuous model, martingale theory is used. (Some more background and explanation is provided by Appendix B.3.) Let us summarize the situation in a remark:

Remark 1.14 (Risk-Neutral Valuation Principle) For modeling options with underlying GBM, the original probability is adjusted to the risk-neutral probability Q. To simulate the process under Q, the return rate μ is replaced by the risk-free interest rate r, and W_t^{γ} is approximated as Wiener process.

1.7.4 Mean Reversion

The assumptions of a constant interest rate r and a constant volatility σ are quite restrictive. To overcome this simplification, SDEs for processes r_t and σ_t have been constructed that control r_t or σ_t stochastically. One class of models is based on the SDE

$$\mathrm{d}r_t = \alpha(R - r_t)\,\mathrm{d}t + \sigma^{\mathrm{r}} r_t^{\beta}\,\mathrm{d}W_t \,, \quad \alpha > 0 \,, \tag{1.55}$$

again with driving force W_t as Wiener process.[15] The drift term in (1.55) is positive for $r_t < R$ and negative for $r_t > R$, which causes a pull to R. This effect is called *mean reversion*. A *frequency* parameter α influences the strength of the reversion. The parameter R, which may depend on t, corresponds to a long-run mean of the interest rate over time. SDE (1.55) defines a general class of models, including several interesting special cases known under special names:

- $\beta = 0, R = 0$: Ornstein–Uhlenbeck process (OU)
- $\beta = 0, R > 0$: Vasicek model
- $\beta = \frac{1}{2}, R > 0$: Cox–Ingersoll–Ross process (CIR)

Hull and White have extended the Vasicek model by incorporating time dependence in the parameters. The CIR model [86] is also called *square-root process*.

[15] Notation: β is exponent, σ^{r} is the volatility of r_t.

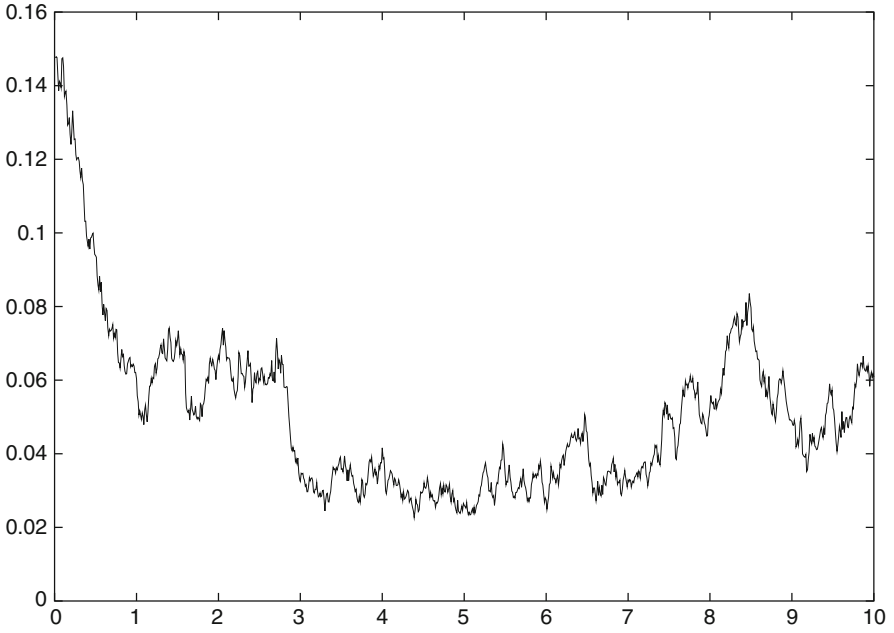

Fig. 1.20 Simulation r_t of the Cox–Ingersoll–Ross model (1.55) with $\beta = 0.5$ for $R = 0.05$, $\alpha = 1$, $\sigma^{\mathrm{r}} = 0.1$, $r_0 = 0.15$, $\Delta t = 0.01$

An illustration of the mean reversion is provided by Fig. 1.20. In a transient phase (until $t \approx 1$ in the run documented in the figure) the relatively large deterministic term dominates, and the range $r \approx R$ is reached quickly. Thereafter the stochastic term dominates, and r dances about the mean value R. Figure 1.20 shows this for a Cox–Ingersoll–Ross model. For a discussion of related models we refer to [191, 234, 237]. The *calibration* of the models (that is, the adaption of the parameters to the data) is a formidable task (⟶ Sect. 1.10).

The SDE (1.55) is of a different kind as the GBM in (1.47). Coupling the SDE for r_t to that for S_t leads to a system of two SDEs. Even larger systems are obtained when further SDEs are coupled to define a stochastic process R_t or to calculate stochastic volatilities. Related examples are given by Examples 1.15 and 1.16 below. In particular for modeling options, stochastic volatilities have shown great potential.

1.7.5 Vector-Valued Stochastic Differential Equations

The Itô equation (1.44) is formulated as scalar equation; accordingly the SDE (1.47) represents a *one-factor model*. The general *multifactor* version can be written in the same notation. Then $X_t = (X_t^{(1)}, \ldots, X_t^{(n)})$, and $a(X_t, t)$ are n-dimensional

vectors. In the general situation, the Wiener process can be m-dimensional, with components $W_t^{(1)}, \ldots, W_t^{(m)}$. Then $b(X_t, t)$ is an $(n \times m)$-matrix, with elements b_{ik}. The interpretation of the SDE systems is componentwise. The scalar stochastic integrals are sums of m stochastic integrals,

$$X_t^{(i)} = X_0^{(i)} + \int_0^t a_i(X_s, s)\,\mathrm{d}s + \sum_{k=1}^{m} \int_0^t b_{ik}(X_s, s)\,\mathrm{d}W_s^{(k)}\,, \tag{1.56}$$

for $i = 1, \ldots, n$, and $t_0 = 0$ for convenience. Or in the symbolic SDE notation, this system reads

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}W_t\,, \tag{1.57}$$

where $b\,\mathrm{d}W$ is a multiplication matrix times vector. When we take the components of the vector $\mathrm{d}W$ as uncorrelated,

$$\mathsf{E}(\mathrm{d}W^{(k)}\mathrm{d}W^{(j)}) = \begin{cases} 0 & \text{for } k \neq j \\ \mathrm{d}t & \text{for } k = j\,, \end{cases} \tag{1.58}$$

then possible correlations between the components of $\mathrm{d}X$ must be carried by b.[16]

Example 1.15 (Mean-Reverting Volatility Tandem) We consider a three-factor model [186] with asset price S_t, instantaneous spot volatility σ_t and an averaged volatility ζ_t serving as mean-reverting parameter:

$$\begin{aligned} \mathrm{d}S &= \sigma S\,\mathrm{d}W^{(1)} \\ \mathrm{d}\sigma &= -(\sigma - \zeta)\mathrm{d}t + \alpha\sigma\,\mathrm{d}W^{(2)} \\ \mathrm{d}\zeta &= \beta(\sigma - \zeta)\mathrm{d}t \end{aligned}$$

Here and sometimes later on, we suppress the subscript t, which is possible when the role of the variables as stochastic processes is clear from the context. The rate of return μ of S is zero; $\mathrm{d}W^{(1)}$ and $\mathrm{d}W^{(2)}$ may be correlated; α and β are two parameters. As seen from the SDE, the stochastic volatility σ follows the mean volatility ζ and is simultaneously perturbed by a Wiener process. Both σ and ζ provide mutual mean reversion, and stick together. Accordingly the two SDEs for σ and ζ may be seen as a tandem controlling the dynamics of the volatility. We recommend numerical tests. For motivation see Fig. 1.21.

Example 1.16 (Heston's Model) Heston [178] uses an Ornstein–Uhlenbeck process to model a stochastic volatility σ_t. Then the variance $v_t := \sigma_t^2$ follows a

[16] We come back to this issue in Sects. 2.3.4 and 3.5.5, and in Exercise 3.17.

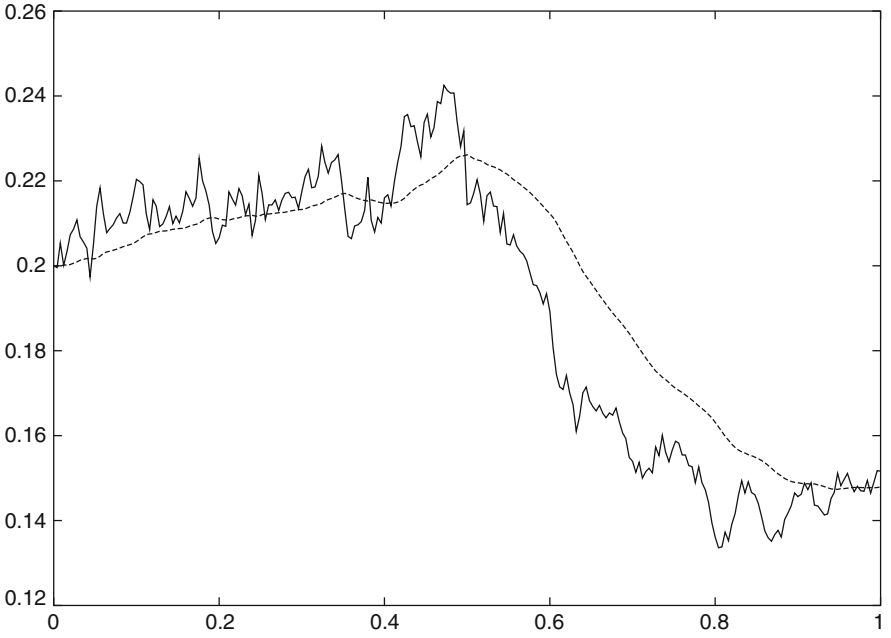

Fig. 1.21 Example 1.15, $\alpha = 0.3$, $\beta = 10$, $\sigma_0 = \zeta_0 = 0.2$, realization of the volatility tandem σ_t, ζ_t (*dashed*) for $0 \le t \le 1$, $\Delta t = 0.004$

Cox–Ingersoll–Ross process (1.55). ($\longrightarrow$ Exercise 1.20) The system of Heston's model is

$$\begin{aligned} \mathrm{d}S_t &= \mu S_t \,\mathrm{d}t + \sqrt{v_t}\, S_t \,\mathrm{d}W_t^{(1)} \\ \mathrm{d}v_t &= \kappa(\theta - v_t)\,\mathrm{d}t + \sigma^{\mathrm{v}} \sqrt{v_t}\,\mathrm{d}W_t^{(2)} \end{aligned} \tag{1.59}$$

with two correlated Wiener processes $W_t^{(1)}$, $W_t^{(2)}$ and suitable parameters $\mu, \kappa, \theta, \sigma^{\mathrm{v}}$, ρ. Here σ^{v} denotes a "volatility" of v_t and ρ is the correlation between $W_t^{(1)}$, $W_t^{(2)}$. Hidden parameters are the initial values $S_0,\ v_0$, if not available. Hence five or more parameters need to calibrated. The frequently used model of Heston establishes a correlation between price and volatility.

Computational Matters
Stochastic differential equations are simulated in the context of Monte Carlo methods. Thereby, the SDE is integrated N times, with N large ($N = 10,000$ or much larger). Then the weight of any single trajectory is almost negligible.

Expectation and variance are calculated over the N trajectories. Generally this costs an enormous amount of computing time. The required instruments are:

- Generating $\mathcal{N}(0,1)$-distributed random numbers ($\longrightarrow$ Chap. 2)
- Integration methods for SDEs ($\longrightarrow$ Chap. 3)

Applying Algorithm 1.11 to S_t from (1.47) or to the squared volatility v_t from (1.59) should produce positive values. For this matter, see Sect. 3.3.1.

1.8 Itô Lemma and Applications

Itô's lemma is most fundamental for stochastic processes. It may help, for example, to derive solutions of SDEs ($\longrightarrow$ Exercise 1.21). Suppose a "chain" of two functions X_t and $g(X_t, t)$. When a differential equation for X_t is given, what is the differential equation for $g(X_t, t)$?

1.8.1 Itô Lemma

Itô's lemma is the stochastic counterpart of the chain rule for deterministic functions $x(t)$ and $y(t) := g(x(t), t)$, which is

$$\frac{\mathrm{d}}{\mathrm{d}t} g(x(t), t) = \frac{\partial g}{\partial x} \cdot \frac{\mathrm{d}x}{\mathrm{d}t} + \frac{\partial g}{\partial t} ,$$

and can be written

$$\mathrm{d}x = a(x(t), t)\,\mathrm{d}t \;\Rightarrow\; \mathrm{d}g = \left(\frac{\partial g}{\partial x}\, a + \frac{\partial g}{\partial t} \right) \mathrm{d}t \,.$$

Here we state the one-dimensional version of the Itô lemma; for the multidimensional version see the Appendix B.2.

Lemma 1.17 (Itô) *Suppose X_t follows an Itô process (1.44),*

$$\mathrm{d}X_t = a(X_t, t)\mathrm{d}t + b(X_t, t)\mathrm{d}W_t \,,$$

and let $g(x, t)$ be a $\mathcal{C}^{2,1}$-smooth function (continuous $\frac{\partial g}{\partial x}$, $\frac{\partial^2 g}{\partial x^2}$, $\frac{\partial g}{\partial t}$). Then $Y_t := g(X_t, t)$ follows an Itô process with the same Wiener process W_t:

$$\mathrm{d}Y_t = \left(\frac{\partial g}{\partial x}\, a + \frac{\partial g}{\partial t} + \frac{1}{2}\, \frac{\partial^2 g}{\partial x^2}\, b^2 \right) \mathrm{d}t + \frac{\partial g}{\partial x}\, b\, \mathrm{d}W_t \tag{1.60}$$

where the derivatives of g as well as the coefficient functions a and b in general depend on the arguments (X_t, t).

For a proof we refer to [12, 291, 307, 345]. Here we confine ourselves to the basic idea: When t varies by Δt, then X by $\Delta X = a \cdot \Delta t + b \cdot \Delta W$ and Y by

$$\Delta Y = g(X + \Delta X, t + \Delta t) - g(X, t)\,.$$

The Taylor expansion of ΔY begins with the linear part $\frac{\partial g}{\partial x}\Delta X + \frac{\partial g}{\partial t}\Delta t$, in which $\Delta X = a\Delta t + b\Delta W$ is substituted. The additional term with the derivative $\frac{\partial^2 g}{\partial x^2}$ is new and is introduced via the $O(\Delta X^2)$-term of the Taylor expansion,

$$\frac{1}{2}\,\frac{\partial^2 g}{\partial x^2}(\Delta X)^2 = \frac{1}{2}\,\frac{\partial^2 g}{\partial x^2}\,b^2(\Delta W)^2 + \text{ t.h.o.}$$

Because of (1.41), $(\Delta W)^2 \approx \Delta t$, the leading term is also of the order $O(\Delta t)$ and belongs to the linear terms. Taking correct limits (similar as in Lemma 1.9) one obtains the integral equation represented by (1.60).

1.8.2 Consequences for Geometric Brownian Motion

Suppose the stock price follows a geometric Brownian motion, hence $X_t = S_t$, $a = \mu S_t$, $b = \sigma S_t$, for constant μ, σ. The value V_t of an option depends on S_t, $V_t = V(S_t, t)$. Assuming a $\mathcal{C}^2$-smooth value function V depending on S and t, we apply Itô's lemma. For $V(S, t)$ in the place of $g(x, t)$ the result is

$$\mathrm{d}V_t = \left(\frac{\partial V}{\partial S}\mu S_t + \frac{\partial V}{\partial t} + \frac{1}{2}\frac{\partial^2 V}{\partial S^2}\sigma^2 S_t^2\right)\mathrm{d}t + \frac{\partial V}{\partial S}\sigma S_t\,\mathrm{d}W_t\,. \tag{1.61}$$

This SDE is used to derive the Black–Scholes equation, see Appendix A.4.

As second application of Itô's lemma consider the log transformation $Y_t = \log(S_t)$ for $S > 0$, viz $g(x, t) := \log(x)$, for S_t solving GBM with constant μ, σ. Itô's lemma leads to the linear SDE

$$\mathrm{d}Y_t = \mathrm{d}\log S_t = (\mu - \frac{1}{2}\sigma^2)\,\mathrm{d}t + \sigma\mathrm{d}W_t\,. \tag{1.62}$$

In view of (1.44) the solution is straightforward:

$$\begin{aligned} Y_t &= Y_{t_0} + (\mu - \tfrac{1}{2}\sigma^2)\int_{t_0}^{t}\mathrm{d}s + \sigma\int_{t_0}^{t}\mathrm{d}W_s \\ &= Y_{t_0} + (\mu - \tfrac{1}{2}\sigma^2)(t - t_0) + \sigma(W_t - W_{t_0}) \end{aligned} \tag{1.63}$$

From the properties of the Wiener process W_t we conclude that Y_t is distributed normally. To write down the density function $\hat{f}(Y_t)$, the mean $\hat{\mu} := \mathsf{E}(Y_t)$ and the variance $\hat{\sigma}$ are needed. For this linear SDE (1.62) the expectation $\mathsf{E}(Y_t)$ satisfies the deterministic part

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{E}(Y_t) = \mu - \frac{\sigma^2}{2}\,.$$

The solution of $\dot{y} = \mu - \frac{\sigma^2}{2}$ with initial condition $y(t_0) = y_0$ is

$$y(t) = y_0 + (\mu - \frac{\sigma^2}{2})(t - t_0)\,.$$

In other words, the expectation of the Itô process Y_t is

$$\hat{\mu} := \mathsf{E}(\log S_t) = \log S_0 + (\mu - \frac{\sigma^2}{2})(t - t_0)\,.$$

Analogously, we see from the differential equation for $\mathsf{E}(Y_t^2)$ (or from the analytic solution of the SDE for Y_t) that the variance of Y_t is $\sigma^2(t - t_0)$. In view of (1.62) the simple SDE for Y_t implies that the stochastic fluctuation of Y_t is that of σW_t, namely, $\hat{\sigma}^2 := \sigma^2(t - t_0)$. So, from (B.9) with $\hat{\mu}$ and $\hat{\sigma}$, the density of Y_t is

$$\widehat{f}(Y_t) := \frac{1}{\sigma\sqrt{2\pi(t-t_0)}} \exp\left\{ -\frac{\left(Y_t - y_0 - \left(\mu - \frac{\sigma^2}{2}\right)(t-t_0)\right)^2}{2\sigma^2(t-t_0)} \right\}.$$

Back transformation using $Y = \log(S)$ and considering $\mathrm{d}Y = \frac{1}{S}\mathrm{d}S$ and $\widehat{f}(Y)\mathrm{d}Y = \frac{1}{S}\widehat{f}(\log S)\mathrm{d}S = f(S)\mathrm{d}S$ yields the density of $S_t > 0$:

$$f_{\mathrm{GBM}}(S, t-t_0;\, S_0, \mu, \sigma) := \frac{1}{S\sigma\sqrt{2\pi(t-t_0)}} \exp\left\{ -\frac{\left(\log(S/S_0) - \left(\mu - \frac{\sigma^2}{2}\right)(t-t_0)\right)^2}{2\sigma^2(t-t_0)} \right\} \tag{1.64}$$

This is the density of the *lognormal* distribution, conditional on $S_{t_0} = S_0$. It describes the probability of a transition

$$(S_0, t_0) \longrightarrow (S, t)$$

under the basic assumption that the stock price S_t follows a geometric Brownian motion (1.47). The distribution is skewed, see Fig. 1.22. Now the skewed behavior coming out of the experiment reported in Fig. 1.19 is clear. Notice that the

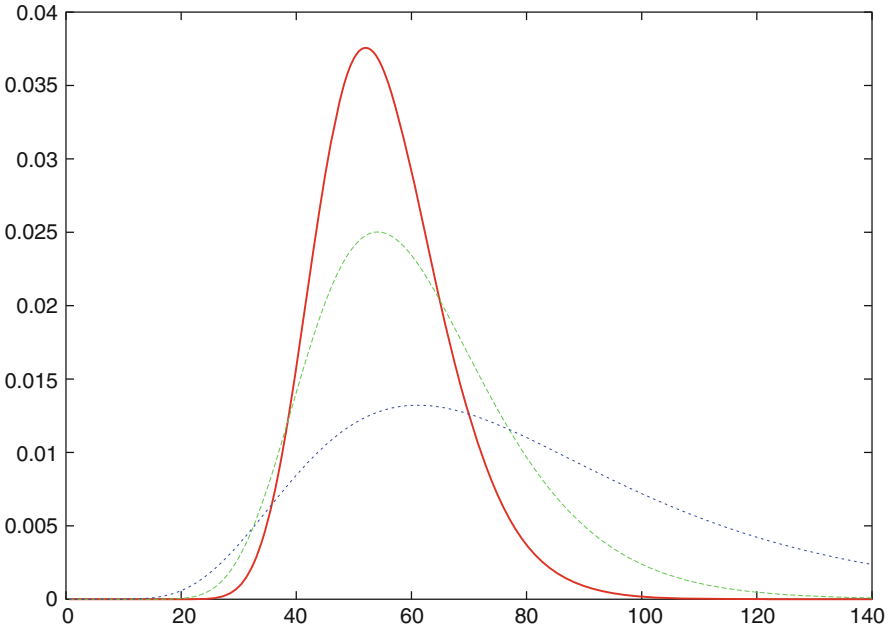

Fig. 1.22 Density (1.64) over S for $\mu = 0.1$, $\sigma = 0.2$, $S_0 = 50$, $t_0 = 0$ and $t = 0.5$ (*dotted green curve* with steep gradient), $t = 1$ (*solid red curve*), $t = 2$ (*dashed green*) and $t = 5$ (*dotted blue* with flat gradient)

parameters in Figs. 1.19 and 1.22 match. Figure 1.19 is an approximation of the solid curve in Fig. 1.22.

In summary, the assumption of GBM amounts to

$$S_t = S_0 \, \exp(Y_t) \,, \tag{1.65}$$

where the log-price Y_t is a Brownian motion with drift, $Y_t = (\mu - \frac{1}{2}\sigma^2)t + \sigma W_t$.

Having derived the density (1.64), we now can prove Eq. (1.15), with $\mu = r$ according to Remark 1.14 ($\longrightarrow$ Exercise 1.22). For vector-valued SDEs an appropriate version of the Itô lemma is (B.17).

1.8.3 Integral Representation

An important application of a known density function is that it allows for an integral representation of European options. This will be revisited in Sect. 3.5.1, where we show for a European put under GBM

$$V(S_0, 0) = \mathrm{e}^{-rT} \int_0^\infty (K - S_T)^+ f_{\mathrm{GBM}}(S_T, T;\, S_0, r, \sigma) \, \mathrm{d}S_T \,. \tag{1.66}$$

Note the risk-free interest rate r as argument in the density. This reflects that the integral is the conditional expectation of the payoff Ψ under the assumed risk-neutral measure,

$$\mathsf{E}_{\mathsf{Q}} = \int_0^\infty \Psi(S_T) \cdot \text{density}\, \mathrm{d}S_T\,.$$

The integral representation for European-style options

$$V(S_0, 0) = \mathrm{e}^{-rT}\mathsf{E}_{\mathsf{Q}}(V(S_T, T) \mid S_t \text{ starting from } (S_0, 0))\,. \tag{1.67}$$

holds for arbitrary payoff functions and density functions of a general class of valuation models.

1.8.4 Bermudan Options

The integral representation (1.66)/(1.67) for European options can be applied to approximate American options. To this end, discretize the time interval $0 \le t \le T$ into an equidistant grid of time instances t_i, similar as done for the binomial method of Sect. 1.4:

$$\Delta t := \frac{T}{M}\,, \quad t_i := i\,\Delta t \quad (i = 0, \ldots, M)\,.$$

This defines lines in the (S, t)-domain, and cuts it into M slices. An option that restricts early exercise to specified discrete dates during its life is called a *Bermudan option*. The above slicing defines an artificial Bermudan option, constructed for the purpose of approximating the corresponding American option.

Let $V^{\mathrm{Ber}(M)}$ denote the value of the Bermudan option in the above setting of M slices of equal size. Clearly,

$$V^{\mathrm{Eur}} \le V^{\mathrm{Ber}(M)} \le V^{\mathrm{Am}} \text{ for all } M,$$

because of the additional exercise possibilities of an otherwise identical option. Note that the Bermudan options serve as lower bounds for the American option, and $V^{\mathrm{Eur}} = V^{\mathrm{Ber}(1)}$. One can show

$$\lim_{M\to\infty} V^{\mathrm{Ber}(M)} = V^{\mathrm{Am}}\,.$$

Hence, for suitable M the value $V^{\mathrm{Ber}(M)}$ can be used as approximation to V^{Am}.

Let us consider the time slice $t_i \le t \le t_{i+1}$ for any i. For the valuation of the option's value at t_i, the "inner payoff" is $V(S, t_{i+1})$ along the line $t = t_{i+1}$. Since a Bermudan option can not be exercised for $t_i < t < t_{i+1}$, its continuation value for

t_i is given by the integral representation of a European option. This continuation value is

$$V^{\text{cont}}(x, t_i) = \mathrm{e}^{-r(t_{i+1}-t_i)} \int_{-\infty}^{\infty} V(\xi, t_{i+1})\, f(\xi,\ t_{i+1} - t_i;\ x, \ldots)\, \mathrm{d}\xi \tag{1.68}$$

for arbitrary x. Here a value S at line $t = t_i$ is represented by x, and the price at t_{i+1} by ξ. The dots stand for the parameters of the risk-neutral valuation of the chosen model, and f is its density conditional on $S_{t_i} = x$. For an n-factor model, the domain of integration is $\mathbb{R}^n$.

Since the Bermudan option can be exercised at t_i, its value is again given by the dynamic programming principle,

$$V(x, t_i) = \max\{\, \Psi(x),\ V^{\text{cont}}(x, t_i)\,\}, \tag{1.69}$$

where Ψ denotes the payoff. Equations (1.68)/(1.69) define a backward recursive algorithm for $i = M-1, \ldots, 0$. It starts from the given payoff at T, which provides $V(S, t_M)$. That is, only for the first time level $i = M-1$, the option is "vanilla," whereas for $i < M-1$ the inner payoffs are given by (1.69).

In the algorithm, the evaluation of the integral in (1.68) is done by numerical quadrature (⟶ Appendix C.1), and the continuation value functions V^{cont} are approximated by interpolating functions $C(x)$ based on m nodes in x-space [309]. In the simplest case of a one-factor model ($n = 1$), the nodes may represent equidistantly chosen S_j ($1 \le j \le m$). The inner payoffs are denoted g_i, and the Bermudan option is to be evaluated at $(x, 0) := (S, 0)$.

Algorithm 1.18 (Bermudan Option)

Set m *nodes* $x_1, \ldots, x_m \in \mathbb{R}^n$.
$g_M(x) := V(x, t_M) = V(x, T) = \Psi(x)$.
Calculate recursively backwards ($i = M-1, \ldots, 0$)*:*
(1) input: g_{i+1}
loop ($j = 1, \ldots, m$)*: calculate by* ***quadrature***

$$q_j := \mathrm{e}^{-r(t_{i+1}-t_i)} \int g_{i+1}(\xi)\, f(\xi,\ t_{i+1} - t_i;\ x_j, \ldots)\, \mathrm{d}\xi$$

output: $q_1, \ldots, q_m$
(2) ***interpolate*** $(x_1, q_1), \ldots, (x_m, q_m)$.
output: $C(x)$
(3) $g_i(x) := \max\{\Psi(x),\ C(x)\}$.

The final $g_0(x)$ is the approximation of $V^{\text{Ber}(M)}(x, 0)$, which in turn approximates $V^{\text{Am}}(x, 0)$. The integral (1.68) is taken over a suitably truncated interval $\xi_{\min} \le \xi \le \xi_{\max}$. The method works also for general non-GBM models, as long as they are not path-dependent. The order of convergence in Δt is linear. If necessary, the nodes

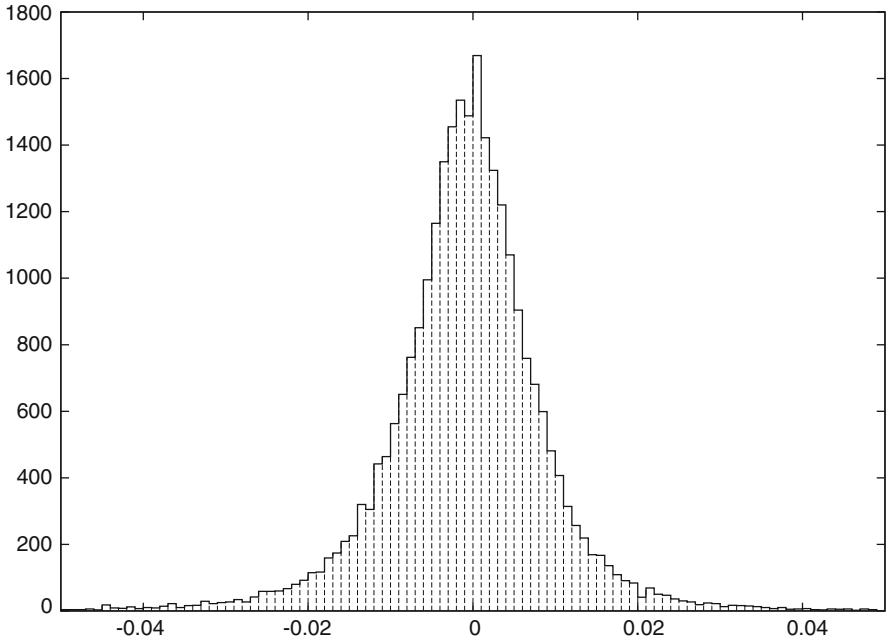

Fig. 1.23 Histogram (compare Exercise 1.23): frequency of daily log returns $R_{i,i-1}$ of the Dow in the time period 1901–1999

x_j can be re-adjusted after each i; extrapolation is possible. For example, when two values $V^{\mathrm{Ber}(M)}(x,0)$, $V^{\mathrm{Ber}(2M)}(x,0)$ are available, an improved approximation is

$$\bar{V} = 2\,V^{\mathrm{Ber}(2M)}(x,0) - V^{\mathrm{Ber}(M)}(x,0)\,.$$

For details see [309].

1.8.5 Empirical Tests

It is inspiring to test the idealized Model 1.13 of a geometric Brownian motion against actual empirical data. Suppose the time series $S_1, \ldots, S_M$ represents consecutive quotations of a stock price. To test the data, histograms of the returns are helpful (⟶ Fig. 1.23). The transformation $y = \log(S)$ is most practical. It leads to the notion of the *log return*, defined by[17]

$$R_{i,i-1} := \log \frac{S_i}{S_{i-1}}\,. \tag{1.70}$$

[17] Since $S_i = S_{i-1}\exp(R_{i,i-1})$, the log return is also called *continuously compounded return* in the ith time interval [365].

Let Δt be the equally spaced sampling time interval between the quotations S_{i-1} and S_i, measured in years. Then (1.64) leads to

$$R_{i,i-1} \sim \mathcal{N}((\mu - \frac{\sigma^2}{2})\Delta t\ ,\ \sigma^2 \Delta t)\ .$$

Comparing with (1.49) we realize that the variances of the simple return and of the log return are identical. The sample variance $\sigma^2 \Delta t$ of the data allows to calculate estimates of the historical volatility σ ($\longrightarrow$ Exercise 1.23). But the shape of actual market histograms is usually not in good agreement with the well-known bell shape of the Gaussian density. The symmetry may be perturbed, and in particular the tails of the data are not well modeled by the hypothesis of a geometric Brownian motion: The exponential decay expressed by (1.64) amounts to *thin tails*. This underestimates extreme events and hence hardly matches the reality of stock prices.

We conclude this section by listing the analytic solution of the basic linear constant-coefficient SDE (1.47)

$$\mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t$$

of GBM. From (1.63) or (1.65), the process

$$S_t := S_0 \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right) \tag{1.71}$$

solves the linear constant-coefficient SDE (1.47). Equation (1.71) generalizes to the case of nonconstant coefficients ($\longrightarrow$ Exercise 1.24). As a consequence we note that $S_t > 0$ for all t, provided $S_0 > 0$.

1.9 Jump Models

The geometric Brownian motion Model 1.13 has continuous paths S_t. As noted before, the continuity is at variance with those rapid asset price movements that can be considered almost instantaneous. Such rapid changes can be modeled as jumps. This section introduces a basic building block of a jump process, namely, the Poisson process. Related simulations (like that of Fig. 1.24) may look more authentic than continuous paths. But one has to pay a price: With a jump process the risk of an option in general can not be hedged away to zero. And calibration becomes more involved.

1.9.1 Poisson Process

To define a Poisson process, denote the time instances for which a jump arrives τ_j, with

$$\tau_1 < \tau_2 < \tau_3 < \ldots$$

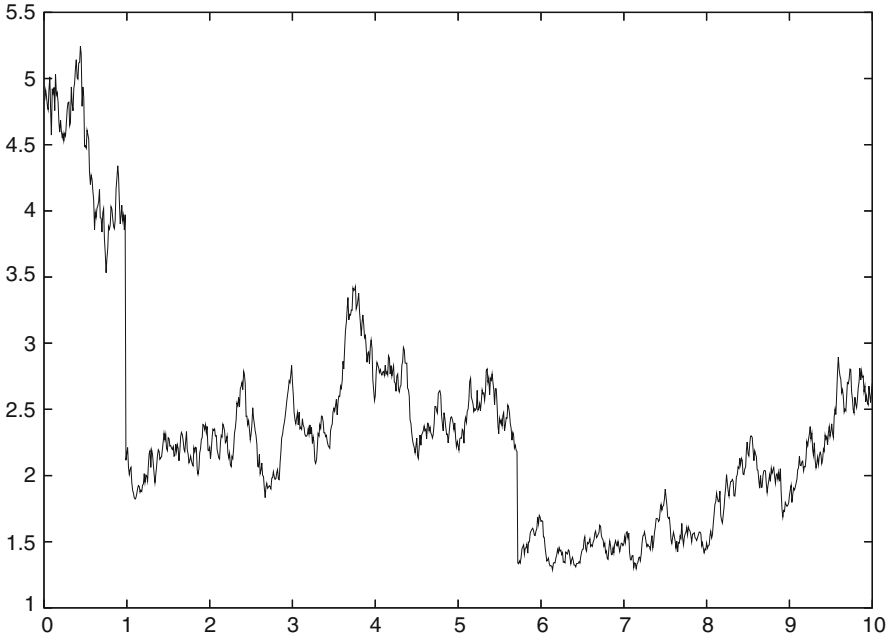

Fig. 1.24 Example 1.21: sample path S_t of (1.74); jump report in Table 1.3

Let the number of jumps be counted by the counting variable J_t, where

$$\tau_j = \inf\{t \geq 0\ ,\ J_t = j\}\,.$$

A Bernoulli experiment describes the probability that a jump occurs. For this local discussion and an arbitrary time instant t, consider n subintervals of length $\Delta t := \frac{t}{n}$ and allow for only two outcomes, jump *yes* or *no*, with the probabilities

$$\begin{aligned} \mathsf{P}(J_t - J_{t-\Delta t} = 1) &= \lambda \Delta t \\ \mathsf{P}(J_t - J_{t-\Delta t} = 0) &= 1 - \lambda \Delta t \end{aligned} \tag{1.72}$$

for some λ such that $0 < \lambda \Delta t < 1$. The parameter λ is referred to as the *intensity* of this jump process. Consequently k jumps in $0 \leq \tau \leq t$ have the probability

$$\mathsf{P}(J_t - J_0 = k) = \binom{n}{k} (\lambda \Delta t)^k (1 - \lambda \Delta t)^{n-k}\,,$$

where the trials in each subinterval are considered independent. A little reasoning reveals that for $n \to \infty$ this probability converges to

$$\frac{(\lambda t)^k}{k!} \mathrm{e}^{-\lambda t},$$

which is known as the Poisson distribution with parameter $\lambda > 0$ ($\longrightarrow$ Appendix B.1). This leads to the Poisson process.

Definition 1.19 (Poisson Process) The stochastic process $\{J_t\ ,\ t \geq 0\}$ is called Poisson process if the following conditions hold:

(a) $J_0 = 0\,.$
(b) $J_t - J_s$ are integer-valued for $0 \leq s < t < \infty$ and

$$\mathsf{P}(J_t - J_s = k) = \frac{\lambda^k (t-s)^k}{k!}\, \mathrm{e}^{-\lambda(t-s)} \text{ for } k = 0, 1, 2 \ldots$$

(c) The increments $J_{t_2} - J_{t_1}$ and $J_{t_4} - J_{t_3}$ are independent for all $0 \leq t_1 < t_2 < t_3 < t_4$.

Several properties hold as consequence of this definition:

Properties 1.20 (Poisson Process)

(d) J_t is right-continuous and nondecreasing.
(e) The times between successive jumps are independent and exponentially distributed with parameter λ. Thus,

$$\mathsf{P}(\tau_{j+1} - \tau_j > \Delta\tau) = \mathrm{e}^{-\lambda \Delta\tau} \quad \textit{for each } \Delta\tau\,.$$

(f) J_t is a Markov process.
(g) $\mathsf{E}(J_t) = \lambda t\,,\ \mathsf{Var}(J_t) = \lambda t\,.$

1.9.2 Simulating Jumps

Following the above introduction of Poisson processes, there are two possibilities to calculate jump instances τ_j such that the above probabilities are met. First, the Eq. (1.72) may be used together with uniform deviates ($\longrightarrow$ Chap. 2). In this way a Δt-discretization of a t-grid can be easily exploited by drawing a random number to decide whether a jump occurs in a subinterval. The other alternative is to calculate exponentially distributed random numbers $h_1, h_2, \ldots$ ($\longrightarrow$ Sect. 2.2.2) to simulate the intervals $\Delta\tau$ between consecutive jump instances, and set

$$\tau_{j+1} := \tau_j + h_j\,.$$

The expectation of the h_j is $\frac{1}{\lambda}$.

The unit amplitudes of the jumps of the Poisson counting process J_t are not relevant for the purpose of establishing a market model. The jump *sizes* of the price of a financial asset should be considered random. This requires—in addition to the arrival times τ_j—another random variable.

Let the random variable S_t jump at τ_j, and denote τ^+ the (infinitesimal) instant immediately after the jump, and τ^- the moment before. Then the absolute size of the jump is

$$\Delta S = S_{\tau+} - S_{\tau-} ,$$

which we model as a *proportional jump*,

$$S_{\tau+} = qS_{\tau-} \quad \text{with } q > 0 . \tag{1.73}$$

So, $\Delta S = qS_{\tau-} - S_{\tau-} = (q-1)S_{\tau-}$. The jump sizes equal $q-1$ times the current asset price. Accordingly, this model of a jump process depends on a random variable q_t and is written

$$\mathrm{d}S_t = (q_t - 1)S_{t-}\,\mathrm{d}J_t , \text{ where } J_t \text{ is a Poisson process.}$$

We assume that $q_{\tau_1}, q_{\tau_2}, \ldots$ are i.i.d. The resulting process with the two involved processes J_t, q_t is called **compound Poisson process**.

1.9.3 Jump Diffusion

Next we superimpose the jump process to stochastic diffusion, here to GBM. The combined geometric Brownian and compound Poisson process is given by

$$\mathrm{d}S_t = S_{t-}\,(\,\mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t + (q_t - 1)\,\mathrm{d}J_t\,) . \tag{1.74}$$

Here σ is the same as for the GBM, hence conditional on no jump. Such a combined model represented by (1.74) is called **jump-diffusion process**. It involves three different stochastic driving processes, namely, W_t, J_t, and q_t. We assume that J, q, W are independent of one another. Figure 1.24 shows a simulation of the SDE (1.74).

An analytic solution of (1.74) can be calculated on each of the jump-free subintervals $\tau_j < t < \tau_{j+1}$ where the SDE is just the GBM diffusion $\mathrm{d}S = S(\mu\mathrm{d}t + \sigma\mathrm{d}W)$. For example, in the first subinterval until τ_1, the solution is given by (1.71). At τ_1 a jump of the size

$$(\Delta S)_1 := (q_{\tau_1} - 1)S_{\tau_1^-}$$

occurs, and thereafter the solution continues with

$$S_t = S_0 \cdot \exp\left(\left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right) + (q_{\tau_1} - 1)S_{\tau_1^-} ,$$

until τ_2. The interchange of continuous parts and jumps proceeds in this way, all jumps are added. So the SDE can be written as

$$S_t = S_0 + \int_0^t S_s(\mu \mathrm{d}s + \sigma \mathrm{d}W_s) + \sum_{j=1}^{J_t} S_{\tau_j -}(q_{\tau_j} - 1)\,, \tag{1.75}$$

or

$$S_t = S_0 \exp\left(\left((\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t\right) \cdot \prod_{j=1}^{J_t} q_j\,.$$

This is the model based on Merton's paper [270]. The Eq. (1.75) can be rewritten in the log-framework, with $Y_t := \log S_t$. The log-jump sizes according to model (1.73) are

$$\begin{aligned}(\Delta Y)_\tau :&= Y_{\tau+} - Y_{\tau-} = \log(qS_{\tau-}) - \log S_{\tau-} \\ &= \log q_\tau\,.\end{aligned}$$

Following (1.71), the model can be written

$$Y_t = Y_0 + \left(\mu - \frac{\sigma^2}{2}\right)t + \sigma W_t + \sum_{j=1}^{J_t} (\Delta Y)_{\tau_j} \tag{1.76}$$

—that is the sum of a drift term, a Brownian motion, and a jump process. The summation term $\sum(\Delta Y)$ in (1.76) is the compound process. Merton assumes normally distributed ΔY, which amounts to lognormal q. In summary we emphasize again that the jump-diffusion process has three driving processes, namely, W, J, and q. As in the GBM case, see (1.65)/(1.71), the price process is of the form $S_t = S_0 \exp(Y_t)$.

Example 1.21 (Jump Diffusion) Here we assume an interest rate $r = 0.06$, and a process S_t following (1.74) with diffusion volatility $\sigma = 0.3$. For a hypothetical crash modeling, let us assume Poisson jumps with an intensity rate $\lambda = 0.2$, which means that on the average one jump occurs every 5 years. Following Merton's model, we take $\log(q) \sim \mathcal{N}(\mu_\mathrm{J}, \sigma_\mathrm{J}^2)$, and choose $\mu_\mathrm{J} = -0.3$ and $\sigma_\mathrm{J} = 0.4$. To get random numbers with distribution $\sim \mathcal{N}(\mu_\mathrm{J}, \sigma_\mathrm{J}^2)$, we calculate random numbers $Z \sim \mathcal{N}(0,1)$ (Chap. 2), and set $\log q = \sigma_\mathrm{J} Z + \mu_\mathrm{J}$. The chosen value of μ_J corresponds to a mean $q = \exp(\mu_\mathrm{J}) = 0.7408$, which amounts to an average 26% drop in S_τ at a jump instant τ. For the integration of (1.74), a growth rate is chosen such that risk neutrality is achieved. As will be explained in Sect. 7.3, the martingale property is satisfied with

$$\mu = r - \lambda\,(\exp[\mu_\mathrm{J} + \tfrac{1}{2}\sigma_\mathrm{J}^2] - 1)\,,$$

Table 1.3 Jumps in Fig. 1.24

τ	$\log(q)$	q	Jump
0.99	−0.642	0.526	47% down
4.76	0.0495	1.05	5% up
5.72	−0.534	0.586	41% down

which for our numbers gives the growth rate 0.0995. This rate μ is larger than r, and—roughly speaking—compensates for the tendency that in case $\mu_J < 0$ down jumps are more likely than up jumps. Now we are ready to solve (1.74) numerically. Figure 1.24 shows one calculated trajectory. We see three jumps, with data in Table 1.3. In this particular simulation, there are two heavy down jumps within the time interval $0 \le t \le 10$, which are clearly visible in Fig. 1.24.

The task of valuing options leads to a partial integro-differential equation (A.20), shown in Appendix A.4, and in Sect. 7.3.

The above jump-diffusion process is not the only jump process used in finance. There are also processes with an infinite number of jumps in finite time intervals. To model such processes, building blocks are provided by a more general class of jump processes, namely, the Lévy processes. Simply speaking, think of relaxing the properties (b), (d) of Definition 1.7 of the Wiener process such that non-normal distributions and jumps are permitted. Consult Sect. 7.3 for some basics on Lévy processes.

1.10 Calibration

Which model should be chosen for a particular application?

This is a truly fundamental question. The question involves two views, namely, a qualitative and a quantitative aspect.

When one speaks of a "model," the focus is on its quality. This refers to the structure and the type of equation. Important ingredients of a model are, for example, a diffusion term, a jump feature, a specific nonlinearity, or whether the volatility is considered as a constant or a stochastic process. Ideally, the model and its equations represent economical laws. On the other hand, the quantitative aspect of the model consists in the choice of specific numbers for the coefficients or parameters of the model. "Modeling" refers to the setup of a chosen equation, and "calibration" is the process of matching the parameters of the chosen model to the data that represent reality.

The distinction between modeling and calibration is not always obvious. For example, consider the class of mean-reversion models represented by (1.55). There is the exponent β in the factor r_t^β. This exponent β can be regarded either as parameter, or as a structural element of the model. The three cases

$$\beta = 0: \quad \text{the factor is unity, } r^\beta = 1\,, \text{ it "disappears,"}$$

$$\beta = 1: \quad \text{the factor is linear, it represents a proportionality,}$$
$$\beta = 1/2: \quad \text{the factor } \sqrt{r} \text{ is a specific nonlinearity,}$$

point at the qualitative aspect of this specific parameter. Typically, modeling sets forth some argument why a certain parameter is preset in a specific way, and not subjected to calibration. Modeling places emphasis on capturing market behavior rather than the peculiarities of a given data set.

Let us denote N parameters to be calibrated by $c_1, \ldots, c_N$. Examples are the volatility σ in GBM (1.47), or α, R for the mean-reversion term in (1.55), or the jump intensity λ of a jump-diffusion process. For the mean-reverting volatility tandem of Example 1.15, the vector to be calibrated consists of five parameters,

$$c = (\alpha, \beta, \rho, \sigma_0, \zeta_0)\,.$$

Here ρ is the correlation between the two Wiener processes $W^{(1)}, W^{(2)}$, and σ_0, ζ_0 are the initial values for the processes σ_t, ζ_t. For the volatility tandem it makes sense to assume $\zeta_0 = \sigma_0$, which cuts down the calibration dimension N from five to four. The initial stock price S_0 is known. The interest rates r that match a maturity T are obtained, for example, from EURIBOR, and are not object of the calibration. Any attempt to cut down the calibration dimension N is welcome because the costs of calibration are significant.

Suppose an initial guess of the calibration vector c. Then the calibration procedure is based on the three steps

(1) simulate the model—that is, solve it numerically,
(2) compare the calculated results with the market data—that is, calculate the defect, and
(3) adapt c such that the model better matches the data—that is, the defect should decrease.

These three steps are repeated iteratively. How to perform step (3) is not obvious; there is no unique way how to decrease the defect. A standard approach is to minimize the defect in a least-squares fashion.

In our context of calibrating models for finance, data of vanilla options are available as follows: The price S of the underlying is known as well as market prices V^{mar} for several strikes K and maturities T. Let the option prices V^{mar} be observed for M pairs $(T_1, K_1), \ldots, (T_M, K_M)$. That is, the available data are

$$S,\ (T_k, K_k, V_k^{\text{mar}})\,, \quad k = 1, \ldots, M\,.$$

For definiteness of the calibration require sufficiently many data in the sense $M \geq N$. Raw data may be subjected to a smoothing process [154].

First, a model is specified. Then, in step (1), the chosen model is evaluated for each of the M data (S, T_k, K_k), which gives model prices $V(S, 0; T_k, K_k; c)$. In general, this valuation process is expensive. An excellent approach for the

simultaneous valuation of a large number of European options is the FFT method of Carr and Madan [68], see Sect. 7.4. In step (2), the result of the valuation is compared to the market prices. There will be a defect. Therefore, in step (3), an iteration is set up to improve the current fit c. The least-squares approach is to minimize the sum of the squares of all defects, over all c,

$$\min_c \sum_{k=1}^{M} (V_k^{\text{mar}} - V(S,0;T_k,K_k;c))^2 \,. \tag{1.77}$$

The sum in (1.77) is a function of c and can be visualized as a surface over the parameter c-space. It can be modified by weighting the terms appropriately. Finally, the calibration results in a minimizing c ($\longrightarrow$ Appendix C.4). In view of the data error, it hardly makes sense to calculate the minimizing parameter vector c with high accuracy.

A simple example is provided by the implied volatility, see Exercise 1.19. Here $N = 1, M = 1, c := \sigma$, and it is possible to make the defect vanish—the minimum in (1.77) becomes zero. But in general the minimum of (1.77) will be a positive value. It is tempting to regard this value as a measure of the defect of the chosen model. But this would be misleading; we come back to this below.

As a numerical example, we calibrate two models on the same data set of standard European calls on the DAX index observed in the time period January 2002 through September 2005. For this example, the calibration of Heston's model (1.59) results in the five parameters

$$\kappa = 1.63\,,\ \theta = 0.0934\,,\ \sigma_{\text{v}} = 0.473\,,\ v_0 = 0.0821\,,\ \rho = -0.8021\,,$$

with $\mu = r$ for the risk-neutrality. This parameter set matches the criterion $2\kappa\theta \geq \sigma_{\text{v}}$ which guarantees $v > 0$. The same data are applied to calibrate the Black–Scholes model: The data are matched by GBM with the constant $\sigma = 0.239$ (from [122]). This is comparable to the calibration of the Heston model with its $\sqrt{v_0} \approx 0.28$.

So far, we have not come close to an answer to the initial question on the "best" choice of an appropriate model. An attempt to decide on the quality of a model might be to compare the defects. For instance, compare the values of the sums in (1.77). In the above experiment, Heston's model has the smaller defect; the defect of the Black–Scholes model is five times as large.

One might think that one model is better than another one, when the defect is smaller. But this is a wrong conclusion! Admitting a large enough number of parameters enables to reach a seemingly best fit with a small defect. The danger with a large number of parameters is *overfitting*. Overfitting can be detected as follows: Divide the data into halves, fit the model on the one half (*in-sample fit*), and then test the quality of the fit on the other half of the data (*out-of-sample fit*). In case the out-of-sample fit matches the data much worse than the in-sample fit, we have a strong clue on overfitting. Then any predictive power of the model may be lost. A vanishing defect might be seen as hint of the model being useless. Overfitting is related to the

stability of parameters. If the parameters c change drastically when exchanging one data set by a similar data set, then the model is considered unstable. In order to obtain information on the parameter uncertainty, the defect must be analyzed more closely around the calculated best fit c. The defect function (1.77) can exhibit a large flat region. Then significantly different values of c yield a similar error. In this sense, a calibration problem can be ill-posed [173].

There is another test of the quality of a model, namely, how well hedging works. A hedging strategy based on the model is compared to the reality of the data. Empirical tests and comparisons in [94, 122] suggest that in the context of option pricing, a stochastic volatility may be a more basic ingredient of a good model than jump processes are. In terms of stability, out-of-sample fitting, and hedging of options, Heston's model (Example 1.16) is recommendable—these conclusions have been based on the prices of European options on the DAX 2002–2005. In terms of hedging capabilities, the classic Black–Scholes model is competitive.

To summarize, it is obvious that calibration is a formidable task, in particular if several parameters are to be fitted. The attainable level of calibration quality depends on the chosen model. In case the structure of the equation is not designed properly, an attempt to improve parameters may be futile. For a given model, it might well happen that a perfect calibration is never found. It is unlikely that some model eventually might emerge as generally "most recommendable." Calibration does not remove the risk of having chosen the wrong model. With our focus on computational tools, it does make sense to consider the Black–Scholes model as a benchmark. It captures a significant part of the essence of option markets.

1.11 Notes and Comments

On Sect. 1.1

This section presents a brief introduction to standard options. For more comprehensive studies of financial derivatives we refer, for example, to [88, 191, 376]. Mathematical detail can be found in [123, 217, 237, 282, 339, 345]. Other books on financial markets include [97, 120, 146, 268]. (All hints on the literature are examples; an extensive overview on the many good books in this rapidly developing field is hardly possible.)

On Sect. 1.2

Black, Merton and Scholes developed their approaches concurrently, with basic papers in 1973 ([41], [269]; compare also [271]). Merton and Scholes were awarded the Nobel Prize in economics in 1997. (Black had died in 1995.) One of the results

of these authors is the so-called Black–Scholes equation (1.5) with its analytic solution formulas (1.7)–(1.10). For reference on discrete-time models, consult [132, 302]. Transaction costs and market illiquidity or feedback effects are discussed in Sect. 7.1.

On Sect. 1.3

References on specific numerical methods are given where appropriate. As computational finance is concerned, most quotations refer to research papers. Other general text books discussing computational issues include [2, 181, 376]; further hints can be found in [321]. For the calculation of the sample variance (Exercise 1.6) see [72, 183].

On Sect. 1.4

Binomial or trinomial methods are sometimes found under the heading *tree methods* or *lattice methods*. Basic versions of the binomial method were introduced in 1979 by [87][18] and [313]. Cox et al. [87] suggested

$$\tilde{u} := \mathrm{e}^{\sigma\sqrt{\Delta t}},\ \tilde{d} := \mathrm{e}^{-\sigma\sqrt{\Delta t}},\ \tilde{p} := \frac{1}{2}\left(1 + \frac{r - \frac{1}{2}\sigma^2}{\sigma}\sqrt{\Delta t}\right), \tag{1.78}$$

where $\tilde{p}$ is a first-order approximation to the p of (1.13). (The reader may check; for $\tilde{u}, \tilde{d}$ see Exercise 1.8.) The influential paper by Cox, Ross and Rubinstein has coined the name CRR for their approach. Hull and White [192] pointed out that (1.18) is slightly more correct than the CRR choice. Rendleman and Bartter [313] suggested the choice $p = \frac{1}{2}$, which leads to values of u and d ($\longrightarrow$ Exercise 1.26). Of course, another set of parameters u, d, p leads to a different approximation. Example 1.6, which is from [191], and $M = 100$ yields $V = 4.28041$ with the parameter set (1.18), and $V = 4.27806$ with u, d from (1.78). But for $M \to \infty$ convergence is maintained in either case. The dynamic programming principle is due to [33]: Of each optimal path, any piece (subpath) must be optimal too. In the literature, the result of the dynamic programming procedure is often listed under the name Snell envelope.

Table 1.2 might suggest that it is easy to obtain high accuracy with binomial methods. This is not the case; flaws were observed in particular close to the early-exercise curve [83]. As illustrated by Fig. 1.10, the described standard version

[18] William Sharpe has been credited for suggesting the advantages of the discrete-time approach.

wastes many nodes $S_{j,i}$ close to zero and far away from the strike region even for small M.

For advanced binomial methods and for speeding up convergence, consult also [51, 224, 243, 244, 359]. Figlewski and Gao [129] inserts a patch of higher resolution close to $(S, t) = (K, T)$ into the trinomial tree. The resulting adaptive mesh model exhibits higher accuracy. In order to maintain accuracy for barrier options one takes care that layers coincide with the barrier, see for instance [96]. For a detailed account of the binomial method see also [88]. By correcting the terminal probabilities, which come out of the binomial distribution ($\longrightarrow$ Exercise 1.10), it is possible to adjust the tree to actual market data [322], see also the *implied tree* of [100], outlined in Appendix D.5 or [337]. Honoré and Poulsen [189] explains how to implement the binomial method in spreadsheets. Many applications of binomial trees are found in [251].

On Sect. 1.5

When we expect Δ to be positive, then we should assume the option is a call. But the argumentation is the same for a put, then $\Delta < 0$. As shown in Sect. 1.5, a valuation of options based on a hedging strategy is equivalent to the risk-neutral valuation described in Sect. 1.4. Another equivalent valuation is obtained by a *replication* portfolio. This basically amounts to including the risk-free investment, to which the hedged portfolio of Sect. 1.5 was compared, into the portfolio. To this end, the replication portfolio includes a bond with the initial value $B_0 := -(\Delta \cdot S_0 - V_0) = -\Pi_0$ and interest rate r. The portfolio consists of the bond and Δ shares of the asset. At the end of the period T the final value of the portfolio is $\Delta \cdot S_T + \mathrm{e}^{rT}(V_0 - \Delta \cdot S_0)$. The hedge parameter Δ and V_0 are determined such that the value of the portfolio is V_T, independent of the price evolution. By adjusting B_0 and Δ in the right proportion we are able to replicate the option position. This strategy is *self-financing*: No initial net investment is required. The result of the self-financing strategy with the replicating portfolio is the same as what was derived in Sect. 1.5. The reader may like to check this. For the continuous-time case, see Appendix A.4.

Frequently discounting is done with the factor $(1 + r \cdot \Delta t)^{-1}$. This r would not be a continuously compounding interest rate. Our $\mathrm{e}^{-r\Delta t}$ or e^{-rT} is consistent with the approach of Black, Merton and Scholes. For references on risk-neutral valuation we mention [191, 234, 282], and [340].

On Sect. 1.6

Introductions into stochastic processes and further hints on advanced literature can be found in [12, 37, 108, 139, 225, 314, 328, 339, 340]. In the literature, the terms Wiener process and Brownian motion are often used as synonyms, and the

modifier "standard" is used to specialize on the drift-free case. Here we follow the convention as in Definition 1.7, where the term Wiener process is mostly reserved for the "standard" scalar drift-free Brownian motion. The definition of a Wiener process depends on the underlying probability measure P, which enters through the definition of independence, and by its distribution being Gaussian, see (B.1). For more hints on martingales, see Appendix B.2. Algorithm 1.8 is also called "Gaussian random walk."

For a proof of the nondifferentiability of Wiener processes, see [193]. In contrast to the results for Wiener processes, differentiable functions W_t satisfy for $\delta_N \to 0$

$$\sum |W_{t_j} - W_{t_{j-1}}| \longrightarrow \int |W'_s| \, \mathrm{d}s \,, \quad \sum (W_{t_j} - W_{t_{j-1}})^2 \longrightarrow 0 \,.$$

The Itô integral and the alternative Stratonovich integral are explained in [12, 78, 108, 216, 225, 274, 291, 314, 333, 340]. The class of Itô-stochastically integrable functions is characterized by the properties $f(t)$ is $\mathcal{F}_t$-adapted and $\mathsf{E} \int f(s)^2 \mathrm{d}s < \infty$. We assume that all integrals occurring in the text exist. The integrator W_t needs not be a Wiener process. The stochastic integral can be extended to semimartingales [193].

On Sect. 1.7

The Algorithm 1.11 is sometimes named after Euler and Maruyama.

A general linear SDE is of the form

$$\mathrm{d}X_t = (a_1(t)X_t + a_2(t)) \, \mathrm{d}t + (b_1(t)X_t + b_2(t)) \, \mathrm{d}W_t \,.$$

The expectation $\mathsf{E}(X_t)$ of a solution process X_t of a linear SDE satisfies the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{E}(X_t) = a_1 \mathsf{E}(X_t) + a_2 \,,$$

and for $\mathsf{E}(X_t^2)$ we have

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathsf{E}(X_t^2) = (2a_1 + b_1^2)\mathsf{E}(X_t^2) + 2(a_2 + b_1 b_2)\mathsf{E}(X_t) + b_2^2 \,.$$

This is obtained by taking the expectation of the SDEs for X_t and X_t^2, the latter one derived by Itô's lemma [225, 274]. Combining both differential equations allows to calculate the variance. Kloeden and Platen [225] gives a list of SDEs that are analytically solvable or reducible.

A process (1.47) with variable $\mu(t), \sigma(t)$ is called generalized GBM [340]. For CIR of Example 1.16, provided $r_0 > 0$, $R > 0$, and a strong enough upward drift in the sense

$$\alpha R \geq \frac{1}{2}\sigma_{\mathrm{r}}^2 ,$$

the solution of (1.55) satisfies $r_t > 0$ for all t; this criterion is attributed to Feller. For a PDE, the Feller condition is replaced by a boundary condition at $r = 0$ [117]. Based on the CIR system and a dependent variable $u(S, v, t)$ a two-dimensional PDE is presented in [178], see Example 5.8.

The model of a geometric Brownian motion of Eq. (1.47) is the classic model describing the dynamics of stock prices. It goes back to Samuelson (1965; Nobel Prize in economics in 1970). Already in 1900 Bachelier had suggested to model stock prices with Brownian motion. Bachelier used the arithmetic version, which can be characterized by replacing the left-hand side of (1.47) by the absolute change dS. This amounts to the process of the drifting Brownian motion $S_t = S_0 + \mu t + \sigma W_t$. Here even the theoretical stock price can become negative. Main advantages of the geometric Brownian motion are its exponential growth or decay, the success of the approaches of Black, Merton and Scholes, which is based on that motion, and the existence of moments (as the expectation). For positive S, the form (1.47) of GBM is not as restrictive as it might seem, see Exercise 1.24. A variable volatility $\sigma(S, t)$ is called *local volatility*. Such a volatility can be used to make the Black–Scholes model compatible with observed market prices [112] (⟶ Appendix A.6).

On Sect. 1.8

The Itô lemma is also called Doeblin-Itô formula, after the early manuscript [107] was disclosed. The transformation that leads to the density f_{GBM} in (1.64) will be stated formally in Sect. 2.2. The Algorithm 1.18 was suggested by [309], including the use of radial basis functions, a tricky control of truncation errors, and a convergence analysis. The approximation quality of American options is quite satisfactory even for small values of M.

In view of their continuity, GBM processes are not appropriate to model jumps, which are characteristic for the evolution of stock prices. Jumps lead to relatively *heavy tails* in the distribution of empirical returns (see Fig. 1.23).[19] As already mentioned, the tails of the lognormal distribution are too thin. Other distributions match empirical data better. One example is the Pareto distribution, which

[19]The thickness is measured by the *kurtosis* $\mathsf{E}((X-\mu)^4)/\sigma^4$. The normal distribution has kurtosis 3. So the *excess kurtosis* is the difference to 3. Frequently, data of returns are characterized by large values of excess kurtosis.

has tails behaving like $x^{-\alpha}$ for large x and a constant $\alpha > 0$. A correct modeling of the tails is an integral basis for *value at risk* (VaR) calculations. For the risk aspect consult [14, 25, 109, 121], and the survey [113]. For distributions that match empirical data see [43, 49, 114, 254, 339]. Estimates of future values of the volatility are obtained by (G)ARCH methods, which work with different weights of the returns [138, 191, 325, 339, 365]. Promising are models of behavioral finance that consider the market as *dynamical system* [38, 39, 60, 76, 106, 249, 258, 344]. These systems experience the nonlinear phenomena *bifurcation* and *chaos*, which require again numerical methods. Such methods exist, and are explained elsewhere [336].

On Sect. 1.9

Section 1.9 concentrates on Merton's jump-diffusion process. For building Lévy models we refer to [84, 328], and Sect. 7.3.

On Sect. 1.10

The CIR-based Heston model can be extended to jump-diffusion. This can be applied to both processes S_t and v_t in (1.59), which defines a general class of models with 10 parameters [111]. But applying jumps only for S_t, one obtains the same quality with eight parameters [30]. Also the Ornstein-Uhlenbeck-based Schöbel–Zhu model is recommendable [329]. Another FFT based valuation approach is [125]. Artificial smoothing of the least-squares function (1.77) allows to apply gradient-based methods. This is discussed in [209]. For hedging issues and practical aspects, consult [207].

1.12 Exercises

1.1 (Put-Call Parity)
Consider a portfolio consisting of three positions related to the same asset, namely, one share (price S), one European put (value V_P), plus a short position of one European call (value V_C). Put and call have the same expiration date T, and no dividends are paid.

(a) Assume a no-arbitrage market without transaction costs. Show

$$S + V_\mathrm{P} - V_\mathrm{C} = K\mathrm{e}^{-r(T-t)}$$

for all t, where K is the strike and r the risk-free interest rate.

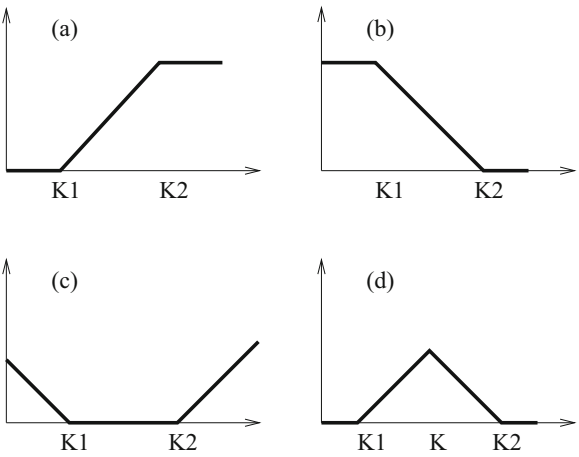

Fig. 1.25 Four payoffs, value over S; see Exercise 1.3

(b) Use this put-call parity to realize

$$V_{\mathrm{C}}(S,t) \geq S - K\mathrm{e}^{-r(T-t)}\,,$$
$$V_{\mathrm{P}}(S,t) \geq K\mathrm{e}^{-r(T-t)} - S\,.$$

1.2 (Bounds and Arbitrage)
Using arbitrage arguments, show the following bounds for the values V_{C} of vanilla call options:

(a) $0 \leq V_{\mathrm{C}}$
(b) $(S-K)^+ \leq V_{\mathrm{C}}^{\mathrm{Am}} \leq S$

1.3 (Portfolios)
Figure 1.25 sketches some payoffs over S: (a) bull spread, (b) bear spread, (c) strangle, (d) butterfly spread. For each of these payoffs, construct portfolios out of two or three vanilla options (same expiry, same underlying) such that the portfolio meets the payoff.

1.4 (Transforming the Black–Scholes Equation)
Show that the Black–Scholes equation (1.5)

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

for $V(S,t)$ with constant σ and r is equivalent to the equation

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2}$$

for $y(x,\tau)$. For proving this, you may proceed as follows:

(a) Use the transformation $S = K\mathrm{e}^x$ and a suitable transformation $t \leftrightarrow \tau$ to show that (1.5) is equivalent to

$$-\dot{V} + V'' + \alpha V' + \beta V = 0$$

with $\dot{V} = \frac{\partial V}{\partial \tau}$, $V' = \frac{\partial V}{\partial x}$, α, β depending on r and σ.

(b) The next step is to apply a transformation of the type

$$V = K \exp(\gamma x + \delta \tau)\, y(x, \tau)$$

for suitable γ, δ.

(c) Transform the terminal condition of the Black–Scholes equation accordingly.

1.5 (Standard Normal Distribution Function)
Establish an algorithm to calculate

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp(-\frac{t^2}{2})\, \mathrm{d}t\,.$$

(a) Construct an algorithm to calculate the *error function*

$$\mathrm{erf}(x) := \frac{2}{\sqrt{\pi}} \int_0^x \exp(-t^2)\, \mathrm{d}t\,,$$

and use $\mathrm{erf}(x)$ to calculate $F(x)$. Use composite trapezoidal sums ($\longrightarrow$ Appendix C.1) to calculate approximations $F_n(x)$.

(b) Apply the approximation formula of Appendix E.2 to calculate an approximation $\tilde{F}(x)$, and set up a computer program.

(c) Full accuracy for comparison can be obtained by the generic code `derf`. For a series of values x evaluate the errors of the above algorithms. Enter the computing times and the errors of $\tilde{F}$ and F_n for several n into a diagram similar as Fig. 4.19.

1.6 (Calculating the Sample Variance)
An estimate of the variance of M numbers $x_1, \ldots, x_M$ is

$$s_M^2 := \frac{1}{M-1} \sum_{i=1}^{M} (x_i - \bar{x})^2, \quad \text{with } \bar{x} := \frac{1}{M} \sum_{i=1}^{M} x_i\,.$$

The alternative formula

$$s_M^2 = \frac{1}{M-1} \left(\sum_{i=1}^{M} x_i^2 - \frac{1}{M} \left(\sum_{i=1}^{M} x_i \right)^2 \right) \qquad (\diamond)$$

can be evaluated with only one loop $i = 1, \ldots, M$, but should be avoided because of the danger of cancellation. The following single-loop algorithm is recommended instead of ($\diamond$):

$\alpha_1 := x_1,\ \beta_1 := 0$
for $i = 2, \ldots, M$:
$\alpha_i := \alpha_{i-1} + \frac{x_i - \alpha_{i-1}}{i}$
$\beta_i := \beta_{i-1} + \frac{(i-1)(x_i - \alpha_{i-1})^2}{i}$

(a) Show $\bar{x} = \alpha_M,\ s_M^2 = \frac{\beta_M}{M-1}$.
(b) For the ith *update* in the algorithm carry out a rounding error analysis. What is your judgment on the algorithm?

1.7 (Anchoring the Binomial Grid at K)
The equation $ud = \gamma$ (1.17) anchors the tree. One may anchor the grid by requiring (for even M)

$$S_0 u^{M/2} d^{M/2} = K\,.$$

(a) Give a geometrical interpretation.
(b) Derive from Eqs. (1.12), (1.16) and $ud = \gamma$ for some constant γ the relation

$$u = \beta + \sqrt{\beta^2 - \gamma} \quad \text{for} \quad \beta := \frac{1}{2}(\gamma \mathrm{e}^{-r\Delta t} + \mathrm{e}^{(r+\sigma^2)\Delta t})\,.$$

(c) Show that the solution is given by

$$ud = \gamma := \exp\left[\frac{2}{M} \log \frac{K}{S_0}\right].$$

(d) Show $d < \mathrm{e}^{r\Delta t} < u$.

1.8 (Price Evolution of the Binomial Method)
For β from (1.18) with $\gamma = 1$, and $u = \beta + \sqrt{\beta^2 - 1}$ analyze possible cancellation, and show

$$u = \exp\left(\sigma\sqrt{\Delta t}\right) + O\left(\sqrt{(\Delta t)^3}\right).$$

1.9 (Implementing a Binomial Method)
Design and implement an algorithm for calculating the value $V^{(M)}$ of a European or American option. Use the basic version of Algorithm 1.4.

INPUT: r (interest rate), σ (volatility), T (time to expiration in years), K (strike price), S (price of asset), and the choices *put* or *call*, and *European* or *American*, and an (initial) M.

Control the mesh size $\Delta t = T/M$ adaptively. A crude strategy would be, for example, to calculate V for $M = 8$ and $M = 16$ and in case of a significant change in V use $M = 32$ and possibly $M = 64$.

Test examples:

(a) put, European, $r = 0.06$, $\sigma = 0.3$, $T = 1$, $K = 10$, $S = 5$
(b) put, American, $S = 9$, otherwise as in (a)
(c) call, otherwise as in (a)
(d) The mesh size control must be done carefully and has little relevance to error control. To make this evident, calculate for the test numbers a) a sequence of $V^{(M)}$ values, say for $M = 100, 101, 102, \ldots, 150$, and plot the error $|V^{(M)} - 4.430465|$.

1.10 (Limiting Case of the Binomial Model)
Consider a European Call in the binomial model of Sect. 1.4. Suppose the calculated value is $V_0^{(M)}$. In the limit $M \to \infty$ the sequence $V_0^{(M)}$ converges to the value $V_C(S_0, 0)$ of the continuous Black–Scholes model given by (1.7)–(1.10) ($\longrightarrow$ Appendix A.4). To prove this, proceed as follows:

(a) Let j_K be the smallest index j with $S_{jM} \geq K$. Find an argument why

$$\sum_{j=j_K}^{M} \binom{M}{j} p^j (1-p)^{M-j} (S_0 u^j d^{M-j} - K)$$

is the expectation $\mathsf{E}(V_T)$ of the payoff. (For an illustration see Fig. 1.26.)

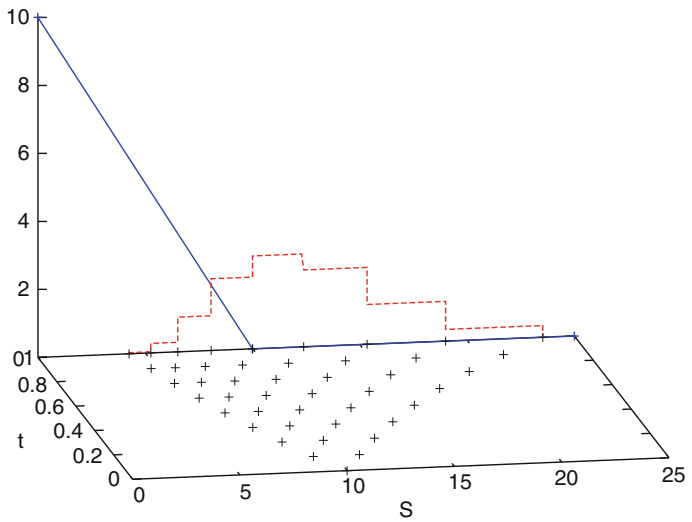

Fig. 1.26 Binomial tree in the (S, t)-plane and payoff (in *blue*) of a put; (S, t)-node points for $M = 8$, $K = S_0 = 10$. The binomial density of the risk-free probability is shown in *red dashed line*, scaled with factor 10. (Illustration for Exercise 1.10)

(b) The value of the option is obtained by discounting, $V_0^{(M)} = \mathrm{e}^{-rT}\mathsf{E}(V_T)$. Show

$$V_0^{(M)} = S_0 B_{M,\tilde{p}}(j_K) - \mathrm{e}^{-rT} K B_{M,p}(j_K)\,.$$

Here $B_{M,p}(j)$ is defined by the binomial distribution (⟶ Appendix B.1), and $\tilde{p} := pu\mathrm{e}^{-r\Delta t}$.

(c) For large M the binomial distribution is approximated by the normal distribution with distribution $F(x)$. Show that $V_0^{(M)}$ is approximated by

$$S_0 F\left(\frac{M\tilde{p}-\alpha}{\sqrt{M\tilde{p}(1-\tilde{p})}}\right) - \mathrm{e}^{-rT} K F\left(\frac{Mp-\alpha}{\sqrt{Mp(1-p)}}\right),$$

where

$$\alpha := -\frac{\log\frac{S_0}{K} + M\log d}{\log u - \log d}\,.$$

(d) Substitute the p, u, d by their expressions from (1.18) with $\gamma = 1$ to show

$$\frac{Mp-\alpha}{\sqrt{Mp(1-p)}} \longrightarrow \frac{\log\frac{S_0}{K} + (r-\frac{\sigma^2}{2})T}{\sigma\sqrt{T}}$$

for $M \to \infty$. *Hint:* Use Exercise 1.8: Up to terms of high order the approximations $u = \mathrm{e}^{\sigma\sqrt{\Delta t}}$, $d = \mathrm{e}^{-\sigma\sqrt{\Delta t}}$ hold. (In an analogous way the other argument of F can be analyzed.)

1.11 (Extrapolation)
Assume a (differential) equation, and let $\eta^* \in \mathbb{R}$ represent its exact solution. For a discretization, Δ denotes the grid size of the corresponding numerical approximation scheme, and $\eta(\Delta)$ the approximating solution. Further assume the error model

$$\eta(\Delta) - \eta^* = c\,\Delta^q,$$

with $c, q \in \mathbb{R}$. The exponent q is the *order* of the approximation scheme. Suppose that for two grid sizes Δ_1, Δ_2 with

$$\Delta_2 = \frac{1}{2}\Delta_1$$

approximations $\eta_1 := \eta(\Delta_1)$, $\eta_2 := \eta(\Delta_2)$ are calculated.

Assignment:

(a) For the case of a known η^* (or η^* approximated with very high accuracy) establish a formula for the order q out of η^*, η_1, η_2.

(b) For a known order q show that

$$\eta^* = \frac{1}{2^q - 1}(2^q \eta_2 - \eta_1)\,.$$

In general, the error model holds only approximately. Hence this formula for η^* is only an approximation of the exact η^* ("extrapolation").

1.12 (Smoothing a Function)
Assume a function Ψ, for example the payoff of a vanilla call $\Psi(S) := (S-K)^+$. Ψ can be approximated by the smoother $\overline{\Psi}$,

$$\overline{\Psi}(S) := \frac{1}{2\xi}\int_{-\xi}^{\xi} \Psi(S-y)\,\mathrm{d}y\,,$$

for a suitable chosen small $\xi > 0$.

(a) Calculate $\overline{\Psi}$ analytically for the payoff of a vanilla call, and a digital call.
(b) Set up an algorithm to calculate $\overline{\Psi}$ numerically for a given function Ψ. Use trapezoidal quadrature and program it on a computer; plot $\overline{\Psi}$.

1.13 (Greeks)
The Algorithm 1.4 with the anchoring of Exercise 1.7 is to be modified as follows:

> Extend the tree by starting at $-2\Delta t$ as discussed in Sect. 1.4.6, and calculate approximations for the greeks delta and gamma by using difference quotients.

Use Example 1.5 to compare these approximations with those from the analytic values from Appendix A.4. Implement this in a computer program.

1.14 (Dividend Payment and the Binomial Method)
A dividend yield δ can be calculated by annualizing a known dividend payment D per year by setting $\delta = D/S$. For a binomial tree, the effects of paying either

(a) a fixed amount D or
(b) a proportional amount δS

are different.

Assume a dividend payment at time $t_D < T$ and a node of the tree at $t_\nu = t_D$. For a share value of S at $t_{\nu-1}$ discuss the tree evolution at $t_{\nu+1}$ with focus on recombination, comparing the two scenarios (a) and (b).

1.15 (Trinomial Method)
Extend the classical binomial model to a trinomial model as follows: Allow for three prices S_{i+1} of the underlying at t_{i+1}, namely,

$$\begin{aligned} uS_i &\quad \text{with probability } p_1\,,\\ mS_i &\quad \text{with probability } p_2\,,\\ dS_i &\quad \text{with probability } p_3\,. \end{aligned}$$

For the six parameters u, m, d, p_1, p_2, p_3 six equations are needed. Clearly, the probabilities must be nonnegative, and $p_1 + p_2 + p_3 = 1$ must hold.

(a) Set up the two equations that equate expectation and variance with the corresponding values of the continuous model (similar as for the binomial model).
(b) The tree should be recombining. Cast this requirement into an equation.
(c) For the special choice of equal probabilities derive the parameters.
Hint: For

$$\alpha := \mathrm{e}^{r\Delta t}, \quad \beta := \mathrm{e}^{\sigma^2 \Delta t}, \quad \rho := \frac{\alpha}{4}(3 + \beta)$$

show

$$m = \frac{\alpha}{2}(3 - \beta), \quad u = \rho + \sqrt{\rho^2 - m^2}.$$

(d) Analyze possible cancellation in evaluating the expression $\sqrt{\rho^2 - m^2}$, and remove the cancellation.
(e) How many arithmetic operations are needed for the trinomial method with $\Delta t = T/M$ (without the calculation of u, m, d) ?

1.16 In Definition 1.7 the requirement (a) $W_0 = 0$ is dispensable. Then the requirement (b) reads

$$W_t - W_0 \sim \mathcal{N}(0, t).$$

Use these relations to deduce (1.32)/(1.33).
Hint: $(W_t - W_s)^2 = (W_t - W_0)^2 + (W_s - W_0)^2 - 2(W_t - W_0)(W_s - W_0)$

1.17

(a) Suppose that a random variable X_t satisfies $X_t \sim \mathcal{N}(0, \sigma^2)$. Use (B.4) to show

$$\mathsf{E}(X_t^4) = 3\sigma^4.$$

(b) Apply (a) to show the assertion in Lemma 1.9,

$$\mathsf{E}\left(\sum_j ((\Delta W_j)^2 - \Delta t_j)\right)^2 = 2\sum_j (\Delta t_j)^2.$$

1.18 (Negative Prices)
Assume $Z \sim \mathcal{N}(0, 1)$, $S > 0$, $\sigma > 0$, and a step $(t, S) \to (t + \Delta t, S + \Delta S)$ of the discretized GBM

$$\frac{\Delta S}{S} = \mu \Delta t + \sigma Z \sqrt{\Delta t}.$$

What is the probability that the resulting price $S + \Delta S$ is negative? Discuss the result in view of Algorithm 1.11.

1.19 (Implied Volatility)
For European options we take the valuation formula of Black and Scholes of the type $V = v(S, \tau, K, r, \sigma)$, where τ denotes the time to maturity, $\tau := T - t$. For the definition of the function v see the Eqs. (1.9) and (1.10), or Appendix A.4. If actual market data V^{mar} of the price are known, then one of the parameters considered known so far can be viewed as unknown and fixed via the implicit equation

$$V^{\text{mar}} - v(S, \tau, K, r, \sigma) = 0\,. \tag{1.79}$$

In this calibration approach the unknown parameter is calculated iteratively as solution of Eq. (1.79). Consider σ to be in the role of the unknown parameter. The volatility σ determined in this way is called *implied volatility* and is zero of $f(\sigma) := V^{\text{mar}} - v(S, \tau, K, r, \sigma)$.

Assignment:

(a) Implement the evaluation of V_{C} and V_{P} according to (1.9)/(1.10).
(b) Design, implement and test an algorithm to calculate the implied volatility of a call. Use Newton's method to construct a sequence $x_k \to \sigma$. The derivative $f'(x_k)$ can be approximated by the difference quotient

$$\frac{f(x_k) - f(x_{k-1})}{x_k - x_{k-1}}\,.$$

For the resulting *secant iteration* invent a stopping criterion that requires smallness of both $|f(x_k)|$ and $|x_k - x_{k-1}|$.
(c) Calculate the implied volatilities for the data

$$T - t = 0.211\,,\ S_0 = 5290.36\,,\ r = 0.0328$$

and the pairs K, V from Table 1.4 (for more data see www.compfin.de). For each calculated value of σ enter the point (K, σ) into a figure, joining the points with straight lines. (You will notice a convex shape of the curve. This shape has lead to call this phenomenon *volatility smile*.)

1.20 (Ornstein–Uhlenbeck Process)
An Ornstein–Uhlenbeck process is defined as solution of the SDE

$$\mathrm{d}X_t = -\alpha X_t\,\mathrm{d}t + \gamma\,\mathrm{d}W_t\,, \quad \alpha > 0$$

Table 1.4 Calls on the DAX on Jan 4th 1999

K	6000	6200	6300	6350	6400	6600	6800
V	80.2	47.1	35.9	31.3	27.7	16.6	11.4

for a Wiener process W.

(a) Show

$$X_t = \mathrm{e}^{-\alpha t}\left(X_0 + \gamma \int_0^t \mathrm{e}^{\alpha s}\,\mathrm{d}W_s\right).$$

(b) Suppose a volatility σ_t is an Ornstein–Uhlenbeck process. Show that the variance $v_t := \sigma_t^2$ follows a Cox–Ingersoll–Ross process, namely,

$$\mathrm{d}v_t = \kappa(\theta - v_t)\,\mathrm{d}t + \sigma^{\mathrm{v}}\sqrt{v_t}\,\mathrm{d}W_t\,.$$

1.21 (Analytic Solution of Special SDEs)
Apply Itô's lemma to show

(a) $X_t = \exp\left(\lambda W_t - \frac{1}{2}\lambda^2 t\right)$ solves $\mathrm{d}X_t = \lambda X_t\,\mathrm{d}W_t$
(b) $X_t = \exp\left(2W_t - t\right)$ solves $\mathrm{d}X_t = X_t\,\mathrm{d}t + 2X_t\,\mathrm{d}W_t$

Hint: Use suitable functions g with $Y_t = g(X_t, t)$. In (a) start with $X_t = W_t$ and $g(x,t) = \exp(\lambda x - \frac{1}{2}\lambda^2 t)$.

1.22 (Moments of the Lognormal Distribution)
For the density function $f(S; t - t_0, S_0)$ from (1.64) show

(a) $\int_0^\infty Sf(S; t - t_0, S_0)\,\mathrm{d}S = S_0\mathrm{e}^{\mu(t-t_0)}$
(b) $\int_0^\infty S^2 f(S; t - t_0, S_0)\,\mathrm{d}S = S_0^2\mathrm{e}^{(\sigma^2+2\mu)(t-t_0)}$

Hint: Set $y = \log(S/S_0)$ and transform the argument of the exponential function to a squared term.

In case you still have strength afterwards, calculate the value of S for which f is maximal.

1.23 (Return of the Underlying)
Let a time series $S_1, \ldots, S_M$ of a stock price be given (for example data in the domain www.compfin.de). The simple return

$$\hat{R}_{i,j} := \frac{S_i - S_j}{S_j}\,,$$

an index number of the success of the underlying, lacks the desirable property of additivity

$$R_{M,1} = \sum_{i=2}^{M} R_{i,i-1}\,. \tag{1.80}$$

The log return

$$R_{i,j} := \log S_i - \log S_j$$

has better properties.

(a) Show $R_{i,i-1} \approx \hat{R}_{i,i-1}$, and
(b) $R_{i,j}$ satisfies (1.80).
(c) For empirical data calculate the $R_{i,i-1}$ and set up histograms. Calculate sample mean and sample variance.
(d) Suppose S is distributed lognormally. How can a value of the volatility be obtained from an estimate of the variance?
(e) The mean of the 26866 log returns of the time period of 98.66 years of Fig. 1.23 is 0.000199 and the standard deviation is 0.01069. Calculate an estimate of the historical volatility σ.

1.24 (Positive Itô Process)
Let X_t be a positive one-dimensional Itô process for $t \geq 0$.
Show that there exist functions α and β such that

$$\mathrm{d}X_t = X_t(\alpha_t\,\mathrm{d}t + \beta_t\,\mathrm{d}W_t)$$

and

$$X_t = X_0 \exp\left\{\int_0^t (\alpha_s - \frac{1}{2}\beta_s^2)\,\mathrm{d}s + \int_0^t \beta_s\,\mathrm{d}W_s\right\}\,.$$

1.25 (General Black–Scholes Equation)
Assume a portfolio

$$\Pi_t = \alpha_t S_t + \beta_t B_t$$

consisting of α_t units of a stock S_t and β_t units of a bond B_t, which obey

$$\begin{aligned}\mathrm{d}S_t &= \mu(S_t,t)\,\mathrm{d}t + \sigma(S_t,t)\,\mathrm{d}W_t\,,\\ \mathrm{d}B_t &= r(t)B_t\,\mathrm{d}t\,.\end{aligned}$$

The functions μ, σ, and r are assumed to be known, and $\sigma > 0$. Further assume the portfolio is *self-financing* in the sense

$$\mathrm{d}\Pi_t = \alpha_t\,\mathrm{d}S_t + \beta_t\,\mathrm{d}B_t\,,$$

and *replicating* such that Π_T equals the payoff of a European option. (Then Π_t equals the price of the option for all t.) Derive the Black–Scholes equation for this scenario, assuming $\Pi_t = g(S_t,t)$ with g sufficiently often differentiable.
Hint: coefficient matching of two versions of $\mathrm{d}\Pi_t$.

1.26 (Binomial Method with p = 0.5)
Use the Eqs. (1.12), (1.16) and $p = 1/2$ to show

$$u = \mathrm{e}^{r\Delta t}\,(1 + \sqrt{\mathrm{e}^{\sigma^2\Delta t} - 1})\,,$$
$$d = \mathrm{e}^{r\Delta t}\,(1 - \sqrt{\mathrm{e}^{\sigma^2\Delta t} - 1})\,.$$

1.27 (Closeness to the Black–Scholes Equation)
Let $\widetilde{V}$ be a smooth function satisfying the continuous version of the binomial recursion formula

$$\widetilde{V}(S, t - \Delta t) = \mathrm{e}^{-r\Delta t}\,\left(p\widetilde{V}(uS, t) + (1-p)\widetilde{V}(dS, t)\right)\,,$$

for u, d, p as defined in (1.18), with $\gamma = 1$. Apply the Taylor expansion of $\widetilde{V}$ about (S, t) to show that this recursion is close to the Black–Scholes equation up to terms of order $O(\Delta t)$.
Hint: Show

$$p(u-1) - (1-p)(1-d) = \mathrm{e}^{r\Delta t} - 1 = r\,\Delta t + O(\Delta t^2)\,,$$
$$p(u-1)^2 + (1-p)(1-d)^2 = \sigma^2\,\Delta t + O(\Delta t^2)\,.$$

(based on reference [234], Sec.5.1.5)

Chapter 2
Generating Random Numbers with Specified Distributions

Simulation and valuation of finance instruments require numbers with specified distributions. For example, in Sect. 1.6 we have used numbers Z drawn from a standard normal distribution, $Z \sim \mathcal{N}(0,1)$. If possible the numbers should be random. But the generation of "random numbers" by digital computers, after all, is done in a deterministic and entirely predictable way. If this point is to be stressed, one uses the term *pseudo-random*.[1]

Computer-generated random numbers mimic the properties of true random numbers as much as possible. This is discussed for uniformly distributed random numbers in Sect. 2.1. Suitable transformations or rejection methods generate samples from other distributions, in particular, normally distributed numbers (Sects. 2.2 and 2.3). Section 2.3 includes the vector case, where normally distributed numbers are calculated with prescribed correlation.

Another approach is to dispense with randomness and to generate *quasi-random numbers*, which aim at avoiding one disadvantage of random numbers, namely, the potential lack of equidistributedness. The resulting *low-discrepancy* numbers will be discussed in Sect. 2.5. These numbers are used for the deterministic Monte Carlo integration (Sect. 2.4).

Definition 2.1 (Sample from a Distribution) A sequence of numbers is called a *sample from* F if the numbers are independent realizations of a random variable with distribution function F.
If F is the uniform distribution over the interval $[0,1]$, then we call the samples from F *uniform deviates (variates)*, notation $\sim \mathcal{U}[0,1]$. If F is the standard normal distribution then we call the samples from F *standard normal deviates (variates)*; as notation we use $\sim \mathcal{N}(0,1)$. The basis of random-number generation is to draw uniform deviates.

[1] Since in our context the predictable origin is clear we omit the modifier "pseudo," and hereafter use the term "random number." Similarly we talk about randomness of these numbers when we mean apparent randomness.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_2

2.1 Uniform Deviates

A standard approach to calculate uniform deviates is provided by linear congruential generators. We concentrate on algorithms that are easy to implement and ready for experiments.

2.1.1 Linear Congruential Generators

Choose integers M, a, b, with $a, b < M$, $a \neq 0$. For an integer N_0 a sequence of integers N_i is defined by

Algorithm 2.2 (Linear Congruential Generator)

Choose N_0.
For $i = 1, 2, \ldots$ *calculate*

$$N_i = (aN_{i-1} + b) \bmod M\,. \tag{2.1}$$

The *modulo* congruence $N = Y \bmod M$ between two numbers N and Y is an equivalence relation [147]. The initial integer N_0 is called the *seed*. Numbers $U_i \in [0, 1)$ are defined by

$$U_i = N_i/M\,, \tag{2.2}$$

and will be taken as uniform deviates. Whether the numbers U_i or N_i are suitable will depend on the choice of M, a, b and will be discussed next.

Properties 2.3 (Periodicity)

(a) $N_i \in \{0, 1, \ldots, M-1\}$
(b) The N_i *are periodic with period* $\leq M$.
(Because there are not $M+1$ *different* N_i. *So two in* $\{N_0, \ldots, N_M\}$ *must be equal,* $N_i = N_{i+p}$ *with* $p \leq M$.*)*

Obviously, some peculiarities must be excluded. For example, $N = 0$ must be ruled out in case $b = 0$, because otherwise $N_i = 0$ would repeat. In case $a = 1$ the generator settles down to $N_n = (N_0 + nb) \bmod M$. This sequence is predictable too easily. Various other properties and requirements are discussed in the literature, in particular in [226]. In case the period is M, the numbers U_i are distributed "evenly" when exactly M numbers are needed. Then each grid point on a mesh on [0,1] with mesh size $\frac{1}{M}$ is occupied once.

After these observations we start searching for good choices of M, a, b. There are numerous possible choices with bad properties. For serious computations we recommend to rely on suggestions of the literature. Press et al. [306] presents a table of "quick and dirty" generators, for example, $M = 244{,}944$, $a = 1597$,

$b = 51{,}749$. Criteria are needed to decide which of the many possible generators are recommendable.

2.1.2 *Quality of Generators*

What are good random numbers? A practical answer is the requirement that the numbers should meet "all" aims, or rather pass as many tests as possible. The requirements on good number generators can roughly be divided into three groups.

The first requirement is that of a *large period*. In view of Property 2.3 the number M must be as large as possible, because a small set of numbers makes the outcome easier to predict—a contrast to randomness. This leads to select M close to the largest integer machine number. But a period p close to M is only achieved if a and b are chosen properly. Criteria for relations among M, p, a, b have been derived by number-theoretic arguments. This is outlined in [226, 317]. For 32-bit computers, a common choice has been $M = 2^{31} - 1$, $a = 16807$, $b = 0$.

A second group of requirements are *statistical tests* that check whether the numbers are distributed as intended. The simplest of such tests evaluates the sample mean $\hat{\mu}$ and the sample variance $\hat{s}^2$ (B.11) of the calculated random variates, and compares to the desired values of μ and σ^2. (Recall $\mu = 1/2$ and $\sigma^2 = 1/12$ for the uniform distribution.) Another simple test is to check correlations. For example, it would not be desirable if small numbers are likely to be followed by small numbers.

A slightly more involved test checks how well the probability distribution is approximated. This works for general distributions ($\longrightarrow$ Exercise 2.1). Here we briefly summarize an approach for uniform deviates. Calculate j samples from a random number generator, and investigate how the samples distribute on the unit interval. To this end, divide the unit interval into subintervals of equal length ΔU, and denote by j_k the number of samples that fall into the kth subinterval

$$k\Delta U \le U < (k+1)\Delta U\,.$$

Then j_k/j should be close the desired probability, which for this setup is ΔU. Hence a plot of the quotients

$$\frac{j_k}{j\Delta U} \quad \text{for all } k$$

against $k\Delta U$ should be a good approximation of $\mathbf{1}_{[0,1]}$, the density of the uniform distribution. This procedure is just the simplest test; for more ambitious tests, consult [226].

The third group of tests is to check how well the random numbers distribute in higher-dimensional spaces. This issue of the *lattice structure* is discussed next. We derive a priori analytical results on *where* the random numbers produced by Algorithm 2.2 are distributed.

2.1.3 Random Vectors and Lattice Structure

Random numbers N_i can be arranged in m-tuples $(N_i, N_{i+1}, \ldots, N_{i+m-1})$ for $i \geq 1$. Then the tuples or the corresponding points $(U_i, \ldots, U_{i+m-1}) \in [0, 1)^m$ are analyzed with respect to correlation and distribution. The sequences defined by the generator of Algorithm 2.2 lie on $(m-1)$-dimensional hyperplanes. This statement is trivial since it holds for the M parallel planes through $U = i/M$, $i = 0, \ldots, M-1$ (any of the m components). But if all points fall on only a *small* number of parallel hyperplanes (with large empty gaps in between), then the generator would be impractical in many applications. Next we analyze the generator whether such unfavorable planes exist, restricting ourselves to the case $m = 2$.

For $m = 2$ the hyperplanes in (U_{i-1}, U_i)-space are straight lines, and are defined by $z_0 U_{i-1} + z_1 U_i = \lambda$, with parameters z_0, z_1, λ. The modulus operation (2.1) can be written

$$\begin{aligned} N_i &= (aN_{i-1} + b) \bmod M \\ &= aN_{i-1} + b - kM \quad \text{for } kM \leq aN_{i-1} + b < (k+1)M\,, \end{aligned}$$

k an integer, $k = k(i)$. A side calculation for arbitrary z_0, z_1 shows

$$\begin{aligned} z_0 N_{i-1} + z_1 N_i &= z_0 N_{i-1} + z_1 (aN_{i-1} + b - kM) \\ &= N_{i-1}(z_0 + az_1) + z_1 b - z_1 kM \\ &= M \cdot \underbrace{\{N_{i-1} \frac{z_0 + az_1}{M} - z_1 k\}}_{=:c} + z_1 b\,. \end{aligned}$$

We divide by M and obtain the equation of a straight line in the (U_{i-1}, U_i)-plane, namely,

$$z_0 U_{i-1} + z_1 U_i = c + z_1 b M^{-1}\,. \tag{2.3}$$

The points calculated by Algorithm 2.2 lie on these straight lines. To eliminate the seed we take $i > 1$. For each tuple (z_0, z_1), the Eq. (2.3) defines a family of parallel straight lines, one for each number out of the finite set of c's. The question is whether there exists a tuple (z_0, z_1) such that only few of the straight lines cut the square $[0, 1)^2$. In this case wide areas of the square would be free of random points, which violates the requirement of a "uniform" distribution of the points. The minimum number of parallel straight lines (hyperplanes) cutting the square, or equivalently the maximum distance between them, characterizes the worst case and serves as measure of the equidistributedness. Now we analyze the number of straight lines, searching for the worst case.

For analyzing the number of planes, the cardinality of the c matters. To find the worst case, restrict to integers (z_0, z_1) satisfying

$$z_0 + az_1 = 0 \bmod M\,. \tag{2.4}$$

Then the parameter c is integer. By solving (2.3) for $c = z_0U_{i-1} + z_1U_i - z_1bM^{-1}$ and applying $0 \le U < 1$ we obtain the maximal interval I_c such that for each integer $c \in I_c$ its straight line cuts or touches the square $[0, 1)^2$. Count how many such c's exist, and there is the information we need. For some constellations of a, M, z_0 and z_1 it may be possible that the points (U_{i-1}, U_i) lie on very few of these straight lines!

Example 2.4 (Academic Generator) We discuss the generator

$$N_i = 2N_{i-1} \bmod 11$$

that is, the parameters are $a = 2,\ b = 0,\ M = 11$. The choice $z_0 = -2,\ z_1 = 1$ is one tuple satisfying (2.4), and the resulting family (2.3) of straight lines

$$-2U_{i-1} + U_i = c$$

in the (U_{i-1}, U_i)-plane is to be discussed. For $U \in [0, 1)$ the inequality $-2 < c < 1$ results. In view of (2.4) c is integer and so only the two integers $c = -1$ and $c = 0$ remain. The two corresponding straight lines cut the interior of $[0, 1)^2$. As Fig. 2.1 illustrates, the points generated by the algorithm form a lattice. All points on the lattice lie on these two straight lines. The figure lets us discover also other parallel straight lines such that all points are caught (for other tuples z_0, z_1). The practical question is: What is the largest gap? (⟶ Exercise 2.2)

Example 2.5 $N_i = (1229N_{i-1} + 1) \bmod 2048$

The requirement of Eq. (2.4)

$$\frac{z_0 + 1229z_1}{2048} \text{ integer}$$

is satisfied by $z_0 = -1,\ z_1 = 5$, because

$$-1 + 1229 \cdot 5 = 6144 = 3 \cdot 2048\,.$$

For c from (2.3) and $U_i \in [0, 1)$ we have

$$-1 - \frac{5}{2048} < c < 5 - \frac{5}{2048}\,.$$

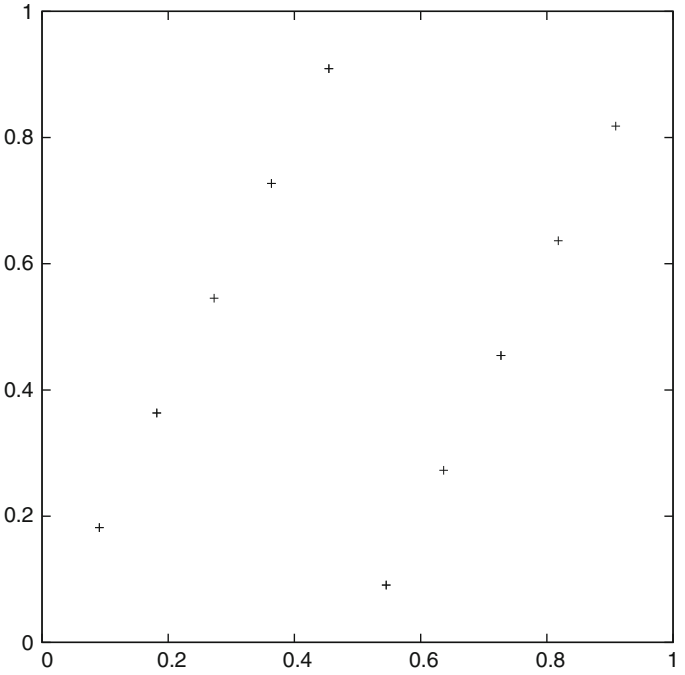

Fig. 2.1 The points (U_{i-1}, U_i) of Example 2.4

Hence $c \in \{-1, 0, 1, 2, 3, 4\}$, and all points (U_{i-1}, U_i) in $[0, 1)^2$ lie on only six straight lines, see Fig. 2.2. On the "lowest" straight line ($c = -1$) there is only one point. The distance between straight lines measured along the vertical U_i–axis is $\frac{1}{z_1} = \frac{1}{5}$. Obviously, the (U_{i-1}, U_i)-points are by far not equidistributed on the square, although the positions U_i appear uniformly distributed on the line.[2]

Higher-dimensional vectors ($m > 2$) are analyzed analogously. The generator called RANDU

$$N_i = aN_{i-1} \bmod M\,, \quad \text{with } a = 2^{16} + 3,\ M = 2^{31}$$

may serve as example. For $m = 2$ experiments show that the points (U_{i-1}, U_i) are nicely equidistributed. But equidistribution for $m = 2$ does not imply equidistribution for larger m. Testing RANDU for $m = 3$ reveals a severe defect: Its random points in the cube $[0, 1)^3$ fall on only 15 planes ($\longrightarrow$ Exercise 2.3 and Topic 14 in the *Topics fCF*).

In Example 2.4 we asked what the maximum gap between the parallel straight lines is. In other words, we have searched for stripes of maximum size in which

[2]The term "equidistributed" will be quantified in Sect. 2.5.1.

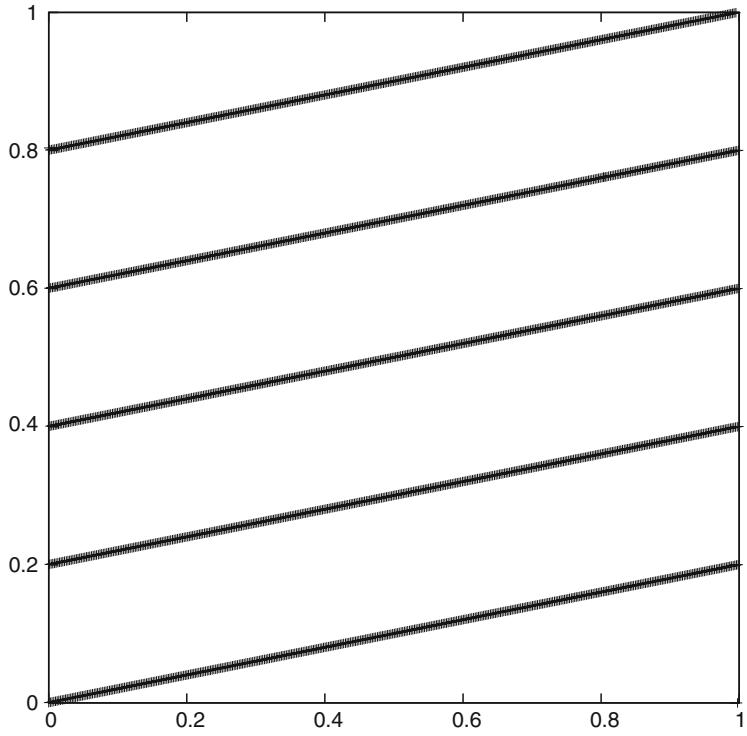

Fig. 2.2 The points (U_{i-1}, U_i) of Example 2.5

no point (U_{i-1}, U_i) falls. Alternatively one can directly analyze the lattice formed by consecutive points. For illustration consider again Fig. 2.1. We follow the points starting with $(\frac{1}{11}, \frac{2}{11})$. By vectorwise adding an appropriate multiple of $(1, a) = (1, 2)$ the next two points are obtained. Proceeding in this way one has to take care that upon leaving the unit square each component with value ≥ 1 must be reduced to $[0, 1)$ to observe $\bmod M$. The reader may verify this with Example 2.4 and numerate the points of the lattice in Fig. 2.1 in the correct sequence. In this way the lattice can be defined. This process of defining the lattice can be generalized to higher dimensions $m > 2$. ($\longrightarrow$ Exercise 2.4) One aims at a good distribution of the points $(U_i, \ldots, U_{i+m-1})$ for as many m are possible.

A disadvantage of the linear congruential generators of Algorithm 2.2 is the boundedness of the period by M and hence by the word length of the computer. The situation can be improved by *shuffling* the random numbers in a random way. For practical purposes, the period gets close enough to infinity. (The reader may test this on Example 2.5.) For practical advice we refer to [306].

2.1.4 *Fibonacci Generators*

The original Fibonacci recursion motivates trying the formula

$$N_{i+1} := (N_i + N_{i-1}) \bmod M\,.$$

It turns out that this first attempt of a three-term recursion is not suitable for generating random numbers (⟶ Exercise 2.5). The modified approach

$$N_{i+1} := (N_{i-\nu} - N_{i-\mu}) \bmod M \tag{2.5}$$

for suitable integers ν, μ is called *lagged Fibonacci* generator. For many choices of ν, μ the approach (2.5) leads to acceptable generators. Kahaner et al. [210] recommends

Example 2.6 (Lagged Fibonacci Generator)

$$U_i := U_{i-17} - U_{i-5}\,,$$
$$\text{in case } U_i < 0 \text{ set } U_i := U_i + 1.0\,.$$

The recursion of Example 2.6 immediately produces floating-point numbers $U_i \in [0, 1)$. This generator requires a prologue in which 17 initial U's are generated by means of another method. The core of the algorithm is

Algorithm 2.7 (Loop of a Fibonacci Generator)

Repeat:
$\zeta = U(i) - U(j)$,
if $(\zeta < 0)$, *set* $\zeta = \zeta + 1$,
$U(i) = \zeta$,
$i = i - 1$,
$j = j - 1$,
if $i = 0$, *set* $i = 17$,
if $j = 0$, *set* $j = 17$.

Initialization: Set $i = 17$, $j = 5$, and calculate $U_1, \ldots, U_{17}$ with a congruential generator, for instance with $M = 714,025$, $a = 1366$, $b = 150{,}889$. Set the seed N_0 equal to your favorite dream number, possibly inspired by the system clock of your computer.

Figure 2.3 depicts 10,000 random points calculated by means of Algorithm 2.7. Visual inspection suggests that the points are not arranged in some apparent structure. The points appear to be sufficiently random. But the generator provided by Example 2.6 is not sophisticated enough for ambitious applications; its pseudo-random numbers are somewhat correlated.

Section 2.1 has introduced some basic aspects of generating uniformly distributed random numbers. Professional algorithms also apply bit operations in

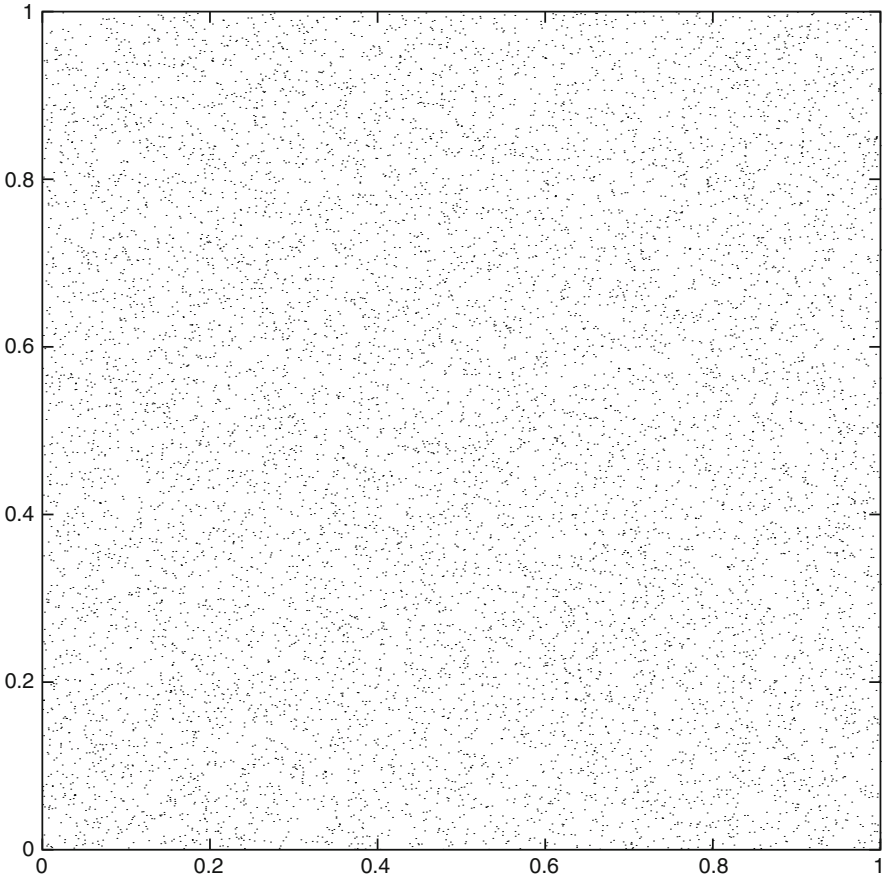

Fig. 2.3 Ten thousand (pseudo-)random points (U_{i-1}, U_i), calculated with Algorithm 2.7

the computer. A generator of uniform deviates that can be highly recommended is a *Mersenne twister* [264]. Its period is truly remarkable, and the points $(U_i, \ldots, U_{i+m-1})$ are well distributed until high values of the dimension m.

2.2 Extending to Random Variables from Other Distributions

Frequently, normal variates are needed. Their generation is based on uniform deviates. The simplest strategy is to calculate

$$X := \sum_{i=1}^{12} U_i - 6, \quad \text{for } U_i \sim \mathcal{U}[0,1]\,.$$

X has expectation 0 and variance 1. The central limit theorem (⟶ Appendix B) assures that X is approximately distributed normally (⟶ Exercise 2.6). But this crude attempt is not satisfying. Better methods calculate nonuniformly distributed random variables, for example, by a suitable transformation out of a uniformly distributed random variable [103]. But the most obvious approach inverts the distribution function.

2.2.1 Inversion

The following theorem is the basis for inversion methods.

Theorem 2.8 (Inversion) *Suppose $U \sim \mathcal{U}[0,1]$ and F be a continuous strictly increasing distribution function. Then $F^{-1}(U)$ is a sample from F.*

Proof Let P denote the underlying probability.

$U \sim \mathcal{U}[0,1]$ means $\quad \mathsf{P}(U \le \xi) = \xi$ for $0 \le \xi \le 1$.

Consequently

$$\mathsf{P}(F^{-1}(U) \le x) = \mathsf{P}(U \le F(x)) = F(x)\,.$$

Application

Following Theorem 2.8, the inversion method[3] generates uniform deviates $u \sim \mathcal{U}[0,1]$ and sets $x = F^{-1}(u)$ (⟶ Exercises 2.7, 2.8, 2.9). There are some examples where the inverse is available analytically. For example, the distribution of the exponential distribution with parameter λ (below in Example 2.10) is $F(x) = 1 - \mathrm{e}^{-\lambda x}$, and its inverse is $F^{-1}(u) = -\frac{1}{\lambda}\log(1-u)$. To judge the inversion method we consider the normal distribution as the most important example. Neither for its distribution function F nor for its inverse F^{-1} there is a closed-form expression (⟶ Exercise 1.5). So numerical methods are used. We discuss two approaches.

Numerical inversion means to calculate iteratively a solution x of the equation $F(x) = u$ for prescribed u. In particular for the normal distribution, this iteration requires tricky termination criteria, in particular when x is large. Then we are in the situation $u \approx 1$, where tiny changes in u lead to large changes in x (Fig. 2.4). An approximation of the solution x of $F(x) - u = 0$ can be calculated with bisection, or Newton's method, or the secant method (⟶ Appendix C.1).

Alternatively the inversion $x = F^{-1}(u)$ can be approximated by a suitably constructed function $G(u)$ with

$$G(u) \approx F^{-1}(u)\,.$$

[3] Also called *inversion sampling*.

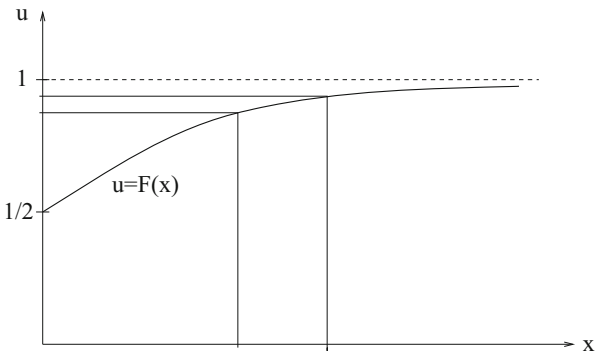

Fig. 2.4 Normal distribution; small changes in u can lead to large changes in x

Then only $x = G(u)$ needs to be evaluated. Constructing such an approximation formula G, it is important to realize that $F^{-1}(u)$ has "vertical" tangents at $u = 1$ (horizontal in Fig. 2.4). The pole behavior must be reproduced correctly by an approximating function G. This suggests to use rational approximation (⟶ Appendix C.1). For the Gaussian distribution one incorporates the point symmetry with respect to $(u, x) = (\frac{1}{2}, 0)$, and the pole at $u = 1$ (and hence at $u = 0$) in the *ansatz* for G (⟶ Exercise 2.10). Rational approximation of $F^{-1}(u)$ with a sufficiently large number of terms leads to high accuracy [278]. The formulas are given in Appendix E.2.

2.2.2 *Transformation in* $\mathbb{R}^1$

Another class of methods uses transformations between random variables. We start the discussion with the scalar case. If we have a random variable X with known density and distribution, what can we say about the density and distribution of a transformed $h(X)$?

Theorem 2.9 (Transformation in Scalar Case) *Suppose X is a random variable with density $f(x)$ and distribution $F(x)$. Further assume $h : S \longrightarrow B$ with $S, B \subseteq \mathbb{R}$, where S is the support[4] of $f(x)$, and let h be strictly monotonic.*

(a) Then $Y := h(X)$ is a random variable. Its distribution F_Y is

$$F_Y(y) = F(h^{-1}(y)) \quad \text{in case } h' > 0\,,$$
$$F_Y(y) = 1 - F(h^{-1}(y)) \quad \text{in case } h' < 0\,.$$

[4] f is zero outside S. (In this section, S is no asset price.) Use Theorem 2.9 to check the derivation of f_{GBM} out of $\hat{f}$ in Sect. 1.8.2.

(b) If h^{-1} is absolutely continuous then for almost all y the density of $h(X)$ is

$$f(h^{-1}(y)) \left| \frac{\mathrm{d}h^{-1}(y)}{\mathrm{d}y} \right| . \tag{2.6}$$

Proof

(a) For $h' > 0$ we have $\mathsf{P}(h(X) \le y) = \mathsf{P}(X \le h^{-1}(y)) = F(h^{-1}(y))$.
(b) For absolutely continuous h^{-1} the density of $Y = h(X)$ is equal to the derivative of the distribution function almost everywhere. Evaluating the derivative $\frac{\mathrm{d}F(h^{-1}(y))}{\mathrm{d}y}$ with the chain rule implies the assertion. The absolute value in (2.6) is necessary such that a positive density comes out in case $h' < 0$. (See for instance [131, Sect. 2.4 C].)

2.2.2.1 Application

Being able to calculate uniform deviates, we start from $X \sim \mathcal{U}[0, 1]$ with the density f of the uniform distribution,

$$f(x) = 1 \text{ for } 0 \le x \le 1, \text{ otherwise } f = 0 .$$

Here the support S is the unit interval. What we need are random numbers Y matching a prespecified target density $g(y)$. It remains to find a transformation h such that the density in (2.6) is identical to $g(y)$,

$$1 \cdot \left| \frac{\mathrm{d}h^{-1}(y)}{\mathrm{d}y} \right| = g(y) .$$

Then only evaluate $h(X)$.

Example 2.10 (Exponential Distribution) The exponential distribution with parameter $\lambda > 0$ has the density

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

Here the range B consists of the nonnegative real numbers. The aim is to generate an exponentially distributed random variable Y out of a $\mathcal{U}[0, 1]$-distributed random variable X. To this end define the monotone transformation from the unit interval $S = [0, 1]$ into B by the decreasing function

$$y = h(x) := -\frac{1}{\lambda} \log x$$

with the inverse function $h^{-1}(y) = \mathrm{e}^{-\lambda y}$ for $y \geq 0$. For this h verify

$$f(h^{-1}(y)) \left| \frac{\mathrm{d}h^{-1}(y)}{\mathrm{d}y} \right| = 1 \cdot \left| (-\lambda)\mathrm{e}^{-\lambda y} \right| = \lambda \mathrm{e}^{-\lambda y} = g(y)$$

as density of $h(X)$. Hence $h(X)$ is distributed exponentially as long as $X \sim \mathcal{U}[0, 1]$.
Application:

In case $U_1, U_2, \ldots$ are nonzero uniform deviates, the numbers $h(U_i)$

$$-\frac{1}{\lambda} \log(U_1), \quad -\frac{1}{\lambda} \log(U_2), \ \ldots$$

are distributed exponentially. This result is similar to that of the inversion. For an application see Exercise 2.11.

2.2.2.2 Attempt to Generate a Normal Distribution

Starting from the uniform distribution ($f = 1$) a transformation $y = h(x)$ is searched such that its density equals that of the standard normal distribution,

$$1 \cdot \left| \frac{\mathrm{d}h^{-1}(y)}{\mathrm{d}y} \right| = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2} y^2 \right) .$$

This is a differential equation for h^{-1} without analytic solution. As we will see, a transformation can be applied successfully in $\mathbb{R}^2$. To this end we need a generalization of the scalar transformation of Theorem 2.9 into $\mathbb{R}^n$.

2.2.3 *Transformations in* $\mathbb{R}^n$

The generalization of Theorem 2.9 to the vector case is

Theorem 2.11 (Transformation in Vector Case) *Suppose X is a random variable in $\mathbb{R}^n$ with density $f(x) > 0$ on the support S. The transformation $h : S \to B$, $S, B \subseteq \mathbb{R}^n$ is assumed to be invertible and the inverse be continuously differentiable on B. $Y := h(X)$ is the transformed random variable. Then Y has the density*

$$f(h^{-1}(y)) \left| \frac{\partial(x_1, \ldots, x_n)}{\partial(y_1, \ldots, y_n)} \right|, \quad y \in B , \tag{2.7}$$

where $x = h^{-1}(y)$ and $\frac{\partial(x_1, \ldots, x_n)}{\partial(y_1, \ldots, y_n)}$ is the determinant of the Jacobian matrix of all first-order derivatives of $h^{-1}(y)$.
(Theorem 4.2 in [103])

2.2.4 Acceptance-Rejection Method

An acceptance-rejection method[5] is based on the following facts: Let f be a density function on the support $S \subset \mathbb{R}$ and $\mathcal{A}_f$ the area between the x-axis and the graph of f. Assume two random variables U and X independent of each other with $U \sim \mathcal{U}[0, 1]$ and X distributed with density f. Then the points

$$(x, y) := (X,\ U \cdot f(X))$$

are distributed uniformly on $\mathcal{A}_f$. And vice versa, the x-coordinates of uniformly distributed points on $\mathcal{A}_f$ are f-distributed. This is illustrated in Fig. 2.5 for the normal distribution. If one cuts off a piece of the area $\mathcal{A}_f$, then the remaining points are still distributed uniformly. This is exploited by rejection methods.

The aim is to calculate f-distributed random numbers; the density f is the target distribution. Let g be another density on S, and assume for a constant $c \geq 1$

$$f(x) \leq c\, g(x) \quad \text{for all } x \in S\,.$$

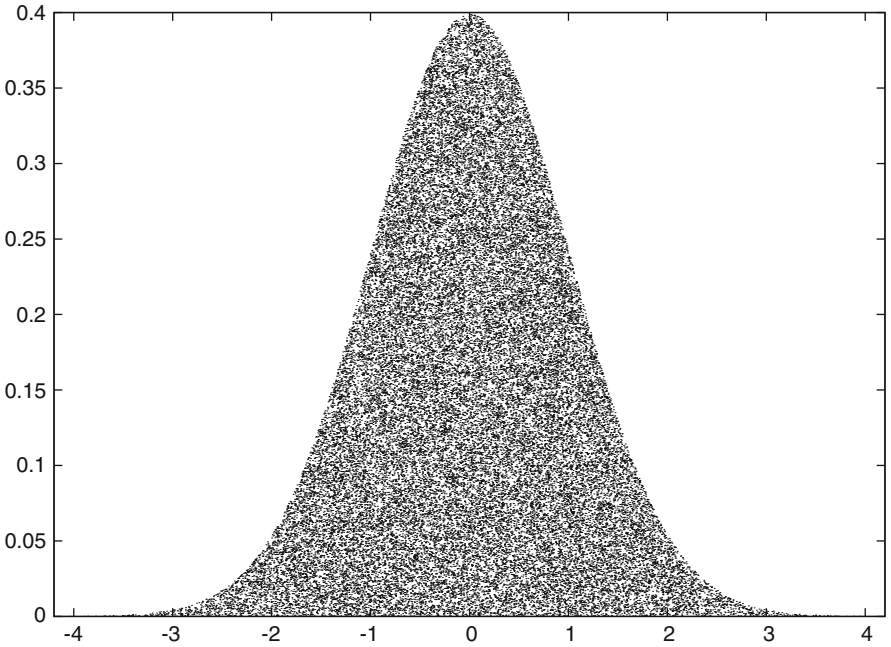

Fig. 2.5 Fifty thousand points $(X,\ Uf(X))$, with $X \sim \mathcal{N}(0, 1)$, $U \sim \mathcal{U}[0, 1]$. The normal density f of X is visible as envelope

[5] Shortly: rejection method, or rejection sampling.

The function cg is major to f, and the set $\mathcal{A}_f$ is subset of the area $\mathcal{A}_{cg}$ underneath the graph of cg. A rejection algorithm assumes that g-distributed x-samples can be calculated easily. Then the points $(x, ucg(x))$ are distributed uniformly on $\mathcal{A}_{cg}$. Cutting off the part of $\mathcal{A}_{cg}$ above $\mathcal{A}_f$ means to reject points with $ucg(x) > f(x)$. The x-coordinates of the remaining points with $ucg(x) \le f(x)$ are accepted and are distributed as desired.

Algorithm 2.12 (Rejection Method)

Repeat:
$x :=$ *random number distributed with density* g,
$u :=$ *random number* $\sim \mathcal{U}[0,1]$ *independent of* x,
until $u\,c\,g(x) \le f(x)$.
return: x

As an application of the rejection method consider the Laplace density $g(x) := \frac{1}{2}\exp(-|x|)$ and the standard normal density f, see Exercises 2.9 and 2.12.[6]

2.3 Normally Distributed Random Variables

In this section the focus is on generating normal variates. Fist we describe the fundamental approach of Box and Muller, which applies the transformation method in $\mathbb{R}^2$ to generate Gaussian random numbers.[7]

2.3.1 Method of Box and Muller

To apply Theorem 2.11 we start with the unit square $S := [0,1]^2$ and the density (2.7) of the bivariate uniform distribution. The transformation is

$$\begin{aligned} y_1 &= \sqrt{-2\log x_1}\,\cos 2\pi x_2 =: h_1(x_1, x_2) \\ y_2 &= \sqrt{-2\log x_1}\,\sin 2\pi x_2 =: h_2(x_1, x_2)\,, \end{aligned} \tag{2.8}$$

$h(x)$ is defined on $[0,1]^2$ with values in $\mathbb{R}^2$. Its inverse function h^{-1} is given by

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

[6] Colored in Topic 3 of the *Topics fCF*.

[7] Inversion is one of several valid alternatives. See also the Notes on this section.

where we take the main branch of arctan. The determinant of the Jacobian matrix is

$$\begin{aligned}\frac{\partial(x_1,x_2)}{\partial(y_1,y_2)} &= \det\begin{pmatrix} \frac{\partial x_1}{\partial y_1} & \frac{\partial x_1}{\partial y_2} \\ \frac{\partial x_2}{\partial y_1} & \frac{\partial x_2}{\partial y_2} \end{pmatrix} = \\ &= \frac{1}{2\pi}\exp\left\{-\frac{1}{2}(y_1^2+y_2^2)\right\}\left(-y_1\frac{1}{1+\frac{y_2^2}{y_1^2}}\frac{1}{y_1} - y_2\frac{1}{1+\frac{y_2^2}{y_1^2}}\frac{y_2}{y_1^2}\right) \\ &= -\frac{1}{2\pi}\exp\left\{-\frac{1}{2}(y_1^2+y_2^2)\right\}\,.\end{aligned}$$

This shows that $\left|\frac{\partial(x_1,x_2)}{\partial(y_1,y_2)}\right|$ is the density (2.7) of the bivariate standard normal distribution. Since this density is the product of the two one-dimensional densities,

$$\left|\frac{\partial(x_1,x_2)}{\partial(y_1,y_2)}\right| = \left[\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}y_1^2\right)\right]\cdot\left[\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}y_2^2\right)\right],$$

the two components of the vector y are independent. So, when the components of the vector X are $\sim \mathcal{U}[0,1]$, the vector $h(X)$ consists of two independent standard normal variates. Let us summarize the application of this transformation:

Algorithm 2.13 (Box–Muller)

Generate $U_1 \sim \mathcal{U}[0,1]$ *and* $U_2 \sim \mathcal{U}[0,1]$.
$\theta := 2\pi U_2\,, \quad \rho := \sqrt{-2\log U_1}\,.$
$Z_1 := \rho\cos\theta$ *is a normal variate (as well as* $Z_2 := \rho\sin\theta$*).*

The variables U_1, U_2 stand for the components of X. Each application of the algorithm provides two standard normal variates. Note that a line structure in $[0,1]^2$ as in Example 2.5 is mapped to curves in the (Z_1,Z_2)-plane. This underlines the importance of excluding an evident line structure.

2.3.2 *Variant of Marsaglia*

The variant of Marsaglia prepares the input in Algorithm 2.13 such that trigonometric functions are avoided. For $U \sim \mathcal{U}[0,1]$ we have $V := 2U-1 \sim \mathcal{U}[-1,1]$. (Temporarily we misuse also the financial variable V for local purposes.) Two values V_1, V_2 calculated in this way define a point in the (V_1,V_2)-plane. Only points within the unit disk $\mathcal{D}$ are accepted:

$$\mathcal{D} := \{\,(V_1,V_2) \;\mid\; V_1^2+V_2^2 < 1\,\}\,;\ \text{accept only } (V_1,V_2)\in\mathcal{D}\,.$$

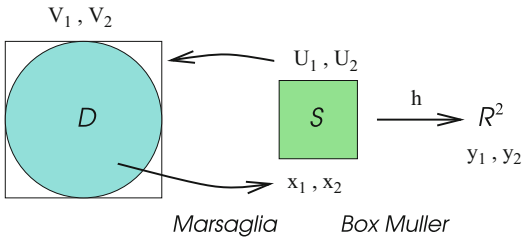

Fig. 2.6 Transformations of the Box–Muller–Marsaglia approach, schematically

In case of rejection both values V_1, V_2 must be rejected. As a result, the surviving (V_1, V_2) are uniformly distributed on $\mathcal{D}$ with density $f(V_1, V_2) = \frac{1}{\pi}$ for $(V_1, V_2) \in \mathcal{D}$. A transformation from the disk $\mathcal{D}$ into the unit square $S := [0, 1]^2$ is defined by

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} V_1^2 + V_2^2 \\ \frac{1}{2\pi} \arg((V_1, V_2)) \end{pmatrix} .$$

That is, the Cartesian coordinates V_1, V_2 on $\mathcal{D}$ are mapped to the squared radius and the normalized angle.[8] For illustration, see Fig. 2.6. These "polar coordinates" (x_1, x_2) are uniformly distributed on S ($\longrightarrow$ Exercise 2.13).

Application
For input in (2.8) use $V_1^2 + V_2^2$ as x_1 and $\frac{1}{2\pi} \arctan \frac{V_2}{V_1}$ as x_2. With these variables the relations

$$\cos 2\pi x_2 = \frac{V_1}{\sqrt{V_1^2 + V_2^2}} , \quad \sin 2\pi x_2 = \frac{V_2}{\sqrt{V_1^2 + V_2^2}} ,$$

hold, which means that it is no longer necessary to evaluate trigonometric functions. The resulting algorithm of Marsaglia has modified the Box–Muller method by constructing input values x_1, x_2 in a clever way.

Algorithm 2.14 (Polar Method)

Repeat:
generate $U_1, U_2 \sim \mathcal{U}[0, 1]$ *;*
calculate $V_1 := 2U_1 - 1$, $V_2 := 2U_2 - 1$
until $w := V_1^2 + V_2^2 < 1$.
$Z_1 := V_1 \sqrt{-2 \log(w)/w}$
$Z_2 := V_2 \sqrt{-2 \log(w)/w}$
are both standard normal variates.

[8] $\arg((V_1, V_2)) = \arctan(V_2/V_1)$ with the proper branch.

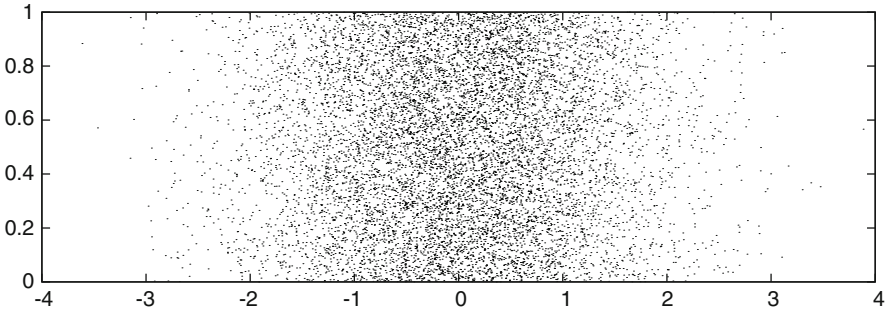

Fig. 2.7 Ten thousand numbers $\sim \mathcal{N}(0,1)$ (values entered horizontally and separated vertically with distance 10^{-4})

The probability that $w < 1$ holds is given by the ratio of the areas, $\pi/4 = 0.785\ldots$ Hence in about 21% of all $\mathcal{U}[0,1]$ drawings the (V_1, V_2)-tuple is rejected because of $w \geq 1$. Nevertheless the savings of the trigonometric evaluations makes Marsaglia's polar method more efficient than the Box–Muller method. Figure 2.7 illustrates normally distributed random numbers ($\longrightarrow$ Exercise 2.14).

2.3.3 Ziggurat

A most efficient algorithm for the generation of normal deviates is the *ziggurat* algorithm, which is a rejection method. The setup consists of a kind of horizontal histogram, which covers the area underneath the graph of a monotonically decreasing f. Figure 2.8, which will explained below, may give an impression of the setup.[9] Here f is the standard normal density $f(x) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{1}{2}x^2)$. Because of the symmetry of f it suffices to take $x \geq 0$; a random sign (each with probability $\frac{1}{2}$) must be attached in the end.

The histogram-like area consists of N horizontal and parallel segments each *of equal area* A. We label them by i, with $i = 0$ for the bottom layer and $i = N-1$ for the top layer. The top $N-1$ segments are rectangles, whereas the lowest segment ($i = 0$) is limited by the infinite tail of f. The lengths of the segments are defined by f, as illustrated in Figs. 2.8 and 2.9. The upper edges of the segments define a major function z with $z(x) \geq f(x)$ for $x \geq 0$. The major z corresponds to cg in Sect. 2.2.4.

The curve of $f(x)$, decreasing for $x > 0$, enters and leaves the layers, which defines the length x_i of the rectangle, as shown in Fig. 2.9. For a chosen value of N, the requirement of equal area A of all segments leads to a system of equations

[9] The shape explains the use of the name ziggurat, which was a terraced pyramid in the ancient world.

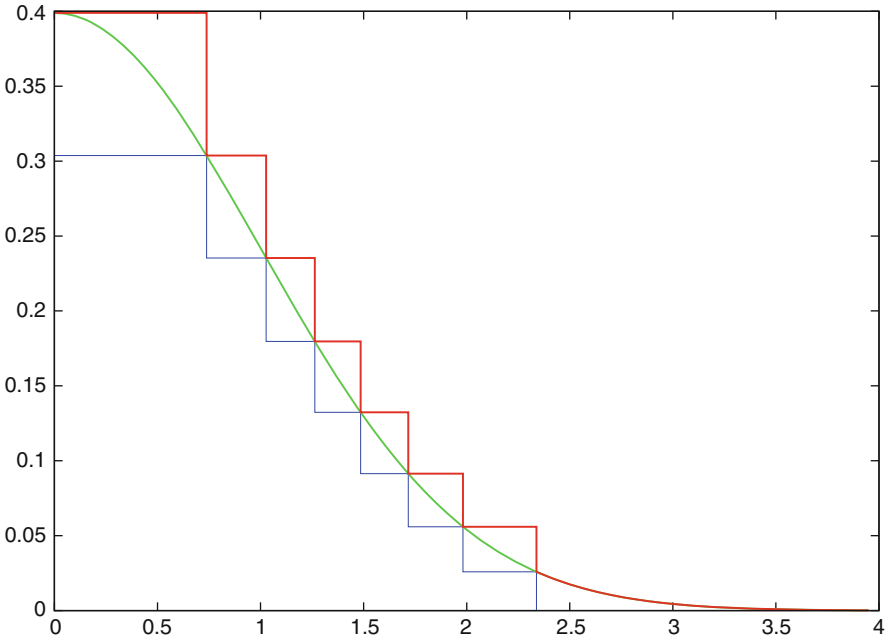

Fig. 2.8 Ziggurat with $N = 8$ layers, for $0 \leq x \leq 4$. The heavy-line zigzag (in *red*) on the *right* and above the graph of the normal density f (in *green*) is the major z with $z(x) \geq f(x)$, which represents the right-hand bound of the horizontal ziggurat boxes. For $N = 8$ the area of each ziggurat segment is 0.070283. The zigzag in *blue* that is below f bounds the area, in which the creation of a normally distributed sample essentially only costs one generation of $U \sim \mathcal{U}[0, 1]$

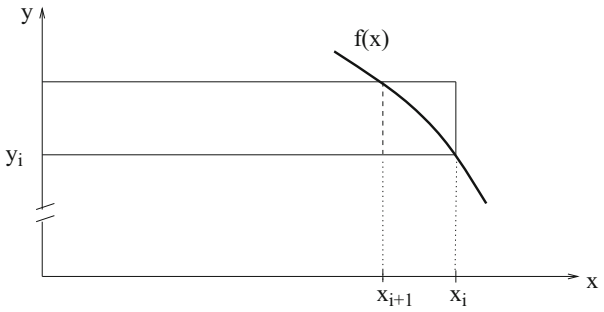

Fig. 2.9 Configuration of the ith layer of the ziggurat, $0 < i < N - 1$, for $x \geq 0$

that defines A and the coordinates (x_i, y_i) of the vertices of the rectangles, where $y_i := f(x_i)$. The coordinates (x_i, y_i) and the value of A are precomputed and stored in a look-up table (⟶ Exercise 2.15). Figure 2.9 illustrates the ith layer ($0 < i < N - 1$). The resulting box consists of two sub-boxes, divided by the coordinate x_{i+1}.

The rejection method needs points (ξ, η) uniformly distributed over the area $\mathcal{A}_z$ underneath the graph of z and above the positive x-axis. In principle, these points are tested for their location relative to f. With the above setup, the check for acceptance or rejection is extremely efficient, because mostly η is not needed explicitly. Since each of the N segments has the same area A, it suffices to draw one of them. Draw the layer i randomly with equal probability $1/N$. Let us first discuss the cases $i > 0$. In rectangle i the next task would be to sample a point (ξ, η), which must be distributed uniformly. Its x-component is given by $\xi := U_1 x_i$, where $U_1 \sim \mathcal{U}[0, 1]$. In case $\xi \le x_{i+1}$ the point falls in the left-hand part of the rectangle underneath the graph of f, and is accepted. In this case no y-component η is needed! (This does not happen for $i = N - 1$, where $x_N = 0$.) Only in the other case, for $\xi > x_i$, an η is required and f must be evaluated to further test for $\eta \le f(\xi)$. This is provided by generating a $U_2 \sim \mathcal{U}[0, 1]$ and $\eta := y_i + U_2(y_{i+1} - y_i)$. Acceptance for $\eta \le f(\xi)$.

The efficiency of the method originates from the fact that the y-component η will be required only in a small portion of samples. In Fig. 2.8 we have chosen $N = 8$ for ease of demonstration. But even for this small value of N the subarea in which no η and no $f(\xi)$ are needed, covers 72.8% of the area underneath z. And when the number N of layers is large, say $N = 256$, the rectangles are narrow, and for $0 < i < N - 1$ the right-hand portions of the rectangles will be much smaller than the left-hand portions. The latter cover the bulk of the area underneath f or z, and there the test for acceptance costs almost nothing: The generated value of U_1 can be compared directly to precomputed ratios x_{i+1}/x_i. In case of acceptance, the output is ξ, and—with attached random sign—the desired number is distributed $\sim \mathcal{N}(0, 1)$. In case of rejection the next i is drawn.

Only the situation of the bottom layer $i = 0$ is more complex. This bottom segment is divided into a rectangle with area $x_1 y_1$, and the infinite tail with $x > x_1$ and area $A - x_1 y_1$. For $i = 0$, the probability of a uniformly sampled point to fall into the rectangle is $x_1 y_1 / A$. So the above simple test can be modified to comparing $\xi := U_1 A / y_1$ to x_1. Accept in case $\xi \le x_1$. Only in the case $\xi > x_1$ the ziggurat algorithm requires a fallback routine, which resorts to more conventional methods. But this fallback routine for $i = 0$ effects only a tiny part of the overall costs. Even for the small value $N = 8$ of Fig. 2.8, the fallback routine is required only in 2% of all samples. For the tricky implementation of the ziggurat algorithm see [261].

2.3.4 Correlated Random Variables

The above algorithms provide independent normal deviates. In many applications random variables are required that depend on each other in a prescribed way. Let us first recall the general n-dimensional density function.

Multivariate normal distribution (notations):

$$X = (X_1, \ldots, X_n), \quad \mu = \mathsf{E}X = (\mathsf{E}X_1, \ldots, \mathsf{E}X_n)$$

The covariance matrix (B.8) of X is denoted Σ, and has elements

$$\Sigma_{ij} = (\mathsf{Cov}X)_{ij} := \mathsf{E}\left((X_i - \mu_i)(X_j - \mu_j)\right), \quad \sigma_i^2 = \Sigma_{ii},$$

for $i, j = 1, \ldots, n$. Using this notation, the correlation coefficients are

$$\rho_{ij} := \frac{\Sigma_{ij}}{\sigma_i \sigma_j} \qquad (\Rightarrow \ \rho_{ii} = 1), \tag{2.9}$$

which set up the correlation matrix. The correlation matrix is a scaled version of Σ. The density function $f(x_1, \ldots, x_n)$ corresponding to $\mathcal{N}(\mu, \Sigma)$ is

$$f(x) = \frac{1}{(2\pi)^{n/2}} \frac{1}{(\det \Sigma)^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu)^{tr} \Sigma^{-1} (x-\mu)\right\}. \tag{2.10}$$

By theory, a covariance matrix (or correlation matrix) Σ is symmetric, and positive semidefinite. If in practice a matrix $\tilde{\Sigma}$ is corrupted by insufficient data, a close matrix Σ can be calculated with the features of a covariance matrix [184, 200]. In case $\det \Sigma \neq 0$ the matrix Σ is positive definite, which we assume now.

Below we shall need a factorization of Σ into $\Sigma = AA^{tr}$. From numerical mathematics we know that for symmetric positive definite matrices Σ the Cholesky decomposition $\Sigma = LL^{tr}$ exists, with a lower triangular matrix L ($\longrightarrow$ Appendix C.1). There are numerous factorizations $\Sigma = AA^{tr}$ other than Cholesky. A more involved factorization of Σ is the principal component analysis, which is based on eigenvectors ($\longrightarrow$ Exercise 2.16).

2.3.4.1 Transformation

Suppose $Z \sim \mathcal{N}(0, I)$ and $x = Az$, $A \in \mathbb{R}^{n \times n}$, where z is a realization of Z, 0 is the zero vector, and I the identity matrix. We apply Theorem 2.11 with $X = h(Z) := AZ$. Accordingly, the density of X is

$$\begin{aligned} f(A^{-1}x)\,|\det(A^{-1})| &= \frac{1}{(2\pi)^{n/2}} \exp\left\{-\frac{1}{2}(A^{-1}x)^{tr}(A^{-1}x)\right\} \frac{1}{|\det(A)|} \\ &= \frac{1}{(2\pi)^{n/2}} \frac{1}{|\det(A)|} \exp\left\{-\frac{1}{2}x^{tr}(AA^{tr})^{-1}x\right\} \end{aligned}$$

for arbitrary nonsingular matrices A. To complete the transformation,[10] we need a matrix A such that $\Sigma = AA^{tr}$. Then $|\det A| = (\det \Sigma)^{1/2}$, and the densities with the respect to x and z are converted correctly. In view of the general density $f(x)$ recalled in (2.10), AZ is normally distributed with $AZ \sim \mathcal{N}(0, AA^{tr})$, and hence the factorization $\Sigma = AA^{tr}$ implies

$$AZ \sim \mathcal{N}(0, \Sigma)\,.$$

Finally, translation with vector μ implies

$$\mu + AZ \sim \mathcal{N}(\mu, \Sigma)\,. \tag{2.11}$$

2.3.4.2 Application

Suppose we need a normal variate $X \sim \mathcal{N}(\mu, \Sigma)$ for given mean vector μ and covariance matrix Σ. This is most conveniently based on the Cholesky decomposition of Σ. Accordingly, the desired random variable can be calculated with the following algorithm:

Algorithm 2.15 (Correlated Normal Random Variables)

Calculate A via the Cholesky decomposition $AA^{tr} = \Sigma$.
Calculate $Z \sim \mathcal{N}(0, I)$ componentwise
by $Z_i \sim \mathcal{N}(0,1)$ for $i = 1, \ldots, n$,
for instance, with Marsaglia's polar algorithm.
$\mu + AZ$ has the desired distribution $\sim \mathcal{N}(\mu, \Sigma)$.

Special case $n = 2$: In this case, in view of (2.9), only one correlation number is involved, namely, $\rho := \rho_{12} = \rho_{21}$, and the covariance matrix must be of the form

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}. \tag{2.12}$$

In this two-dimensional situation it makes sense to carry out the Cholesky decomposition analytically (⟶ Exercise 2.17). Figure 2.10 illustrates a highly correlated two-dimensional situation, with $\rho = 0.85$. An example based on (2.12) is (3.35).

[10] Check this by applying Theorem 2.11.

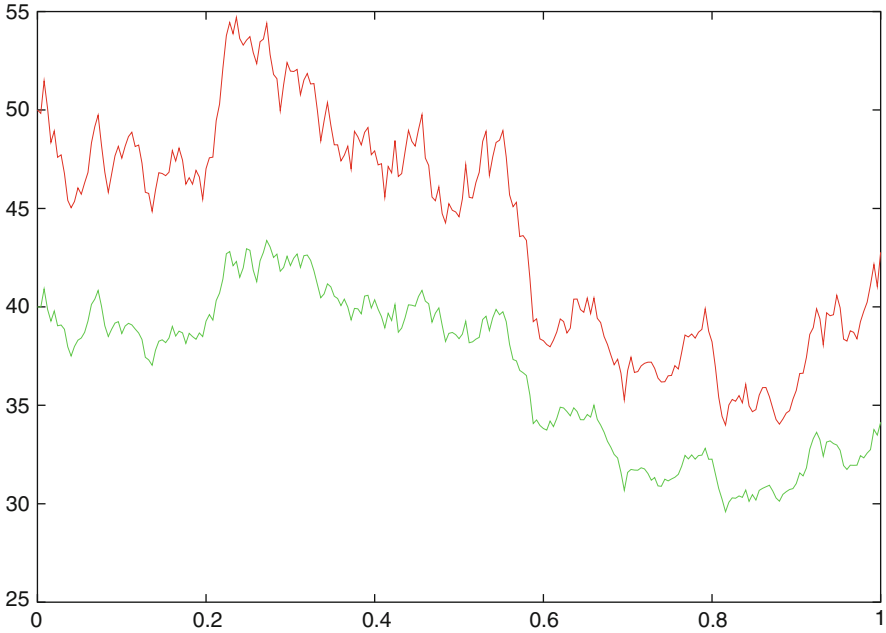

Fig. 2.10 Simulation of a correlated vector process with two components, and $\mu = 0.05$, $\sigma_1 = 0.3$, $\sigma_2 = 0.2$, $\rho = 0.85$, $\Delta t = 1/250$

2.4 Monte Carlo Integration

A classic application of random numbers is Monte Carlo integration. The discussion in this section will serve as background for Quasi Monte Carlo, a topic of the following Sect. 2.5.

Let us begin with the one-dimensional situation. Assume a probability distribution with density f. Then the expectation of a function g is

$$\mathsf{E}(g) = \int_{-\infty}^{\infty} g(x) f(x)\,\mathrm{d}x\,,$$

compare (B.4). For a definite integral on an interval $\mathcal{D} = [a, b]$, we use the uniform distribution with density

$$f = \frac{1}{b-a} \cdot \mathbf{1}_{\mathcal{D}} = \frac{1}{\lambda_1(\mathcal{D})} \cdot \mathbf{1}_{\mathcal{D}}\,,$$

where $\lambda_1(\mathcal{D})$ denotes the length of the interval $\mathcal{D}$ and $\mathbf{1}_\mathcal{D}$ the identity on $\mathcal{D}$. This leads to

$$\mathsf{E}(g) = \frac{1}{\lambda_1(\mathcal{D})} \int_a^b g(x)\,\mathrm{d}x\,,$$

or

$$\int_a^b g(x)\,\mathrm{d}x = \lambda_1(\mathcal{D}) \cdot \mathsf{E}(g)\,,$$

the basis of *Monte Carlo integration*. It remains to approximate $\mathsf{E}(g)$. For independent samples $x_k \sim \mathcal{U}[a,b]$, $k = 1, 2, \ldots$, apply the law of large numbers (⟶ Appendix B.1) to establish the estimator

$$\frac{1}{N} \sum_{k=1}^{N} g(x_k)$$

as approximation to $\mathsf{E}(g)$. The approximation improves as the number of trials N goes to infinity; the error is characterized by the central limit theorem.

This principle of Monte Carlo integration extends to the higher-dimensional case. Let $\mathcal{D} \subset \mathbb{R}^m$ be a domain on which the integral

$$\int_\mathcal{D} g(x)\,\mathrm{d}x$$

is to be calculated. For example, on the hypercube $\mathcal{D} = [0,1]^m$. Such integrals occur in finance, for example, when mortgage-backed securities (CMO, collateralized mortgage obligations) are valuated [64]. The classic or *stochastic Monte Carlo integration* draws random samples $x_1, \ldots, x_N \in \mathcal{D}$ which should be independent and uniformly distributed. Then

$$\theta_N := \lambda_m(\mathcal{D}) \frac{1}{N} \sum_{k=1}^{N} g(x_k) \tag{2.13}$$

is an approximation of the integral. Here $\lambda_m := \lambda_m(\mathcal{D})$ is the volume of $\mathcal{D}$ (or the m-dimensional Lebesgue measure [286]). We assume λ_m to be finite. From the law of large numbers follows convergence of θ_N to $\lambda_m \mathsf{E}(g) = \int_\mathcal{D} g(x)\,\mathrm{d}x$ for $N \to \infty$. The variance of the error

$$\delta_N := \int_\mathcal{D} g(x)\,\mathrm{d}x - \theta_N$$

Table 2.1 Comparison of different convergence rates to zero

N	$\frac{1}{\sqrt{N}}$	$\sqrt{\frac{\log\log N}{N}}$	$\frac{\log N}{N}$	$\frac{(\log N)^2}{N}$	$\frac{(\log N)^3}{N}$
10^1	0.31622777	0.28879620	0.23025851	0.53018981	1.22080716
10^2	0.10000000	0.12357911	0.04605170	0.21207592	0.97664572
10^3	0.03162278	0.04396186	0.00690776	0.04771708	0.32961793
10^4	0.01000000	0.01490076	0.00092103	0.00848304	0.07813166
10^5	0.00316228	0.00494315	0.00011513	0.00132547	0.01526009
10^6	0.00100000	0.00162043	0.00001382	0.00019087	0.00263694
10^7	0.00031623	0.00052725	0.00000161	0.00002598	0.00041874
10^8	0.00010000	0.00017069	0.00000018	0.00000339	0.00006251
10^9	0.00003162	0.00005506	0.00000002	0.00000043	0.00000890

satisfies

$$\mathsf{Var}(\delta_N) = \mathsf{E}(\delta_N^2) - (\mathsf{E}(\delta_N))^2 = \frac{\sigma^2(g)}{N}(\lambda_m)^2\,, \tag{2.14}$$

with the variance of g

$$\sigma^2(g) := \frac{1}{\lambda_m}\int_{\mathcal{D}} g(x)^2\,\mathrm{d}x - \frac{1}{\lambda_m^2}\left(\int_{\mathcal{D}} g(x)\,\mathrm{d}x\right)^2\,. \tag{2.15}$$

Hence the standard deviation of the error δ_N tends to 0 with the order $O(N^{-1/2})$. This result follows from the central limit theorem or from other arguments ($\longrightarrow$ Exercise 2.18). The deficiency of the order $O(N^{-1/2})$ is the slow convergence ($\longrightarrow$ Exercise 2.19 and the second column in Table 2.1). To reach an absolute error of the order ε, Eq. (2.14) tells that the sample size is $N = O(\varepsilon^{-2})$. To improve the accuracy by a factor of 10, the costs (that is the number of trials, N) increase by a factor of 100. Another disadvantage is the lack of a genuine error *bound*. The probabilistic error of (2.14) does not rule out the risk that the result may be completely wrong. The $\sigma^2(g)$ in (2.15) is not known and must be approximated. Monte Carlo integration responds sensitively to changes of the initial state of the used random-number generator. This may be explained by the potential clustering of random points.

In many applications the above deficiencies are balanced by two good features of Monte Carlo integration: A first advantage is that the order $O(N^{-1/2})$ of the error holds independently of the dimension m. Another good feature is that the integrands g need not be smooth, square integrability suffices ($g \in \mathcal{L}^2$, see Appendix C.3).

So far we have described the basic version of Monte Carlo integration, stressing the slow decline of the probabilistic error with growing N. The variance of the error δ can also be diminished by decreasing the numerator in (2.14). This variance of the problem can be reduced by suitable methods. (We will come back to this issue in Sect. 3.5.4.)

We conclude the excursion into the stochastic Monte Carlo integration with the variant for those cases in which $\lambda_m(\mathcal{D})$ is hard to calculate. For $\mathcal{D} \subseteq [0,1]^m$ and $x_1, \ldots, x_N \sim \mathcal{U}[0,1]^m$ use

$$\int_{\mathcal{D}} f(x)\,\mathrm{d}x \approx \frac{1}{N} \sum_{\substack{k=1 \\ x_k \in \mathcal{D}}}^{N} f(x_k)\,. \tag{2.16}$$

For the integral (1.66) with density f_{GBM}, see Sect. 3.5.

2.5 Sequences of Numbers with Low Discrepancy

One difficulty with random numbers is that they may fail to distribute uniformly. Here, "uniform" is not meant in the stochastic sense of a distribution $\sim \mathcal{U}[0,1]$, but has the meaning of an equidistributedness that avoids extreme clustering or holes. The aim is to generate numbers for which the deviation from uniformity is minimal. This deviation is called "discrepancy." Another objective is to obtain good convergence for some important applications.

2.5.1 Discrepancy

The bad convergence behavior of the stochastic Monte Carlo integration is not inevitable. For example, for $m = 1$ and $\mathcal{D} = [0,1]$ an equidistant x-grid with mesh size $1/N$ leads to a formula (2.13) that resembles the trapezoidal sum [(C.2) in Appendix C.1]. For smooth g, the order of the error is at least $O(N^{-1})$. (Why?) But such a grid-based evaluation procedure is somewhat inflexible because the grid must be prescribed in advance and the number N that matches the desired accuracy is unknown beforehand. In contrast, the free placing of sample points with Monte Carlo integration can be performed until some termination criterion is met. It would be desirable to find a compromise in placing sample points such that the fineness advances but clustering is avoided. The sample points should fill the integration domain $\mathcal{D}$ as uniformly as possible. To this end we require a measure of the *equidistributedness*.[11]

For $m \geq 1$ let $Q \subseteq [0,1]^m$ be an arbitrary axially parallel m-dimensional box (hyperrectangle) in the unit cube $[0,1]^m$ of $\mathbb{R}^m$. That is, Q is a product of m intervals. Suppose a set of points $x_1, \ldots, x_N \in [0,1]^m$. The decisive idea behind discrepancy is that for an evenly distributed point set, the fraction of the points lying within the box Q should correspond to the volume of the box (see Fig. 2.11). Let # denote the

[11] The deterministic term "equidistributed" is not to be confused with the probabilistic "uniformly distributed".

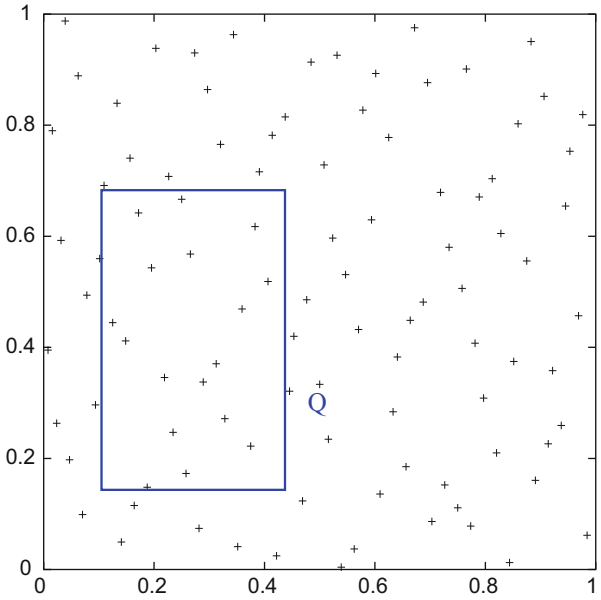

Fig. 2.11 On the idea of discrepancy, here for $m = 2$

number of points, then the goal is

$$\frac{\#\text{ of } x_i \in Q}{\#\text{ of all points in } [0,1]^m} \approx \frac{\text{vol}(Q)}{\text{vol}([0,1]^m)}$$

for as many boxes Q as possible. This leads to the following definition:

Definition 2.16 (Discrepancy) The discrepancy of the point set $\{x_1, \ldots, x_N\} \subset [0,1]^m$ is

$$D_N := \sup_Q \left| \frac{\#\text{ of } x_i \in Q}{N} - \text{vol}(Q) \right| .$$

Obviously, Figs. 2.1 and 2.2 allow to construct relatively large rectangles Q such that no points land on Q. Then D_N will not become small for increasing N. The more evenly the points of a sequence are distributed, the closer the discrepancy D_N is to zero. The criterion

$$\lim_{N\to\infty} D_N = 0$$

will characterize equidistributed points. Here D_N refers to the first N points of a sequence of points (x_i), $i \geq 1$.

Analogously the variant D_N^* *(star discrepancy)* is obtained when the set of boxes is restricted to those Q^*, for which one corner is the origin:

$$Q^* = \prod_{i=1}^{m} [0, y_i)$$

where $y \in \mathbb{R}^m$ denotes the corner diagonally opposite the origin. The discrepancies D_N and D_N^* satisfy [⟶ Exercise 2.20(b)]

$$D_N^* \le D_N \le 2^m D_N^* \,.$$

The discrepancy allows to find a deterministic bound on the error δ_N of Monte Carlo integration,

$$|\delta_N| \le \mathcal{V}(g) D_N^* \,; \tag{2.17}$$

here $\mathcal{V}(g)$ is the variation[12] of the function g with $\mathcal{V}(g) < \infty$, and the domain of integration is $\mathcal{D} = [0, 1]^m$ [280, 286, 363]. This result is known as Theorem of Koksma and Hlawka. The bound in (2.17) underlines the importance to find numbers $x_1, \ldots, x_N$ with small value of the discrepancy D_N. After all, a set of N *randomly* chosen points satisfies

$$\mathsf{E}(D_N) = O\left(\sqrt{\frac{\log \log N}{N}}\right) .$$

This is in accordance with the probabilistic $O(N^{-1/2})$ law. The order of magnitude of these numbers is shown in Table 2.1 (third column).

Definition 2.17 (Low-Discrepancy Point Sequence) A sequence of points or numbers $x_1, x_2, \ldots, x_N, \ldots \in [0, 1]^m$ is called low-discrepancy sequence if

$$D_N \le C_m \frac{(\log N)^m}{N} \tag{2.18}$$

for a constant C_m independent of N.

Deterministic sequences of numbers satisfying (2.18) are also called *quasi*-random numbers, although they are fully deterministic. Table 2.1 reports on the orders of magnitude. Since $\log(N)$ grows only modestly, a low discrepancy essentially means $D_N \approx O(N^{-1})$ as long as the dimension m is small. The Eq. (2.18)

[12] As in Sect. 1.6.2.

expresses some dependence on the dimension m, contrary to Monte Carlo methods. But the dependence on m in (2.18) is less stringent than with classic MC quadrature.

2.5.2 Examples of Low-Discrepancy Sequences

In the one-dimensional case ($m = 1$) the point set

$$x_i = \frac{2i-1}{2N}, \quad i = 1, \ldots, N \tag{2.19}$$

has the value $D_N^* = \frac{1}{2N}$; this value can not be improved [⟶ Exercise 2.20(c)]. The monotone sequence (2.19) can be applied only when a reasonable N is known and fixed; for $N \to \infty$ the x_i would be newly placed and an integrand g evaluated again. Since N is large, it is essential that the previously calculated results can be used when N is growing. This means that the points $x_1, x_2, \ldots$ must be placed "dynamically" so that they are preserved and the fineness improves when N grows. This is achieved by the sequence

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

This sequence is known as van der Corput sequence. To motivate such a dynamical placing of points imagine that you are searching for some item in the interval $[0, 1]$ (or in the cube $[0, 1]^m$). The searching must be fast and successful, and is terminated as soon as the object is found. This defines N dynamically by the process.

The formula that defines the van der Corput sequence can be formulated as algorithm. Let us study an example, say, $x_6 = \frac{3}{8}$. The index $i = 6$ is written as binary number

$$6 = (110)_2 =: (d_2\, d_1\, d_0)_2 \quad \text{with} \quad d_i \in \{0, 1\}\,.$$

Then reverse the binary digits and put the radix point in front of the sequence:

$$(.\, d_0\, d_1\, d_2)_2 = \frac{d_0}{2} + \frac{d_1}{2^2} + \frac{d_3}{2^3} = \frac{1}{2^2} + \frac{1}{2^3} = \frac{3}{8}$$

If this is done for all indices $i = 1, 2, 3, \ldots$ the van der Corput sequence $x_1, x_2, x_3, \ldots$ results. These numbers can be defined with the following function:

Definition 2.18 (Radical-Inverse Function) For $i = 1, 2, \ldots$ let j be given by the expansion in base b (integer ≥ 2)

$$i = \sum_{k=0}^{j} d_k b^k\,,$$

with digits $d_k \in \{0, 1, \ldots, b-1\}$, which depend on b, i. Then the radical-inverse function is defined by

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

The function $\phi_b(i)$ is the digit-reversed fraction of i. This mapping can be seen as reflecting with respect to the radix point. To each index i a rational number $\phi_b(i)$ in the interval $0 < x < 1$ is assigned. Every time the number of digits j increases by one, the mesh becomes finer by a factor $1/b$. This means that the algorithm fills all mesh points on the sequence of meshes with increasing fineness (⟶ Exercise 2.21). Van der Corput's sequence is obtained by

$$x_i := \phi_2(i) .$$

The radical-inverse function can be applied to construct points x_i in the m-dimensional cube $[0, 1]^m$. A simple construction is the Halton sequence.

Definition 2.19 (Halton Sequence) Let $p_1, \ldots, p_m$ be pairwise prime integers. The Halton sequence is defined as the sequence of vectors

$$x_i := \big(\phi_{p_1}(i), \ldots, \phi_{p_m}(i)\big) , \quad i = 1, 2, \ldots$$

Usually one takes $p_1, \ldots, p_m$ as the first m prime numbers. Figure 2.12 shows for $m = 2$ and $p_1 = 2,\ p_2 = 3$ the first 10,000 Halton points. Compared to the pseudo-random points of Fig. 2.3, the Halton points are distributed more evenly.

Halton sequences x_i of Definition 2.19 are easily constructed, but fail to be equidistributed when the dimension m is high, see [155], Sect. 5.2. Then correlations between the radical-inverse functions for different dimensions are observed. This problem can be cured with a simple modification of the Halton sequence, namely, by using only every lth Halton number [227]. The leap l is a prime different from all bases $p_1, \ldots, p_m$. The result is the "Halton sequence leaped"

$$x_k := \big(\phi_{p_1}(lk), \ldots, \phi_{p_m}(lk)\big) , \quad k = 1, 2, \ldots \tag{2.20}$$

This modification has shown good performance for dimensions at least up to $m = 400$. As reported in [227], $l = 409$ is one example of a good leap value.

Other sequences with low discrepancy have been constructed. These include the sequences developed by Sobol, Faure and Niederreiter, see [280, 286, 306]. All these sequences satisfy

$$D_N^* \le C_m \frac{(\log N)^m}{N} + O\left(\frac{(\log N)^{m-1}}{N}\right) .$$

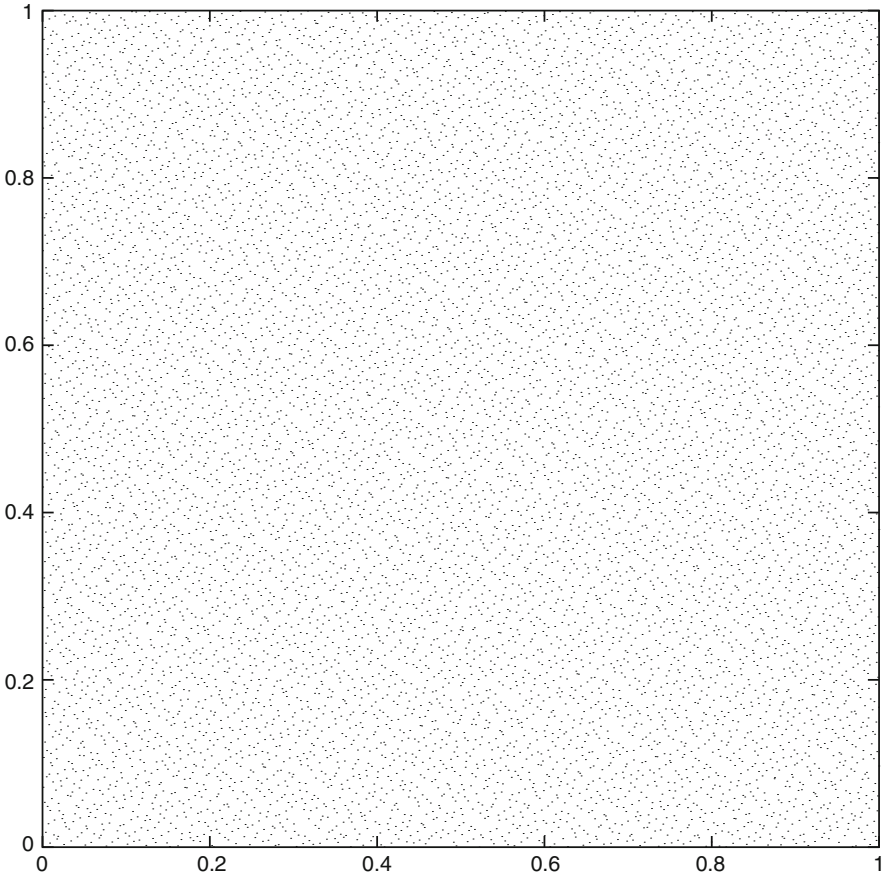

Fig. 2.12 Ten thousand Halton points from Definition 2.19, with $p_1 = 2, p_2 = 3$

Table 2.1 shows how fast the relevant terms $(\log N)^m/N$ tend to zero. If m is large, extremely large values of the denominator N are needed before the terms become small. But it is assumed that the bounds are unrealistically large and overestimate the real error.

Quasi Monte Carlo (QMC) methods approximate the integrals with the arithmetic mean θ_N of (2.13), but use low-discrepancy numbers x_i instead of random numbers. QMC is a *deterministic* method. Practical experience with low-discrepancy sequences are better than might be expected from the bounds known so far. This also holds for the bound (2.17) by Koksma and Hlawka; apparently a large class of functions g satisfy $|\delta_N| \ll \mathcal{V}(g)D_N^*$, see [343].

2.6 Notes and Comments

On Sect. 2.1

The linear congruential method is sometimes called Lehmer generator. Easily accessible and popular generators are RAN1 and RAN2 from [306]. Further references on linear congruential generators include [239, 259, 286, 317]. Example 2.4 is from [130], and Example 2.5 from [317]. Nonlinear congruential generators are of the form

$$N_i = f(N_{i-1}) \mod M\,.$$

Hints on the algorithmic implementation are found in [147]. Generally it is advisable to run the generator in integer arithmetic in order to avoid rounding errors that may spoil the period, see [241]. There are multiplicative Fibonacci generators of the form

$$N_{i+1} := N_{i-\nu} N_{i-\mu} \mod M\,.$$

For Fibonacci generators we refer to [54]. The version of (2.5) is a subtractive generator. Additive versions (with a plus sign instead of the minus sign) are used as well [147, 226]. The codes in [306] are recommendable. For simple statistical tests with illustrations see [181].

Hints on parallelization are given in [262]. For example, parallel Fibonacci generators are obtained by different initializing sequences. Marsaglia's KISS-generator *(keep it simple stupid)* combines different methods and reaches long periods. Programs of professional random number generators (RNG) can be found in the internet. Note that computer systems and software packages often provide built-in random number generators. But often these generators are not clearly specified, and should be handled with care.

On Sects. 2.2 and 2.3

The inversion result of Theorem 2.8 can be formulated placing less or no restrictions on F, see [317, p. 59], [103, p. 28], or [238, p. 270]. The generalized inverse of an arbitrary distribution function F is the *quantile function*

$$Q(u) := \inf_{x \in \mathbb{R}} \{\, x \mid F(x) \geq u \,\}\,,$$

also denoted $F^{-1}(u)$.

For the rejection method, $\frac{1}{c}$ is the proportion of samples distributed from g that are accepted. Hence c should be as small as possible with $c \geq \max_x \frac{f(x)}{g(x)}$. Several

algorithms are based on the rejection method [103, 130]; for a detailed overview with many references see [103].

The Box–Muller approach was suggested in [45]. Marsaglia's modification was published in a report quoted in [260]. Fast algorithms aside from the ziggurat include the Wallace algorithm [372], which works with a pool of random numbers and suitable transformations. Platform-dependent implementation details place emphasis on the one or the other advantage. A survey on Gaussian random number generators is [355]. For simulating Lévy processes, see [84]. For singular symmetric positive *semi*definite matrices Σ ($x^{tr}\Sigma x \geq 0$ for all x), the Cholesky decomposition can be cured, see [157], or [155].

On Sect. 2.4

The bounds on errors of the Monte Carlo integration refer to arbitrary functions g; for smooth functions better bounds can be expected. In the one-dimensional case the variation is defined as the supremum of $\sum_j |g(t_j) - g(t_{j-1})|$ over all partitions, see Sect. 1.6.2. This definition can be generalized to higher-dimensional cases. A thorough discussion is [285, 286].

An advanced application of Monte Carlo integration uses one or more methods of *reduction of variance*, which allows to improve the accuracy in many cases [130, 167, 234, 238, 286, 306, 324]. For example, the integration domain can be split into subsets *(stratified sampling)* [316]. Another technique is used when for a *control variate* v with $v \approx g$ the exact integral is known. Then g is replaced by $(g - v) + v$ and Monte Carlo integration is applied to $g - v$. Another alternative, the method of *antithetic variates*, will be described in Sect. 3.5.4 together with the control-variate technique.

On Sect. 2.5

Besides the supremum discrepancy of Definition 2.16 the $\mathcal{L}^2$-analogy of an integral version is used. Hints on speed and preliminary comparison are found in [280]. For application on high-dimensional integrals see [296]. For large values of the dimension m, the bound (2.18) takes large values, which might suggest to discard its use. But the notion of an *effective dimension* and practical results give a favorable picture at least for CMO applications of order $m = 360$ [64]. The error bound of Koksma and Hlawka (2.17) is not necessarily recommendable for practical use, see the discussion in [343]. The analogy of the equidistant lattice in (2.19) in higher-dimensional space has unfavorable values of the discrepancy, $D_N = O\left(\frac{1}{\sqrt[m]{N}}\right)$. For $m > 2$ this is worse than Monte Carlo, compare [317]. Monte Carlo does not take advantage of smoothness of integrands. In the case of smooth integrands, sparse-grid

approaches are highly competitive. These refined quadrature methods meliorate the *curse of the dimension*, see [148, 149, 312].

Van der Corput sequences can be based also on other bases. Halton's paper is [166]. Computer programs that generate low-discrepancy numbers are available. For example, Sobol numbers are calculated in [306] and Sobol- and Faure numbers in the computer program FINDER [296] and in [354]. At the current state of the art it is open which point set has the smallest discrepancy in the m-dimensional cube. There are generalized Niederreiter sequences, which include Sobol- and Faure sequences as special cases [354]. In several applications deterministic Monte Carlo seems to be superior to stochastic Monte Carlo [295]. A comparison based on finance applications has shown good performance of Sobol numbers; in [206] Sobol numbers are outperformed by Halton sequences leaped (2.20). Niederreiter and Jau-Shyong Shiue [287] and Chap. 5 in [155] provide more discussion and many references.

Besides volume integration, Monte Carlo is needed to integrate over possibly high-dimensional probability distributions. Drawing samples from the required distribution can be done by running a cleverly constructed Markov chain. This kind of method is called Markov Chain Monte Carlo (MCMC). That is, a chain of random variables $X_0, X_1, X_2, \ldots$ is constructed where for given X_j the next state X_{j+1} does not depend on the history of the chain $X_0, X_1, X_2, \ldots, X_{j-1}$. By suitable construction criteria, convergence to any chosen target distribution is obtained. For MCMC we refer to the literature, for example to [32, 153, 164, 238, 365].

2.7 Exercises

2.1 (Testing a Distribution)
Let X be a random variate with density f and let $a_1 < a_2 < \ldots < a_l$ define a partition of the support of f into subintervals, including the unbounded intervals $x < a_1$ and $x > a_l$. Recall from (B.1), (B.2) that the probability of a realization of X falling into $a_k \le x < a_{k+1}$ is given by

$$p_k := \int_{a_k}^{a_{k+1}} f(x)\,\mathrm{d}x\,,\ k = 1, 2, \ldots, l-1\,,$$

which can be approximated by $(a_{k+1} - a_k) f\left(\frac{a_k + a_{k+1}}{2}\right)$. Perform a sample of j realizations $x_1, \ldots, x_j$ of a random number generator, and denote j_k the number of samples falling into $a_k \le x < a_{k+1}$. For normal variates with density f from (B.9) design an algorithm that performs a simple statistical test of the quality of the $x_1, \ldots, x_j$.

Hints: See Sect. 2.1 for the special case of uniform variates. Argue for what choices of a_1 and a_l the probabilities p_0 and p_l may be neglected. Think about a reasonable relation between l and j.

2.2 (Academic Number Generator)
Consider the random number generator $N_i = 2N_{i-1} \bmod 11$. For $(N_{i-1}, N_i) \in \{0, 1, \ldots, 10\}^2$ and integer tuples with $z_0 + 2z_1 = 0 \bmod 11$ the equation

$$z_0 N_{i-1} + z_1 N_i = 0 \bmod 11$$

defines families of parallel straight lines, on which all points (N_{i-1}, N_i) lie. These straight lines are to be analyzed. For which of the families of parallel straight lines are the gaps maximal?

2.3 (Deficient Random Number Generator)
For some time the generator

$$N_i = aN_{i-1} \bmod M, \quad \text{with } a = 2^{16} + 3,\ M = 2^{31}$$

was in wide use. Show for the sequence $U_i := N_i/M$

$$U_{i+2} - 6U_{i+1} + 9U_i \text{ is integer.}$$

What does this imply for the distribution of the triples (U_i, U_{i+1}, U_{i+2}) in the unit cube?

2.4 (Lattice of the Linear Congruential Generator)

(a) Show by induction over j

$$N_{i+j} - N_j = a^j (N_i - N_0) \bmod M$$

(b) Show for integer $z_0, z_1, \ldots, z_{m-1}$

$$\begin{pmatrix} N_i \\ N_{i+1} \\ \vdots \\ N_{i+m-1} \end{pmatrix} - \begin{pmatrix} N_0 \\ N_1 \\ \vdots \\ N_{m-1} \end{pmatrix} = (N_i - N_0) \begin{pmatrix} 1 \\ a \\ \vdots \\ a^{m-1} \end{pmatrix} + M \begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ z_{m-1} \end{pmatrix}$$

$$= \begin{pmatrix} 1 & 0 & \cdots & 0 \\ a & M & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ a^{m-1} & 0 & \cdots & M \end{pmatrix} \begin{pmatrix} z_0 \\ z_1 \\ \vdots \\ z_{m-1} \end{pmatrix}$$

2.5 (Quality of Fibonacci-Generated Numbers)
Analyze and visualize the planes in the unit cube, on which all points fall that are generated by the Fibonacci recursion

$$U_{i+1} := (U_i + U_{i-1}) \bmod 1 \,.$$

2.6 (Coarse Approximation of Normal Deviates)
Let $U_1, U_2, \ldots$ be independent random numbers $\sim \mathcal{U}[0, 1]$, and

$$X_k := \sum_{i=k}^{k+11} U_i - 6 \,.$$

Calculate mean and variance of the X_k.

2.7 (Cauchy-Distributed Random Numbers)
A Cauchy-distributed random variable has the density function

$$f_c(x) := \frac{c}{\pi} \frac{1}{c^2 + x^2} \,.$$

Show that its distribution function F_c and its inverse F_c^{-1} are

$$F_c(x) = \frac{1}{\pi} \arctan \frac{x}{c} + \frac{1}{2} \,, \quad F_c^{-1}(y) = c \tan(\pi(y - \frac{1}{2})) \,.$$

How can this be used to generate Cauchy-distributed random numbers out of uniform deviates?

2.8 (Inversion)
Use the inversion method and uniformly distributed $U \sim \mathcal{U}[0, 1]$ to calculate a stochastic variable X with distribution

$$F(x) = 1 - \mathrm{e}^{-2x(x-a)} \,, \; x \geq a \,.$$

2.9 (Laplace Distribution)
The density function of the Laplace distribution is

$$g(x) := \frac{1}{2} \exp(-|x|) \,.$$

(a) Derive the distribution function

$$G(x) := \int_{-\infty}^{x} g(s) \, \mathrm{d}s$$

and its inverse.

(b) Formulate an algorithm that calculates random variates from the G-distribution, applying the inversion method and using $U \sim \mathcal{U}[0,1]$ as input.

2.10 (Inverting the Normal Distribution)
Suppose $F(x)$ is the standard normal distribution function. Construct a rough approximation $G(u)$ to $F^{-1}(u)$ for $0.5 \le u < 1$ as follows:

(a) Construct a rational function $G(u)$ ($\longrightarrow$ Appendix C.1) with correct asymptotic behavior, point symmetry with respect to $(u, x) = (0.5, 0)$, using only one parameter.
(b) Fix the parameter by interpolating a given point $(x_1, F(x_1))$.
(c) What is a simple criterion for the error of the approximation?

2.11 (Time-Changed Wiener Process)
For a time-changing function $\tau(t)$ set $\tau_j := \tau(j\,\Delta t)$ for some time increment Δt.

(a) Argue why Algorithm 1.8 changes to $W_j = W_{j-1} + Z\sqrt{\tau_j - \tau_{j-1}}$ (last line).
(b) Let τ_j be the exponentially distributed jump instances of a Poisson experiment, see Sect. 1.9 and Property 1.20(e). How should the jump intensity λ be chosen such that the expectation of the $\Delta\tau$ is Δt? Implement and test the algorithm, and visualize the results. Experiment with several values of the jump intensity λ.

2.12 (Rejection)
Two density functions g and f are given by

$$f(x) := \frac{1}{\sqrt{2\pi}} \exp(-\frac{x^2}{2}) \qquad \text{(Gaussian density)}$$

$$g(x) := \frac{1}{2} \exp(-|x|) \qquad \text{(Laplace density)}$$

Establish the smallest c such that $cg(x) \ge f(x)$ for all $x \in \mathbb{R}$. Apply the rejection method to generate normally distributed x; use Exercise 2.9.

2.13 (Uniform Distribution)
For the uniformly distributed random variables (V_1, V_2) on the unit disk consider the transformation

$$\begin{pmatrix} X_1 \\ X_2 \end{pmatrix} = \begin{pmatrix} V_1^2 + V_2^2 \\ \frac{1}{2\pi} \arg((V_1, V_2)) \end{pmatrix}$$

where $\arg((V_1, V_2))$ denotes the corresponding angle. Show that (X_1, X_2) is distributed uniformly.

2.14 (Programming Assignment: Normal Deviates)

(a) Write a computer program that implements the *Fibonacci generator*

$$U_i := U_{i-17} - U_{i-5}$$
$$U_i := U_i + 1 \text{ in case } U_i < 0$$

in the form of Algorithm 2.7.

Tests: Visual inspection of 10,000 points in the unit square.

(b) Write a computer program that implements *Marsaglia's Polar Algorithm* (Algorithm 2.14). Use the uniform deviates from a).

Tests:

1.) For a sample of 5000 points calculate estimates of mean and variance.
2.) For the discretized SDE

$$\Delta x = 0.1\Delta t + Z\sqrt{\Delta t}, \quad Z \sim \mathcal{N}(0,1)$$

calculate some trajectories for $0 \le t \le 1,\ \Delta t = 0.01,\ x_0 = 0$.

2.15 (Ziggurat)

Let f be the normal density function, and (x_i, y_i) for $i = 1, \ldots, N-1$ the coordinates of the vertices of the ziggurat, as indicated in Fig. 2.8, and $y_i := f(x_i)$. (Compare Sect. 2.3.3.) Label the segments $i = 0, \ldots, N-1$ from bottom to top; for $i > 0$ these are rectangular boxes. All segments have equal area A, which is to be determined iteratively.

(a) Assume for a moment the parameter A to be given. Set up an equation that defines x_{N-1} implicitly as function $x_{N-1} = \alpha(A)$.
(b) Set up an equation that defines x_1 implicitly, again depending on A. Then set up a recursion that defines $x_2, \ldots, x_{N-1}$ based on the value x_1. After numerically solving these implicit equations one obtains another version for x_{N-1}, which can be regarded as a function $x_{N-1} = \beta(A)$. Of course both values must be the same, $\alpha(A) = \beta(A)$. This equation can be solved iteratively for A, say, by bisection.
(c) For $N = 8$ formulate an algorithm that calculates A. What is a reasonable initial guess for A? Note that neither α nor β are given explicitly; they can be evaluated numerically.

2.16 (Spectral Decomposition of a Covariance Matrix)

For symmetric positive definite $n \times n$ matrices Σ there exists a set of orthonormal eigenvectors $v^{(1)}, \ldots, v^{(n)}$ and eigenvalues $\lambda_1 \ge \cdots \ge \lambda_n > 0$ such that

$$\Sigma v^{(j)} = \lambda_j v^{(j)}, \quad j = 1, \ldots, n\,.$$

Arrange the n eigenvector columns into the $n \times n$ matrix $B := (v^{(1)}, \ldots, v^{(n)})$, and the eigenvalues into the diagonal matrices $\Lambda := \mathrm{diag}(\lambda_1, \ldots, \lambda_n)$ and $\Lambda^{\frac{1}{2}} := \mathrm{diag}(\sqrt{\lambda_1}, \ldots, \sqrt{\lambda_n})$.

(a) Show $\Sigma B = B\Lambda$.
(b) Show that

$$A := B\Lambda^{\frac{1}{2}}$$

factorizes Σ in the sense $\Sigma = AA^{tr}$.
(c) Show

$$AZ = \sum_{j=1}^{n} \sqrt{\lambda_j}\, Z_j\, v^{(j)}\,.$$

(d) And the reversal of Sect. 2.3.4 holds: For a random vector $X \sim \mathcal{N}(0, \Sigma)$ the transformed random vector $A^{-1}X$ has uncorrelated components: Show $\mathsf{Cov}(A^{-1}X) = I$ and $\mathsf{Cov}(B^{-1}X) = \Lambda$.
(e) For the 2×2 matrix

$$\Sigma = \begin{pmatrix} 5 & 1 \\ 1 & 10 \end{pmatrix}$$

calculate the Cholesky decomposition and $B\Lambda^{\frac{1}{2}}$.

Hint: The above is the essence of the *principal component analysis*. Here Σ represents a covariance matrix or a correlation matrix. (For an example see Fig. 2.13.) The matrix B and the eigenvalues in Λ reveal the structure of the data. B defines a linear transformation of the data to a rectangular coordinate system, and the eigenvalues λ_j measure the corresponding variances. In case $\lambda_{k+1} \gg \lambda_k$ for some index k, the sum in (c) can be truncated after the kth term in order to reduce the dimension. The computation of B and Λ (and hence A) is costly, but a dominating λ_1 allows for a simple approximation of $v^{(1)}$ by the power method.

2.17 (Correlated Distributions)
Suppose we need a two-dimensional random variable (X_1, X_2) that must be distributed normally with mean 0, and given variances σ_1^2, σ_2^2 and prespecified correlation ρ. How is X_1, X_2 obtained out of $Z_1, Z_2 \sim \mathcal{N}(0, 1)$?

2.18 (Error of the Monte Carlo Integration)
The domain for integration is $\mathcal{D} = [0, 1]^m$. For

$$\theta_N := \frac{1}{N} \sum_{i=1}^{N} g(x_i)\,, \quad \mathsf{E}(g) := \int g\,\mathrm{d}x\,, \quad v := g - \mathsf{E}(g)\,, \quad \delta_N := \int g\,\mathrm{d}x - \theta_N$$

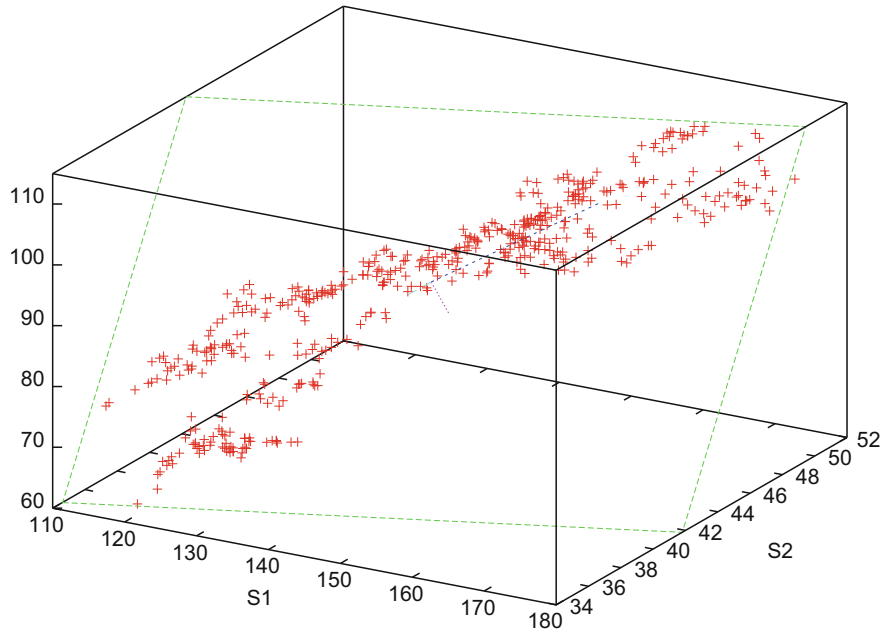

Fig. 2.13 Prices of the DAX assets Allianz (S1), BMW (S2), and HeidelbergCement; 500 trading days from Nov 5, 2005 (in *red*); eigenvalues of the covariance matrix are 400.8, 25.8, 2.73; eigenvectors centered at the mean point and scaled by $\sqrt{\lambda}$ are shown, and the plane (in *green*) spanned by $v^{(1)}, v^{(2)}$

and the variance $\sigma^2(g)$ from (2.15) show

(a) $\mathsf{E}(v) = 0$
(b) $\sigma^2(v) = \sigma^2(g)$
(c) $\sigma^2(\delta_N) = \mathsf{E}(\delta_N^2) = \frac{1}{N^2} \int (\sum v(x_i))^2 \, \mathrm{d}x = \frac{1}{N} \sigma^2(g)$
Hint on (c): When the random points x_i are i.i.d. (independent identical distributed), then also $g(x_i)$ and $v(x_i)$ are i.i.d. A consequence is $\int v(x_i) v(x_j) \, \mathrm{d}x = 0$ for $i \neq j$.

2.19 (Experiment on Monte Carlo Integration)
To approximate the integral

$$\int_0^1 g(x) \, \mathrm{d}x$$

calculate a Monte Carlo sum

$$\frac{1}{N} \sum_{i=1}^{N} g(x_i)$$

for $g(x) = 5x^4$ and, for example, $N = 100,000$ random numbers $x_i \sim \mathcal{U}[0,1]$. The absolute error behaves like $cN^{-1/2}$. Compare the approximation with the exact integral for several N and seeds to obtain an estimate of c.

2.20 (Bounds on the Discrepancy)
(Compare Definition 2.16) Show

(a) $0 \le D_N \le 1$,
(b) $D_N^* \le D_N \le 2^m D_N^*$ (show this at least for $m \le 2$),
(c) $D_N^* \ge \frac{1}{2N}$ for $m = 1$.

2.21 (Algorithm for the Radical-Inverse Function)
Use the idea

$$i = \left(d_k b^{k-1} + \ldots + d_1\right) b + d_0$$

to formulate an algorithm that obtains $d_0, d_1, \ldots, d_k$ by repeated division by b. Reformulate $\phi_b(i)$ from Definition 2.18 into the form $\phi_b(i) = z/b^{j+1}$ such that the result is represented as rational number. The numerator z should be calculated in the same loop that establishes the digits $d_0, \ldots, d_k$.

Chapter 3
Monte Carlo Simulation with Stochastic Differential Equations

Sections 1.5 and 1.7.3 have introduced the principle of risk-neutral valuation for European options, which can be summarized by

$$V(S_0, 0) = \mathrm{e}^{-rT} \mathsf{E}_\mathsf{Q}(V(S_T, T) \mid S_t \text{ starting from } (S_0, 0)) ,$$

where E_Q represents the expectation under a risk-neutral measure, and $V(S_T, T)$ is given by the payoff $\Psi(S_T)$. For the Black–Scholes model, this expectation is an integral as in (1.66). This suggests two approaches to calculate the value function V: Either approximate the integral by quadrature methods, or apply Monte Carlo. This chapter is devoted to Monte Carlo methods and their role in pricing options.

To give a brief sketch of Monte Carlo (MC) methods in the option context, compare the illustration in Fig. 3.1. Monte Carlo methods calculate the expectation by simulating the underlying stochastic differential equation (SDE) repeatedly. In Fig. 3.1, five paths S_t are calculated for $0 \le t \le T$ in the risk-neutral fashion, each starting from S_0. Then for each resulting S_T the payoff is calculated, here for a European put. The figure rather hides the bulk of the work: In reality, thousands of paths are calculated. It remains the comparably cheap final task of calculating the mean of the payoffs as approximation of E_Q. This is the essence of MC. The Monte Carlo approach works for general models, for example, for pricing exotic options.

This chapter on MC is based on the ability to calculate paths—that is, to integrate SDEs numerically. Therefore a significant part of the chapter is devoted to this topic. Again X_t denotes a stochastic process and a solution of an SDE (1.44),

$$\mathrm{d}X_t = a(X_t, t)\,\mathrm{d}t + b(X_t, t)\,\mathrm{d}W_t \quad \text{for } 0 \le t \le T ,$$

where the driving process W_t is a Wiener process. We assume a t-grid with $0 = t_0 < t_1 < \ldots < t_M = T$. For convenience, the step length $\Delta t = t_{j+1} - t_j$ is taken equidistant. As is common usage in numerical analysis, we also use the h-notation, $h := \Delta t$. For $\Delta t = h = T/M$ the index j runs from 0 to $M - 1$. The solution of a

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_3

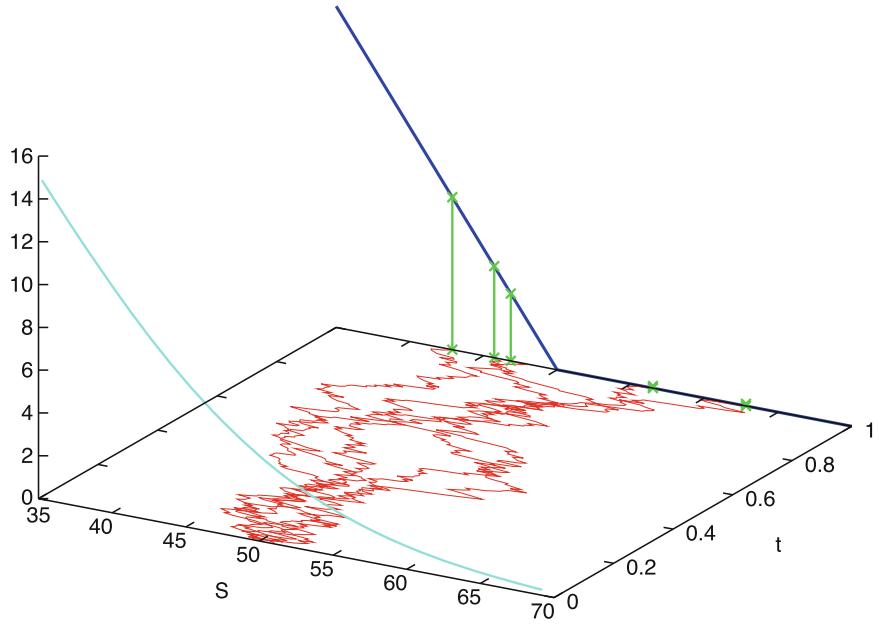

Fig. 3.1 Illustration of the Monte Carlo approach for a European put, with $K = 50$, $S_0 = 50$, $T = 1$, $\sigma = 0.2$, $r = 0$; five simulated paths S_t in the (S, t)-plane (in *red*) with payoff (*dark blue*); vertical axis: the value function V. The front curve $V(S, 0)$ is shown

discrete version of the SDE is denoted y_j. That is, y_j should be an approximation to X_{t_j}, or y_t an approximation to X_t. Weaker requirements will be discussed below. The initial value for $t = 0$ is assumed a given constant,

$$y_0 = X_0 \,.$$

For example, from Algorithm 1.11 we know the Euler discretization

$$\begin{aligned} y_{j+1} &= y_j + a(y_j, t_j)\Delta t + b(y_j, t_j)\Delta W_j \,, \quad t_j = j\Delta t \,, \\ \Delta W_j &= W_{t_{j+1}} - W_{t_j} = Z\sqrt{\Delta t} \quad \text{with } Z \sim \mathcal{N}(0,1) \,. \end{aligned} \tag{3.1}$$

Since an approximation y_T also depends on the chosen step length h, we also write y_T^h. From numerical methods for deterministic ODEs we know that for $b \equiv 0$ the discretization error of Euler's method is $O(h)$,

$$X_T - y_T^h = O(h) \,.$$

As we shall see, this will not generally hold for $b \neq 0$. Algorithm 1.11 [repeated in Eq. (3.1)] is an *explicit* method in that in every step $j \to j + 1$ the values of the

functions a and b are evaluated at the previous approximation (y_j, t_j). Evaluating b at the left-hand mesh point (y_j, t_j) is consistent with the Itô integral and the Itô process.

Chapter 2 has explained how $Z \sim \mathcal{N}(0,1)$ can be calculated, so all elements of Algorithm 1.11 are known, and we are equipped with a method to numerically integrate SDEs ($\longrightarrow$ Exercise 3.1). In this chapter we learn about other methods, and discuss the accuracy of numerical solutions of SDEs. This will be explained in Sects. 3.1 through 3.3, following [225]. Readers content with Euler's method (3.1) may like to skip these sections.

After a brief exposition on constructing bridges (Sect. 3.4), we turn to the main theme, namely, Monte Carlo methods for pricing options. The basic principle is outlined for European options (in Sect. 3.5). For American options parametric methods and regression methods are introduced in Sect. 3.6. The final Sect. 3.7 discusses the calculation of sensitivities.

3.1 Approximation Error

To study the accuracy of numerical approximations of paths, we choose as example the linear SDE

$$\mathrm{d}X_t = \alpha X_t \,\mathrm{d}t + \beta X_t \,\mathrm{d}W_t\,, \qquad \text{initial value } X_0 \text{ for } t = 0\,,$$

with constant coefficients α, β. For this GBM equation we derived in Sect. 1.8 the analytic solution

$$X_t = X_0 \exp\left(\left(\alpha - \frac{1}{2}\beta^2\right) t + \beta W_t\right)\,. \tag{3.2}$$

Given a realization of the Wiener process W_t we obtain as solution a trajectory *(sample path)* X_t. For another realization of the Wiener process the same theoretical solution (3.2) takes other values. If a Wiener process W_t is given, we call a solution X_t of the SDE a *strong solution*. In this sense the solution (3.2) is a strong solution. If one is free to select a Wiener process, then a solution of the SDE is called *weak solution*. For a weak solution (X_t, W_t), only the distribution of X_t is of interest, not its individual path.

Assuming an identical sample path of a Wiener process W_t for the SDE and for a numerical approximation y_t^h, a pathwise comparison of the trajectories X_t with y_t^h is possible for all t_j. For example, for $t_M = T$ the absolute error of a strong solution for a given Wiener process is $|X_T - y_T^h|$. For another path of the Wiener process the error is somewhat different. We average the error over "all" sample paths of the Wiener process:

Definition 3.1 (Absolute Error) For a strong solution X_t of the SDE with approximation y_t^h the absolute error at T is $\epsilon(h) := \mathsf{E}(|X_T - y_T^h|)$.

Table 3.1 Example 3.2, table of the $\hat{\epsilon}(h)$

$\hat{\epsilon}(h)$	$h = 0.01$	$h = 0.005$	$h = 0.002$	$h = 0.001$	$h = 0.0005$
Series 1 (with seed_1)	0.2825	0.183	0.143	0.089	0.070
Series 2 (with seed_2)	0.2618	0.195	0.126	0.069	0.062
Series 3 (with seed_3)	0.2835	0.176	0.116	0.096	0.065

In practice the set of all sample paths of a Wiener process is represented by N different simulations.

Example 3.2 (Euler Method) For the GBM with $X_0 = 50$, $\alpha = 0.06$, $\beta = 0.3$, $T = 1$ investigate experimentally how the absolute error of the Euler method (3.1) depends on h. Starting with a first choice of h we calculate $N = 50$ simulations and for each realization the values of X_T and y_T—that is $X_{T,k}$, $y_{T,k}$ for $k = 1, \ldots, N$. Again: to obtain pairs of comparable trajectories, also the theoretical solution (3.2) is fed with the same Wiener process used in (3.1). Then we calculate the estimate $\widehat{\epsilon}$ of the absolute error ϵ,

$$\widehat{\epsilon}(h) := \frac{1}{N} \sum_{k=1}^{N} |X_{T,k} - y_{T,k}^h| \,.$$

Such an experiment was performed for five values of h. In this way the first series of results were obtained (first line in Table 3.1). Such a series of experiments was repeated twice, using other seeds. As Table 3.1 shows, $\widehat{\epsilon}(h)$ decreases with decreasing h, but slower than one would expect from the behavior of the Euler method applied to deterministic differential equations. The order can be determined by fitting the values of the table. We bypass this little exercise, and test the order $O(h^{1/2})$ right away. To this end, divide each $\widehat{\epsilon}(h)$ of Table 3.1 by the corresponding $h^{1/2}$. This shows that the order $O(h^{1/2})$ is correct, because each entry of the table essentially leads to the same constant value, here 2.8. Apparently this example satisfies $\widehat{\epsilon}(h) \approx 2.8\, h^{1/2}$. For another example we would expect a different constant.

These results obtained for the estimates $\widehat{\epsilon}$ are assumed to be valid for ϵ. This leads to postulate

$$\epsilon(h) \leq c\, h^{1/2} = O(h^{1/2}) \,.$$

The order of convergence is worse than the order $O(h)$, which Euler's method (3.1) achieves for deterministic differential equations ($b \equiv 0$). But in view of (1.41), $(\mathrm{d}W)^2 = h$, the order $O(h^{1/2})$ is no surprise. This order holds true for SDEs whose coefficient functions a and b satisfy a couple of conditions: They must satisfy a Lipschitz condition, and grow at most linearly [225]. These assumptions are fulfilled for the GBM of Example 3.2.

Definition 3.3 (Strong Convergence) y_T^h converges strongly to X_T with order $\gamma > 0$, if $\epsilon(h) = \mathsf{E}(|X_T - y_T^h|) = O(h^\gamma)$.
y_T^h converges strongly, if

$$\lim_{h\to 0} \mathsf{E}(|X_T - y_T^h|) = 0\,.$$

Hence the Euler method applied to SDEs satisfying the above conditions converges strongly with order $1/2$. Note that convergence refers to fixed finite intervals, here for a fixed value T.

Strongly convergent methods are appropriate when the trajectory itself is of interest. This was the case for Figs. 1.17 and 1.18. Often the pointwise approximation of X_t is not our real aim but only an intermediate result in the effort to calculate a **moment**. For example, many applications in finance need to approximate $\mathsf{E}(X_T)$. A first conclusion from this situation is that of all calculated y_i only the last is required, namely, y_T. A second conclusion is that for the expectation a single sample value of y_T is of little interest. The same holds true if the ultimate interest is $\mathsf{Var}(X_T)$ rather than X_T. In this situation the primary interest is not strong convergence with the demanding requirement $y_T \approx X_T$ and even less $y_t \approx X_t$ for $t < T$. Instead the concern is the weaker requirement to approximate moments or other functionals of X_T. Then the aim is to achieve $\mathsf{E}(y_T) \approx \mathsf{E}(X_T)$, or $\mathsf{E}(|y_T|^q) \approx \mathsf{E}(|X_T|^q)$, or more general $\mathsf{E}(g(y_T)) \approx \mathsf{E}(g(X_T))$ for an appropriate function g. Recall our interest in $\mathsf{E}(\Psi(S_T))$ for a payoff function Ψ.

Definition 3.4 (Weak Convergence) y_T^h converges weakly to X_T with respect to g with order $\beta > 0$, if $\mathsf{E}(g(X_T)) - \mathsf{E}(g(y_T^h)) = O(h^\beta)$.
y_T^h converges weakly to X_T with order β, if this holds for all polynomials g.

The Euler scheme is weakly $O(h^1)$ convergent provided the coefficient functions a and b are four times continuously differentiable [225, Chap. 14]. For the special polynomial $g(x) = x$, (B.4) implies convergence of the mean $\mathsf{E}(x)$. For $g(x) = x^2$ the relation $\mathsf{Var}(X) = \mathsf{E}(X^2) - (\mathsf{E}(X))^2$ implies convergence of the variance (the reader may check). Proceeding in this way implies weak convergence with respect to all moments.

Since the properties of integration on which expectation is based lead to

$$|\mathsf{E}(X) - \mathsf{E}(Y)| = |\mathsf{E}(X - Y)| \le \mathsf{E}(|X - Y|)\,,$$

we confirm that strong convergence implies weak convergence with respect to $g(x) = x$.

When weakly convergent methods are evaluated, the outcomes X_T and y_T need not be based on the same stochastic process, only their probability distributions must be close. This allows for a simplification of Euler's method. The increments ΔW can be replaced by other random variables $\widehat{\Delta W}$ that have similar moment properties, with at least the same expectation and variance. ΔW_j can be replaced by the simple approximation $\widehat{\Delta W}_j = \pm\sqrt{\Delta t}$, where each sign occurs with probability $1/2$. The

first moments match; in particular, expectation and variance of $\widehat{\Delta W}$ and ΔW are the same:

$$\mathsf{E}(\widehat{\Delta W}) = 0\,, \quad \mathsf{E}(\widehat{\Delta W}^2) = \Delta t\,.$$

In case the replacing random variables $\widehat{\Delta W}_j$ are easier to evaluate, costs can be saved significantly ($\longrightarrow$ Exercise 3.2). The simplified Euler method is again weakly convergent with order 1 [276].

3.2 Stochastic Taylor Expansion

The derivation of algorithms for the integration of SDEs is based on stochastic Taylor expansions. To facilitate the understanding of stochastic Taylor expansions we confine ourselves to the scalar and autonomous[1] case, and first introduce the terminology by means of the deterministic case. That is, we begin with $\frac{\mathrm{d}}{\mathrm{d}t}X_t = a(X_t)$. The chain rule for arbitrary $f \in \mathcal{C}^1(\mathbb{R})$ is

$$\frac{\mathrm{d}}{\mathrm{d}t} f(X_t) = a(X_t)\frac{\partial}{\partial x} f(X_t) =: Lf(X_t)\,.$$

With the linear operator L this rule in integral form is

$$f(X_t) = f(X_{t_0}) + \int_{t_0}^{t} Lf(X_s)\,\mathrm{d}s\,. \tag{3.3}$$

The rule (3.3) is resubstituted for the integrand $\tilde{f}(X_s) := Lf(X_s)$, which requires at least $f \in \mathcal{C}^2$, and gives the term in braces:

$$\begin{aligned} f(X_t) &= f(X_{t_0}) + \int_{t_0}^{t} \left\{ \tilde{f}(X_{t_0}) + \int_{t_0}^{s} L\tilde{f}(X_z)\,\mathrm{d}z \right\} \mathrm{d}s \\ &= f(X_{t_0}) + \tilde{f}(X_{t_0}) \int_{t_0}^{t} \mathrm{d}s + \int_{t_0}^{t}\int_{t_0}^{s} L\tilde{f}(X_z)\,\mathrm{d}z\,\mathrm{d}s \\ &= f(X_{t_0}) + Lf(X_{t_0})(t - t_0) + \int_{t_0}^{t}\int_{t_0}^{s} L^2 f(X_z)\,\mathrm{d}z\,\mathrm{d}s \end{aligned}$$

[1] An *autonomous* differential equation does not explicitly depend on the independent variable, here $a(X_t)$ rather than $a(X_t, t)$. The standard GBM Model 1.13 of the stock market is autonomous for constant μ and σ.

This version of the Taylor expansion consists of two terms and the remainder as a double integral. To get the next term of the second-order derivative, apply (3.3) for $L^2 f(X_z)$, and split off the term

$$L^2 f(X_{t_0}) \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}z\, \mathrm{d}s = L^2 f(X_{t_0}) \frac{1}{2}(t - t_0)^2$$

from the remainder double integral. At this stage, the remainder is a triple integral. Repeating the procedure produces the Taylor formula in integral form. Each further step requires more differentiability of f.

Now we devote our attention to stochastic diffusion and investigate the *Itô-Taylor expansion* of the autonomous scalar SDE

$$\mathrm{d}X_t = a(X_t)\,\mathrm{d}t + b(X_t)\,\mathrm{d}W_t\,.$$

Itô's Lemma for $g(x,t) := f(x)$ is

$$\mathrm{d}f(X_t) = \underbrace{\left\{ a \frac{\partial}{\partial x} f(X_t) + \frac{1}{2} b^2 \frac{\partial^2}{\partial x^2} f(X_t) \right\}}_{=:L^0 f(X_t)} \mathrm{d}t + \underbrace{b \frac{\partial}{\partial x} f(X_t)}_{=:L^1 f(X_t)}\, \mathrm{d}W_t\,,$$

or in integral form

$$f(X_t) = f(X_{t_0}) + \int_{t_0}^{t} L^0 f(X_s)\,\mathrm{d}s + \int_{t_0}^{t} L^1 f(X_s)\,\mathrm{d}W_s\,. \tag{3.4}$$

This SDE will be applied for different choices of f. Specifically for $f(x) \equiv x$ the SDE (3.4) recovers the original SDE

$$X_t = X_{t_0} + \int_{t_0}^{t} a(X_s)\,\mathrm{d}s + \int_{t_0}^{t} b(X_s)\,\mathrm{d}W_s\,. \tag{3.5}$$

First apply (3.4) to $f = a$ and to $f = b$. The resulting versions of (3.4) are substituted in (3.5) leading to

$$\begin{aligned} X_t = X_{t_0} &+ \int_{t_0}^{t} \left\{ a(X_{t_0}) + \int_{t_0}^{s} L^0 a(X_z)\,\mathrm{d}z + \int_{t_0}^{s} L^1 a(X_z)\,\mathrm{d}W_z \right\} \mathrm{d}s \\ &+ \int_{t_0}^{t} \left\{ b(X_{t_0}) + \int_{t_0}^{s} L^0 b(X_z)\,\mathrm{d}z + \int_{t_0}^{s} L^1 b(X_z)\,\mathrm{d}W_z \right\} \mathrm{d}W_s\,. \end{aligned}$$

Summarizing the four double integrals into one remainder expression R, we have

$$X_t = X_{t_0} + a(X_{t_0}) \int_{t_0}^{t} \mathrm{d}s + b(X_{t_0}) \int_{t_0}^{t} \mathrm{d}W_s + R\,, \tag{3.6}$$

with

$$\begin{aligned} R = & \int_{t_0}^{t} \int_{t_0}^{s} L^0 a(X_z)\, \mathrm{d}z\, \mathrm{d}s + \int_{t_0}^{t} \int_{t_0}^{s} L^1 a(X_z)\, \mathrm{d}W_z\, \mathrm{d}s \\ & + \int_{t_0}^{t} \int_{t_0}^{s} L^0 b(X_z)\, \mathrm{d}z\, \mathrm{d}W_s + \int_{t_0}^{t} \int_{t_0}^{s} L^1 b(X_z)\, \mathrm{d}W_z\, \mathrm{d}W_s\,. \end{aligned} \tag{3.7}$$

The integrands are

$$\begin{aligned} L^0 a &= aa' + \tfrac{1}{2} b^2 a''\,, \qquad L^1 a = ba'\,, \\ L^0 b &= ab' + \tfrac{1}{2} b^2 b''\,, \qquad L^1 b = bb'\,. \end{aligned} \tag{3.8}$$

This is the first part of the stochastic Taylor expansion. It can be extended by terms of higher order. The order of the terms is limited by the number of repeated integrations. In view of (1.41), $\mathrm{d}W^2 = \mathrm{d}t$, we expect the last of the integrals in (3.7) to be of first order only (and show this below). To obtain higher-order terms, the integrands in (3.7) are replaced using (3.4) with appropriately chosen f, analogously as above. To organize the procedure, a formalization of the integrals is advisable. For example, double integrals

$$\begin{aligned} I_{(0,0)} &:= \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}z\, \mathrm{d}s\,, \qquad I_{(1,0)} := \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}W_z\, \mathrm{d}s\,, \\ I_{(0,1)} &:= \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}z\, \mathrm{d}W_s\,, \qquad I_{(1,1)} := \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}W_z\, \mathrm{d}W_s \end{aligned} \tag{3.9}$$

arise as factors. The first of these integrals $I_{(0,0)}$ is deterministic, of second order and elementary to integrate. The integrals $I_{(1,0)}$, $I_{(0,1)}$, $I_{(1,1)}$ are stochastic variables.[2] We shall return to $I_{(0,0)}$, $I_{(0,1)}$ and $I_{(1,0)}$ in Sect. 3.3.3, and illustrate the procedure by the integral of $f = L^1 b$, which leads to the double integral of lowest order $I_{(1,1)}$.

With $f = L^1 b$, the non-integral term of (3.4) allows to split off another "ground integral" with constant integrand,

$$R = L^1 b(X_{t_0}) \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}W_z\, \mathrm{d}W_s + \tilde{R} = L^1 b(X_{t_0}) I_{(1,1)} + \tilde{R}\,,$$

[2] In this notation, 0 stands for a deterministic integration and 1 for a stochastic integration. A general treatment of the Itô-Taylor expansion with an appropriate formalism is found in [225].

with the remaining integrals collected in $\tilde{R}$, including triple integrals. In view of (3.8) and (3.6) this result can be summarized as

$$\begin{aligned} X_t = X_{t_0} &+ a(X_{t_0}) \int_{t_0}^{t} \mathrm{d}s + b(X_{t_0}) \int_{t_0}^{t} \mathrm{d}W_s \\ &+ b(X_{t_0}) b'(X_{t_0}) \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}W_z \, \mathrm{d}W_s + \tilde{R} \,. \end{aligned} \tag{3.10}$$

The aim is to formulate numerical algorithms out of the equations derived by the stochastic Taylor expansion. To this end the integrals must be evaluated. For (3.10) we need a solution of the double integral $I_{(1,1)}$. Itô's Lemma for $X_t = W_t$ with $a = 0$, $b = 1$ and $y = g(x) := x^2$ leads to the equation

$$\mathrm{d}(W_t^2) = \mathrm{d}t + 2W_t \, \mathrm{d}W_t \,.$$

Consequently,

$$\begin{aligned} I_{(1,1)} &= \int_{t_0}^{t} \int_{t_0}^{s} \mathrm{d}W_z \, \mathrm{d}W_s = \int_{t_0}^{t} (W_s - W_{t_0}) \, \mathrm{d}W_s \\ &= \int_{t_0}^{t} W_s \, \mathrm{d}W_s - W_{t_0} \int_{t_0}^{t} \mathrm{d}W_s \\ &= \int_{t_0}^{t} \frac{1}{2} \left[\mathrm{d}(W_s^2) - \mathrm{d}s \right] - W_{t_0} (W_t - W_{t_0}) \\ &= \frac{1}{2} (W_t^2 - W_{t_0}^2) - \frac{1}{2} (t - t_0) - \frac{2}{2} W_{t_0} (W_t - W_{t_0}) \\ &= \frac{1}{2} (W_t - W_{t_0})^2 - \frac{1}{2} (t - t_0) \quad = \frac{1}{2} (\Delta W_t)^2 - \frac{1}{2} \Delta t \,. \end{aligned}$$

Specifically for $t_0 = 0$ this is the equation

$$\int_0^t \int_0^s \mathrm{d}W_z \, \mathrm{d}W_s = \int_0^t W_s \, \mathrm{d}W_s = \frac{1}{2} W_t^2 - \frac{1}{2} t \,. \tag{3.11}$$

Another derivation of (3.11) uses

$$\sum_{j=0}^{n-1} W_{t_j} (W_{t_{j+1}} - W_{t_j}) = \frac{1}{2} W_t^2 - \frac{1}{2} \sum_{j=0}^{n-1} (W_{t_{j+1}} - W_{t_j})^2$$

for $t = t_n$ and $t_0 = 0$, and takes the limit in the mean on both sides ($\longrightarrow$ Exercise 3.3). Since by (1.41) the double integral (3.11) is of order Δt, it completes the list of first-order terms in (3.10). Further terms in $\tilde{R}$ are of higher order.

The two integrals $I_{(0,1)}$ and $I_{(1,0)}$ depend on each other via the equation

$$\int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}z\,\mathrm{d}W_s + \int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}W_z\,\mathrm{d}s = \int_{t_0}^{t} \mathrm{d}W_s \int_{t_0}^{t} \mathrm{d}s \tag{3.12}$$

($\longrightarrow$ Exercise 3.4). This suggests that the two double integrals $I_{(0,1)}$ and $I_{(1,0)}$ are of order $(\Delta t)^{3/2}$. We will return to these integrals in the following section.

3.3 Examples of Numerical Methods

Now we apply the stochastic Taylor expansion to construct numerical methods for SDEs. First we check how Euler's method (3.1) evolves. Evaluating the integrals in (3.6) and substituting

$$t_0 \to t_j, \quad t \to t_{j+1} = t_j + \Delta t$$

leads to

$$X_{t_{j+1}} = X_{t_j} + a(X_{t_j})\Delta t + b(X_{t_j})\Delta W_j + R\,.$$

After neglecting the remainder R the Euler scheme of (3.1) results, here for autonomous SDEs.

To obtain higher-order methods, further terms of the stochastic Taylor expansions are added. We may expect a "repair" of the half-order $O(\sqrt{\Delta t})$ of the absolute error by including the lowest-order double integral of (3.10), which is calculated in (3.11). The resulting correction term, after multiplying with bb', is added to the Euler scheme. Discarding the remainder $\widetilde{R}$, an algorithm results, which is due to [275].

Algorithm 3.5 (Milstein)

start: $t_0 = 0,\ y_0 = X_0,\ \Delta t = T/M\,.$
loop $j = 0, 1, 2, \ldots, M-1$:
$t_{j+1} = t_j + \Delta t\,.$
Calculate the values $a(y_j),\ b(y_j),\ b'(y_j),$
$\Delta W = Z\sqrt{\Delta t}$ *with* $Z \sim \mathcal{N}(0,1)$, *and*
$$y_{j+1} = y_j + a\Delta t + b\Delta W + \frac{1}{2}bb' \cdot ((\Delta W)^2 - \Delta t).$$

This integration method by Milstein is strongly convergent with order one ($\longrightarrow$ Exercise 3.5). Adding the correction term has raised the strong convergence order of Euler's method to 1.

This holds for general SDEs with $b' \neq 0$. In the specific case $b' = 0$, the result anticipates a small error of the Euler method. This suggests an attempt to transform

the SDE of X_t via a suitable function g and $Y_t := g(X_t)$ into an SDE for Y_t such that its diffusion coefficient is a constant ($\longrightarrow$ Exercise 3.6). Then applying the Euler method to Y_t, and transforming back, leads to a reasonable approximation of X_t. In the specific case of GBM and the log-transformation, the Euler method applied to (1.62) does not suffer from discretization error. (Why?)

3.3.1 Positivity

As mentioned before, positive solutions are characteristic for many SDEs in finance, which should be preserved by numerical approximations. In this context the log transformation $Y_t := \log(X_t)$ can be helpful again. Applying the Euler method to the SDE of Y_t and subsequent back-transformation $X_t = \exp(Y_t)$ ensures the positivity of X_t.

We discuss other means to establish positivity for the CIR process, a building block of the Heston model:

$$\mathrm{d}X_t = \kappa(\theta - X_t)\,\mathrm{d}t + \sigma\sqrt{X_t}\,\mathrm{d}W_t \tag{3.13}$$

with $\kappa, \theta, \sigma > 0$, $X_0 = x_0 > 0$. Positivity of the theoretical solution X_t for all t is established by the Feller condition (see the Notes on Sect. 1.7), which guarantees a strong enough growth rate.[3] The Euler scheme

$$y_{j+1} = y_j + \kappa(\theta - y_j)\Delta t + \sigma\sqrt{y_j}\,\Delta W_j$$

with $y_0 := x_0$, works as long as $y_j \geq 0$. But there is a positive probability that y_{j+1} is negative. Simple remedies replace $\sqrt{y}$ by $\sqrt{|y|}$ or by $\sqrt{y^+}$. Then the scheme is defined for all $y \in \mathbb{R}$. Implicit Euler methods can be applied as well, for example, the drift-implicit scheme

$$y_{j+1} = y_j + a(y_{j+1})\Delta t + b(y_j)\,\Delta W_j\,. \tag{3.14}$$

If this scheme is applied to the SDE (3.13) of the square root process $\sqrt{X_t}$, then a quadratic equation for y_{j+1} results with a unique positive solution [5], see also Exercise 3.7.

[3] In view of strong convergence criteria, we remark in passing that $b(X) = \sigma\sqrt{X}$ does not satisfy a global Lipschitz condition.

3.3.2 Runge–Kutta Methods

A disadvantage of the Taylor-expansion methods is the use of the derivatives a', b', ... Analogously as with deterministic differential equations there is the alternative of Runge–Kutta–type methods, which only evaluate a or b for appropriate arguments, and not their derivatives.

As an example consider the factor bb' of Algorithm 3.5, and see how to replace it by an approximation. Starting from

$$b(y+\Delta y) - b(y) = b'(y)\Delta y + O((\Delta y)^2)$$

and using $\Delta y = a\Delta t + b\Delta W$ we deduce in view of (1.41) that

$$\begin{aligned} b(y+\Delta y) - b(y) &= b'(y)(a\Delta t + b\Delta W) + O(\Delta t) \\ &= b'(y)b(y)\Delta W + O(\Delta t)\,. \end{aligned}$$

Applying (1.41) again, we substitute $\Delta W = \sqrt{\Delta t}$ and arrive at an $O(\sqrt{\Delta t})$-approximation of the product bb', namely,

$$\frac{1}{\sqrt{\Delta t}}\left(b[y_j + a(y_j)\Delta t + b(y_j)\sqrt{\Delta t}] - b(y_j)\right)\,.$$

This expression substituted in the Milstein scheme of Algorithm 3.5 results in the variant

$$\begin{aligned} \widehat{y} &:= y_j + a\Delta t + b\sqrt{\Delta t} \\ y_{j+1} &= y_j + a\Delta t + b\Delta W + \tfrac{1}{2\sqrt{\Delta t}}(\Delta W^2 - \Delta t)[b(\widehat{y}) - b(y_j)]\,, \end{aligned} \tag{3.15}$$

which is a Runge–Kutta method, and also converges strongly with order one. Versions of these schemes for nonautonomous SDEs read analogously.

3.3.3 Taylor Scheme with Weak Second-Order Convergence

A second-order method can be achieved easily and efficiently by extrapolation. But we postpone extrapolation in order to learn how to apply the stochastic Taylor expansion in deriving a method of weak second order.

To this end, investigate the method that results when in the remainder term (3.7) the ground integrals of all double integrals are split off. This is done by applying (3.4) for $f = L^0a$, $f = L^1a$, $f = L^0b$, $f = L^1b$. Then the new remainder $\tilde{R}$ consists of triple integrals. For $f = L^1b$ this analysis was carried out at the end of

Sect. 3.2. With (3.8) and (3.11) the correction term

$$bb'\frac{1}{2}\left((\Delta W)^2-\Delta t\right)$$

has resulted, leading to the strong convergence order one of the Milstein scheme. For $f = L^0 a$ the integral is deterministic and the term

$$\left(aa'+\frac{1}{2}b^2a''\right)\frac{1}{2}\Delta t^2$$

an immediate consequence. For $f = L^1 a$ and $f = L^0 b$ the integrals are again stochastic, namely,

$$I_{(1,0)} = \int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}W_z\,\mathrm{d}s = \int_{t_0}^{t}(W_s - W_{t_0})\,\mathrm{d}s\,,$$
$$I_{(0,1)} = \int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}z\,\mathrm{d}W_s = \int_{t_0}^{t}(s - t_0)\,\mathrm{d}W_s\,.$$

Summarizing all terms into (3.6)/(3.7), the preliminary numerical scheme to obtain a weak approximation is

$$\begin{aligned} y_{j+1} = y_j &+ a\Delta t + b\Delta W + \tfrac{1}{2}bb'\left((\Delta W)^2-\Delta t\right)\\ &+ \tfrac{1}{2}\left(aa'+\tfrac{1}{2}b^2a''\right)\Delta t^2 + ba'I_{(1,0)} + \left(ab'+\tfrac{1}{2}b^2b''\right)I_{(0,1)}\,. \end{aligned} \tag{3.16}$$

It remains to approximate the two stochastic integrals $I_{(0,1)}$ and $I_{(1,0)}$. Setting $\Delta Y := I_{(1,0)}$ we have in view of (3.12)

$$I_{(0,1)} = \Delta W\Delta t - \Delta Y\,.$$

At this state the two stochastic double integrals $I_{(0,1)}$ and $I_{(1,0)}$ are expressed in terms of only one random variable ΔY, in addition to the variable ΔW used before. Since for weak convergence only the correct leading moments are needed, all occurring random variables (here ΔW and ΔY) can be replaced by other random variables with matching moments. The normally distributed random variable ΔY has expectation, variance and covariance

$$\mathsf{E}(\Delta Y) = 0\,,\ \ \mathsf{E}(\Delta Y^2) = \frac{1}{3}(\Delta t)^3,\ \ \mathsf{E}(\Delta Y\Delta W) = \frac{1}{2}(\Delta t)^2 \tag{3.17}$$

($\longrightarrow$ Exercise 3.8). Such a random variable can be realized by two independent normally distributed variates Z_1 and Z_2,

$$\begin{aligned} \Delta Y &= \tfrac{1}{2}(\Delta t)^{3/2}\left(Z_1 + \tfrac{1}{\sqrt{3}}Z_2\right) \\ \text{with} \quad & Z_i \sim \mathcal{N}(0,1), \quad i = 1,2 \end{aligned} \tag{3.18}$$

($\longrightarrow$ Exercise 3.9). With this realization of ΔY approximations of both $I_{(0,1)}$ and $I_{(1,0)}$ are available, and substituted into (3.16).

Next the random variable ΔW can be replaced by other variates for which the moments match. Choosing $\widetilde{\Delta W}$ trivalued such that the two values $\pm\sqrt{3\Delta t}$ occur with probability $1/6$, and the value 0 with probability $2/3$, then the random variable $\widetilde{\Delta Y} := \frac{1}{2}\Delta t\ \widetilde{\Delta W}$ has the moments in (3.17) up to terms of order $O(\Delta t^3)$ ($\longrightarrow$ Exercise 3.10). As a consequence, the simplification of (3.16)

$$\begin{aligned} y_{j+1} = \ & y_j + a\Delta t + b\widetilde{\Delta W} + \tfrac{1}{2}bb'\left((\widetilde{\Delta W})^2 - \Delta t\right) \\ & + \tfrac{1}{2}\left(aa' + \tfrac{1}{2}b^2a''\right)\Delta t^2 + \tfrac{1}{2}\left(a'b + ab' + \tfrac{1}{2}b^2b''\right)\widetilde{\Delta W}\Delta t \end{aligned} \tag{3.19}$$

is second-order weakly convergent.

Instead of discussing efficient ways to evaluate (3.19), we leave this second-order approach in favor of a simple extrapolation alternative: Evaluating y_t^h and y_t^{2h} with Euler's method (3.1), then the extrapolation

$$2\,\mathsf{E}(g(y_T^h)) - \mathsf{E}(g(y_T^{2h})) \tag{3.20}$$

furnishes a weakly second-order method, which is simple and efficient.

3.3.4 Higher–Dimensional Cases

In higher-dimensional cases there are additional mixed terms. We distinguish two kinds of "higher–dimensional":

(1) $y \in \mathbb{R}^n$, $a, b \in \mathbb{R}^n$. Then, for instance, replace bb' by $\frac{\partial b}{\partial y}b$, where $\frac{\partial b}{\partial y}$ is the Jacobian matrix of all first-order partial derivatives.

(2) For multiple Wiener processes the situation is more complicated, because then simple explicit integrals as in (3.11) do not exist. Only the Euler scheme remains simple: for m Wiener processes the Euler scheme is

$$y_{j+1} = y_j + a\Delta t + b^{(1)}\Delta W^{(1)} + \ldots + b^{(m)}\Delta W^{(m)}\,.$$

3.3.5 *Jump Diffusion*

Jump diffusion can be simulated analogously as pure diffusion. Thereby the jump times are not included in the equidistant grid of the $j\Delta t$. An alternative is to simulate the jump times $\tau_1, \tau_2, \ldots$ separately, and superimpose them on the Δt-size grid. Then the jumps can be carried out correctly. With such jump-adapted schemes higher accuracy can be obtained [62], see also [182].

3.4 Intermediate Values

Integration methods as discussed in the previous section calculate approximations y_j only at the grid points t_j. This leaves the question how to obtain intermediate values, namely, approximations $y(t)$ for $t \neq t_j$. For deterministic ODEs we in general have smooth solutions, which suggests to construct an interpolation curve joining the calculated points (y_j, t_j). The deterministic nature guarantees that the interpolation is reasonably close to the exact solution, at least for small steps Δt.

A smooth interpolation is at variance with the stochastic nature of solutions of SDEs. When Δt is small, it may be sufficient to match the "appearance" of a stochastic process. For example, a linear interpolation is easy to be carried out. Such an interpolating continuous polygon was used for the Figs. 1.17 and 1.18. Another easily executable alternative would be to construct an interpolating step function with step length Δt. Such an argumentation is concerned only with the graphical aspects of filling, and does not pay attention to the law given by an underlying SDE.

The situation is different when the gaps between two calculated y_j and y_{j+1} are large. Then the points that are supposed to fill the gaps should satisfy the underlying SDE. A *Brownian bridge* is a proper means to fill the gaps in Brownian motion. For illustration assume a Wiener process W_t is simulated. A stochastic process X_t is to be constructed that starts at t_0 with the value $X_0 = 0$, and ends at $t = T$ again at the value 0. The function $X_t := W_t - \frac{t}{T} W_T$ does the job ($\longrightarrow$ Exercise 3.11). More general, suppose the values y_0 (for $t = 0$) and y_T (for $t = T$) are to be connected. Then the Brownian bridge conditional on W is defined by

$$B_t = y_0 \left(1 - \frac{t}{T}\right) + y_T \frac{t}{T} + \left\{W_t - \frac{t}{T} W_T\right\} . \tag{3.21}$$

The first two terms represent a straight-line connection between y_0 and y_T representing the trend, whereas the term $W_t - \frac{t}{T} W_T$ describes the stochastic fluctuation.

Or, more general than (3.21) and Exercise 3.11,

$$B_t := \frac{(t_{i+1} - t) y_j + (t - t_j) y_{j+1}}{t_{j+1} - t_j} + \sqrt{\frac{(t_{j+1} - t)(t - t_j)}{t_{j+1} - t_j}}\, Z \tag{3.22}$$

for $Z \sim \mathcal{N}(0,1)$ realizes the Brownian bridge at t, $t_j < t < t_{j+1}$, which connects the points (y_j, t_j) and (y_{j+1}, t_{j+1}).

Bridges such as the Brownian bridge have important applications. For example, suppose that for a stochastic process S_t a large step has been taken from S_0 to some value S_T. The question may be, what is the largest value of S_t in the gap $0 < t < T$? Or, does S_t reach a certain barrier B? Of course, answers can be expected only with a certain probability. A crude method to tackle the problem would be to calculate a dense chain of S_{t_j} in the gap with a small step size Δt. This is a costly way to get the information. As an alternative, one can evaluate the relevant probabilities of the behavior of bridges directly, without explicitly constructing intermediate points. In this way, larger steps are possible, and costs are reduced. There are several alternative ways to calculate intermediate values, in particular in the multifactor case [155]. For example, the principal component analysis can be applied to approximate the bridge. Here the covariance matrix is taken from the vector $(W(t_0), \ldots, W(t_m))$, where $t_m = T$.

3.5 Monte Carlo Simulation

As pointed out in Sect. 2.4 in the context of calculating integrals, Monte Carlo is attractive in high-dimensional spaces or for nonsmooth integrands. The same characterization holds when Monte Carlo is applied to the valuation of options. For sake of clarity we describe the MC approach for European vanilla options in context with the one-dimensional Black–Scholes model. But bear in mind that MC is broadly applicable, which will be demonstrated by means of an exotic option at the end of this section.

From Sect. 1.7.2 we take the one-factor model of a geometric Brownian motion of the asset price S_t,

$$\frac{\mathrm{d}S}{S} = \mu \,\mathrm{d}t + \sigma \,\mathrm{d}W\,.$$

Here μ is the expected growth rate. When options are to be priced we assume a risk-neutral world and replace μ accordingly (by r, or by $r - \delta$ in case of a dividend yield δ, compare Sect. 1.7.3 and Remark 1.14). Recall the lognormal distribution of GBM, with density function (1.64).

The Monte Carlo simulation of options can be seen in two ways: either dynamically as a process of simulating numerous paths of prices S_t with subsequent appropriate valuation (as suggested by Fig. 3.1), or as the formal MC approximation of integrals. For the latter view we briefly recall the integral representation of options in Sect. 3.5.1. Both views are equivalent; the simulation aspect can be seen as financial interpretation and implementation of the MC procedure for integrals.

3.5.1 Integral Representation

In the one-period model of Sect. 1.5 the valuation of an option was summarized in (1.30) as the discounted value of a probable payoff,

$$V_0 = \mathrm{e}^{-rT}\mathsf{E}_\mathsf{Q}(V_T)\,.$$

For the binomial model we prove for European options in Exercise 1.10 that this method produces

$$V_0^{(M)} = \mathrm{e}^{-rT}\mathsf{E}(V_T)\,,$$

where E reflects expectation with respect to the risk-free probability of the binomial method. And for the continuous-time Black–Scholes model, the result in (1.10) or (A.17) for a put is

$$V_0 = \mathrm{e}^{-rT}\,[\,K\,F(-d_2) - \mathrm{e}^{(r-\delta)T}S\,F(-d_1)\,]\,, \tag{3.23}$$

similarly for a call. Since F is an integral (⟶ Appendix E.2), Eq. (3.23) is a first version of an integral representation. Its origin is either the analytic solution of the Black–Scholes PDE, or the representation

$$V_0 = \mathrm{e}^{-rT}\int_0^\infty (K - S_T)^+ f_{\mathrm{GBM}}(S_T, T;\, S_0, r, \sigma)\,\mathrm{d}S_T\,. \tag{3.24}$$

Here $f_{\mathrm{GBM}}(S_T, T;\, S_0, \mu, \sigma)$ is the density (1.64) of the lognormal distribution, with $\mu = r$, or μ replaced by $r - \delta$ to match a continuous dividend yield δ. It is not difficult to prove that (3.23) and (3.24) are equivalent (⟶ Exercise 3.12 for $\delta = 0$). We summarize the integral representation as

$$V(S_0, 0) = \mathrm{e}^{-rT}\mathsf{E}_\mathsf{Q}\,(\,V(S_T, T)\,|\,S_0\,)\,. \tag{3.25}$$

The risk-neutral expectation E_Q is explained in Sect. 1.5. All these expectations are conditional on paths starting at $t = 0$ with the value S_0.

We note in passing that an integral representation offers another way to approximate V_0, namely, by means of numerical quadrature methods (see Appendix C.1), rather than applying MC. Of course, in this one-dimensional situation, the approximation of the closed-form solution (3.23) is more efficient. But in higher-dimensional spaces integrals corresponding to (3.24) can be become attractive for computational purposes. Note that the integrand is smooth because the zero branch of the put's payoff $(K - S_T)^+$ needs not be integrated; in (3.24) the integration is cut to the interval $0 \le S_T \le K$. Any numerical quadrature method can be applied, such as sparse-grid quadrature [148, 309, 312]. But in what follows, we stay with Monte Carlo approximations.

3.5.2 Basic Version for European Options

The simulation aspect of Monte Carlo has been described before, see Fig. 3.1. The procedure consists in calculating a large number N of trajectories of the SDE, always starting from S_0, and then average over the payoff values $\Psi((S_T)_k)$ of the samples $(S_T)_k$, $k = 1, \ldots, N$, in order to obtain information on the probable behavior of the process. This is identical to the formal MC method for approximating an integral as (3.24), see Sect. 2.4. The equivalence with the simulation aspect is characterized by the convergence

$$\frac{1}{N}\sum_{k=1}^{N}\Psi((S_T)_k) \;\longrightarrow\; \int_{-\infty}^{\infty}\Psi(S_T)\,f_{\mathrm{GBM}}(S_T)\,\mathrm{d}S_T = \mathsf{E}(\Psi(S_T))$$

for $N \to \infty$. For GBM, the values S_T are distributed lognormally. The probability distribution of the samples $(S_T)_k$ must match the density of the chosen model, here f_{GBM}. For the Black–Scholes model, these samples are provided by integrating the correct SDE of GBM (1.47) under the risk-neutral measure. Finally, the result is discounted at the risk-free rate r to obtain the value for $t = 0$.

After having chosen the three items model, current initial value S_0, and payoff function Ψ, the Monte Carlo method works as follows:

Algorithm 3.6 (Monte-Carlo Simulation of European Options)

(1) For $k = 1, \ldots, N$: Choose a seed and integrate the SDE of the underlying model for $0 \le t \le T$ under the risk-neutral measure. (for example, $\mathrm{d}S = rS\,\mathrm{d}t + \sigma S\,\mathrm{d}W$)
Let the final result be $(S_T)_k$.

(2) By evaluating the payoff function Ψ one obtains the values

$$(V(S_T, T))_k := \Psi((S_T)_k), \quad k = 1, \ldots, N.$$

(3) An estimate of the risk-neutral expectation is

$$\widehat{\mathsf{E}}(V(S_T, T)) := \frac{1}{N}\sum_{k=1}^{N}(V(S_T, T))_k\,.$$

(4) The discounted variable

$$\widehat{V} := \mathrm{e}^{-rT}\,\widehat{\mathsf{E}}(V(S_T, T))$$

is a random variable with $\mathsf{E}(\widehat{V}) = V(S_0, 0)$, provided the estimate is unbiased.

In case the underlying receives a continuous dividend yield δ, replace the r in step (1) by $r - \delta$. (not in step (4)!) The resulting $\widehat{V}$ is the desired approximation $\widehat{V} \approx V(S_0, 0)$. In this simple form, the Monte Carlo simulation can only be applied

to European options where the exercise date is fixed. Only the value $V(S_0, 0)$ is approximated, and the lack of other information on $V(S, t)$ does not allow to check whether the early-exercise constraint of an American option is violated. For American options a greater effort in simulation is necessary, see Sect. 3.6. The convergence behavior corresponds to that discussed for Monte Carlo integration, see Sect. 2.4. In practice the number N must be chosen large, for example, $N = 10,000$. This explains why Monte Carlo simulation in general is expensive. For standard European options with univariate underlying that satisfies the Assumptions 1.3, the alternative of evaluating the Black–Scholes formula is by far cheaper. But in principle both approaches provide the same result, where we neglect that accuracies and costs are different.

For multivariate options the MC algorithm works analogously, see the example in Sect. 3.5.5. But the integration of a system of n SDEs in general has costs depending on n. So the costs of MC depend on n. In practice, this can affect the error. In case the budget in computing time is limited, which is standard for realtime calculations, a limit on the budget will limit the number N of paths, and in turn, the error. If one path costs κ seconds, and the budget for N paths is b seconds, then (2.14) states that the attainable error is of the order $\sqrt{\kappa}/\sqrt{b}$. In this sense, $\kappa = O(n)$ does influence the error.

Note that the above Algorithm 3.6 is a crude version of Monte Carlo simulation, which needs to be refined. In practical applications, methods of variance reduction are applied, see Sect. 3.5.4.

Example 3.7 (European Put) Consider a European put with parameters $S_0 = 5$, $K = 10$, $r = 0.06$, $\sigma = 0.3$, $T = 1$. For the linear SDE $\mathrm{d}S = rS\,\mathrm{d}t + \sigma S\,\mathrm{d}W$ with constant coefficients the theoretical solution is known, see Eq. (1.71). For the chosen parameters we have

$$S_1 = 5\exp(0.015 + 0.3W_1)\,,$$

which requires "the" value of the Wiener process at $t = 1$. Related values W_1 can be obtained from (1.35) with $\Delta t = T$ as $W_1 = Z\sqrt{T}$, $Z \sim \mathcal{N}(0, 1)$. The normal variates Z were generated by Algorithm 2.14, based on the input of the generator of Algorithm 2.7. For this illustration we do not take advantage of the analytic solution formula, because MC is not limited to GBM with constant coefficients. Also, we do not take advantage of a log-transformation, as suggested in Sect. 3.3.1. To demonstrate the general procedure we integrate the SDE numerically with step length $\Delta t < T$, in order to calculate an approximation to S_1. Thereby we tolerate a systematic error due to discretization. Any of the methods derived in Sect. 3.3 can be applied. For simplicity we use Euler's method. Since the chosen value of r is small, the discretization error of the drift term is small compared to the standard deviation of W_1. As a consequence, the accuracy of the integration for small values of Δt is hardly better than for larger values of the step size. Artificially we choose $\Delta t = 0.02$ for the time step. Hence each trajectory requires to calculate 50 normal variates $\sim \mathcal{N}(0, 1)$. Figure 3.2 shows the values $\widehat{V} \approx V(S_0, 0)$ for 10 sequences

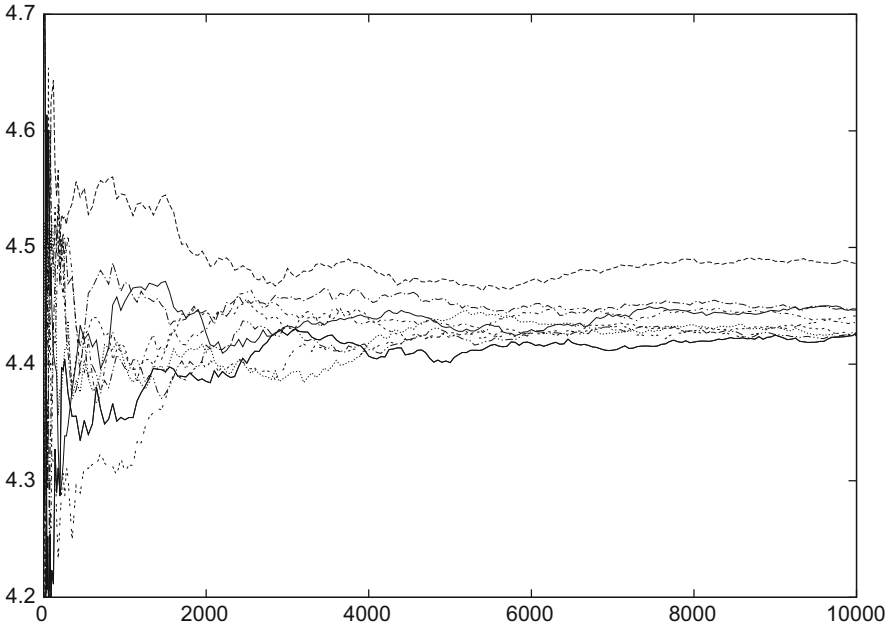

Fig. 3.2 Ten sequences of Monte Carlo simulations on Example 3.7, each with a maximum of 10,000 paths. Horizontal axis: N, vertical axis: mean value $\widehat{V}$ (suffers from bias, see Sect. 3.5.3)

of simulations, each with a maximum of $N = 10,000$ trajectories, calculated with Algorithm 3.6. Each sequence has started with a different seed for the calculation of the random numbers.

The Example 3.7 is a European put with the same parameters as Example 1.5. This allows to compare the results of the simulation with the more accurate results from Table 1.2, where we have obtained $V(5,0) \approx 4.43$. The simulations reported in Fig. 3.2 have difficulties to come close to this value. Since Fig. 3.2 depicts all intermediate results for sample sizes $N < 10,000$, the convergence behavior of Monte Carlo can be observed. For this example and $N < 2000$ the accuracy is bad; for $N \approx 6000$ it reaches acceptable values, and hardly improves for $6000 < N \leq 10,000$. Note that the "convergence" is not monotone, and one of the simulations delivers a frustratingly inaccurate result.[4] ($\longrightarrow$ Exercise 3.13)

[4] Again we emphasize that this simulation is a first attempt far from being sophisticated. In particular, the crude experimental setup suffers from a discretization error. Better MC results will be obtained below (Fig. 3.4).

Implementation of the Monte Carlo Method
As mentioned above, in case the underlying model is GBM, the analytic solution

$$S_t = S_0 \exp\left\{(r - \tfrac{1}{2}\sigma^2)t + \sigma W_t\right\}$$

can be applied. For options that are not path-dependent this requires only one random number for each path, for generating W_T and S_T. For more general models without analytic solution formula, one must resort to numerical integration [say, with Euler's method (3.1)]. Then Monte Carlo consists of two loops: the outer loop of sampling ($k = 1, \ldots, N$), and the inner loop of the integration ($j = 1, \ldots, M$), where $\Delta t = \frac{T}{M}$ is the step size of integration ($t_j = j\Delta t$). For GBM models, the analytic formula can be applied in a piecewise fashion,

$$S_{t_{j+1}} = S_{t_j} \exp\left\{(r - \tfrac{1}{2}\sigma^2)\,\Delta t + \sigma\,\Delta W\right\}$$

for all j, with $\Delta W = \sqrt{\Delta t}\,Z$, $Z \sim \mathcal{N}(0,1)$.

3.5.3 Accuracy

When in Algorithm 3.6 no theoretical solution S_T is available, the Monte Carlo method must be based on an approximation $\hat{S}_T$. In particular, $\hat{S}_T$ stands for one of the approximations y_T^h of Sect. 3.3. This causes an error

$$\epsilon := \mathrm{e}^{-rT}\hat{\mathsf{E}}(\Psi(\hat{S}_T)) - \mathrm{e}^{-rT}\mathsf{E}(\Psi(S_T))$$

in the algorithm, neglecting rounding errors. Up to the discounting factor, which is close to 1, the error ϵ can be represented as the sum of two errors,

$$\begin{aligned}\epsilon\,\mathrm{e}^{rT} &= \tfrac{1}{N}\textstyle\sum_{k=1}^{N}\Psi((\hat{S}_T)_k) - \mathsf{E}(\Psi(S_T)) \\ &= \underbrace{(\tfrac{1}{N}\textstyle\sum_{k=1}^{N}\Psi((\hat{S}_T)_k) - \mathsf{E}(\Psi(\hat{S}_T)))}_{=:\epsilon_1} + \underbrace{(\mathsf{E}(\Psi(\hat{S}_T)) - \mathsf{E}(\Psi(S_T)))}_{=:\epsilon_2}. \end{aligned} \tag{3.26}$$

The first error ϵ_1 is the statistical error. The second error ϵ_2 is the weak discretization error, which is known to us from Definition 3.4. In the MC context, this error is called *bias*. First we discuss the statistical error ϵ_1.

3.5.3.1 Statistical Error

Denote by $\hat{\mu}$ and $\hat{s}^2$ the estimates of mean and variance of a discrete sample, (B.11) in Appendix B.1, and $\mu = \mathsf{E}(\hat{\mu})$. For the MC Algorithm 3.6 $x_k := \mathrm{e}^{-rT}\Psi((S_T)_k)$,

$k = 1, \ldots, N$, or alternatively related to the approximations $\hat{S}_T$, so $\hat{\mu} = \hat{V}$ and $\mu := V(S_0, 0)$. The simulations are independent. According to the central limit theorem (⟶ Appendix B.1), the approximation $\hat{\mu}$ is $\mathcal{N}(\mu, \sigma^2)$-distributed,

$$\mathsf{P}\left(\hat{\mu} - \mu \le a\frac{\sigma}{\sqrt{N}}\right) = F(a)\,,$$

for any a, with standard normal cumulative distribution function F. In practice σ^2 is replaced by its approximation $\hat{s}^2$. Confidence intervals can be applied and a probabilistic error control incorporated.

The error behaves as $\frac{\hat{s}}{\sqrt{N}}$. To reduce this statistical error, either reduce the numerator σ or $\hat{s}$ *(variance reduction)*, or enlarge the denominator $\sqrt{N}$. The latter means to increase the number N of simulations, and is costly. As in Sect. 2.4, to gain one additional correct decimal, the error must be reduced by a factor $\xi := \frac{1}{10}$, which amounts to raise the costs by a factor of $\xi^{-2} = 100$. For variance reduction see Sect. 3.5.4.

3.5.3.2 Bias

In (3.26), $\mathsf{E}(\Psi(S_T))$ is the true value, and $\Psi(\hat{S}_T)$ an estimate. The quality of the approximation $\Psi(\hat{S}_T)$ in relation to the true value can be characterized by the bias.

Definition 3.8 (Bias) Let x be a true value to be estimated, and $\hat{x}$ be an estimate of x. Then the bias is defined by

$$\text{bias}(\hat{x}) := \mathsf{E}(\hat{x}) - x\,.$$

Unbiased $\hat{x}$ means $\text{bias}(\hat{x}) = 0$. By Definition 3.8, the error ϵ_2 in (3.26) is the bias of $\Psi(\hat{S})$. We illustrate Definition 3.8 by two further examples:

(1) For a *lookback* option the valuation depends on

$$x := \mathsf{E}\left(\max_{0\le t\le T} S_t\right)\,.$$

The discretely sampled

$$\hat{x} := \max_{0\le j\le M} S_{t_j}$$

is an approximation based on a finite number M of S_{t_j}-values. The estimator $\hat{x}$ almost surely underestimates the true value x, $\mathsf{E}(\hat{x}) < x$, hence the bias of $\hat{x}$ is nonzero. Such examples typically occur when the option is path-dependent—that is, its value depends on S_t for possibly all $t \le T$.

(2) Euler's method (3.1) suffers from bias. For GBM (1.47) the Euler step

$$S_{t+\Delta t} = S_t\,(1 + \mu\,\Delta t + \sigma\,\Delta W)$$

is biased, whereas the analytic step

$$S_{t+\Delta t} = S_t\,\exp[(\mu - \tfrac{1}{2}\sigma^2)\,\Delta t + \sigma\,\Delta W]$$

is unbiased. The bias from Euler steps producing $\hat{S}_T$ explains the inaccurate result in Fig. 3.2. Using the unbiased analytic solution produces better results; compare Figs. 3.2 and 3.3 for results with and without bias.

Fortunately, when sufficient computing time is available, the bias of the above examples can be made arbitrarily small by taking sufficiently large values of M. That is, asymptotically, for infinite costs, the results will be unbiased. There is a tradeoff between making the variance small ($N \to \infty$), and making the bias small ($M \to \infty$, $\Delta t \to 0$). The *mean square error*

$$\mathrm{MSE}(\hat{x}) := \mathsf{E}[(\hat{x} - x)^2] \tag{3.27}$$

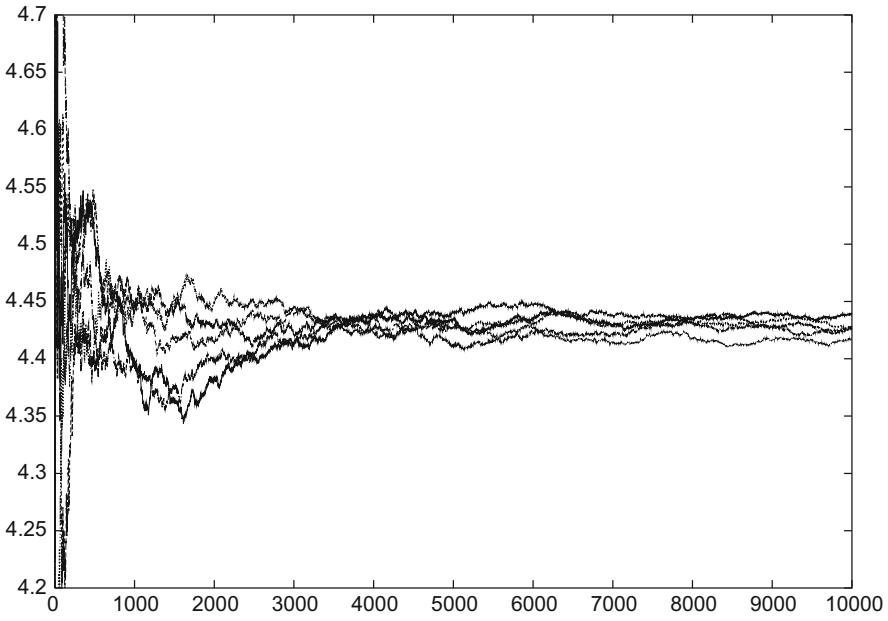

Fig. 3.3 Five series of unbiased Monte Carlo simulations on Example 3.7, using the analytic solution of the SDE (compare to Fig. 3.2)

measures both errors: A straightforward calculation (which the reader may check) shows

$$\begin{aligned} \mathrm{MSE}(\hat{x}) &= (\mathsf{E}(\hat{x}) - x)^2 + \mathsf{E}[(\hat{x} - \mathsf{E}(\hat{x}))^2] \\ &= (\mathrm{bias}(\hat{x}))^2 + \mathsf{Var}(\hat{x})\,. \end{aligned} \tag{3.28}$$

The final aim is to make MSE small, and the investigator must balance the effort in controlling the bias or the sampling error.

We outline this for a Monte Carlo approximation that makes use of a numerical integration scheme such as Euler's method. For brevity, write again h for the step Δt, and $\hat{x} := y_T^h$ for the result of a weakly convergent discretization scheme, see Definition 3.4. Assume the bias of the discretization is of the order β,

$$\mathrm{bias}(\hat{x}) = \alpha_1 h^\beta\,, \quad \alpha_1 \text{ a constant.}$$

Since the variance of Monte Carlo is of the order N^{-1} [N the sample size, see (2.14)], (3.28) leads to model the mean square error as

$$\mathrm{MSE} = \alpha_1^2 h^{2\beta} + \frac{\alpha_2}{N}$$

for some constant α_2. This error model allows to analyze the tradeoff ($N \to \infty$ or $h \to 0$) more closely ($\longrightarrow$ Exercise 3.14). It turns out that for optimally chosen h, N, the error $\sqrt{\mathrm{MSE}}$ behaves like

$$\sqrt{\mathrm{MSE}} \sim C^{-\frac{\beta}{1+2\beta}}$$

where C denotes the costs of the approximation. Applying Euler's method ($\beta = 1$) gives the exponent $-1/3$, clearly worse than the exponent $-1/2$ of an unbiased Monte Carlo. This result emphasizes the importance of high-order schemes ($\beta > 1$) for high demands of accuracy.

3.5.4 Variance Reduction

To improve the accuracy of simulation and thus the efficiency, it is essential to apply methods of variance reduction. We explain the methods of the *antithetic variates* and the *control variate*. In many cases these methods decrease the variance.

3.5.4.1 Antithetic Variates

If a random variable satisfies $Z \sim \mathcal{N}(0,1)$, then also $-Z \sim \mathcal{N}(0,1)$. Let $\widehat{V}$ denote the approximation obtained by Monte Carlo simulation. With little extra effort

during the original Monte Carlo simulation we can run in parallel a side calculation which uses $-Z$ instead of Z. For each original path this creates a "partner" path, which looks like a mirror image of the original. The partner paths also define a Monte Carlo simulation of the option, called the *antithetic variate*, here denoted by V^-. The average

$$V_{\mathrm{AV}} := \frac{1}{2}\left(\widehat{V} + V^-\right) \tag{3.29}$$

(AV for *antithetic variate*) is a new approximation, which in many cases is more accurate than $\widehat{V}$. Since $\widehat{V}$ and V_{AV} are random variables we can only aim at

$$\mathsf{Var}(V_{\mathrm{AV}}) < \mathsf{Var}(\widehat{V})\,.$$

In view of the properties of variance and covariance [Eq. (B.7)],

$$\begin{aligned} \mathsf{Var}(V_{\mathrm{AV}}) &= \tfrac{1}{4}\mathsf{Var}(\widehat{V} + V^-) \\ &= \tfrac{1}{4}\mathsf{Var}(\widehat{V}) + \tfrac{1}{4}\mathsf{Var}(V^-) + \tfrac{1}{2}\mathsf{Cov}(\widehat{V}, V^-)\,. \end{aligned} \tag{3.30}$$

From

$$|\mathsf{Cov}(X, Y)| \le \frac{1}{2}[\mathsf{Var}(X) + \mathsf{Var}(Y)]$$

[follows from (B.7)] we deduce

$$\mathsf{Var}(V_{\mathrm{AV}}) \le \frac{1}{2}(\mathsf{Var}(\widehat{V}) + \mathsf{Var}(V^-))\,.$$

By construction, $\mathsf{Var}(\widehat{V}) = \mathsf{Var}(V^-)$ should hold. Hence $\mathsf{Var}(V_{\mathrm{AV}}) \le \mathsf{Var}(\widehat{V})$. This shows that in the worst case only the efficiency is slightly deteriorated by the additional calculation of V^-. The favorable situation is when the covariance is negative. Then (3.30) shows that the variance of V_{AV} can become significantly smaller than that of $\widehat{V}$.

In our context, the anti-symmetric construction of the mirror paths inspires some confidence that the results are negatively correlated, $\mathsf{Cov}(\hat{V}, V^-) < 0$. This holds true in case the dependence of the output V on the input Z is monotonic. Then the negative correlation between Z and $-Z$ is carried over to the corresponding V-values. For example, for GBM the solution (1.71)

$$S_T = S_0 \exp\left\{\left(r - \frac{\sigma^2}{2}\right)T + \sigma\sqrt{T}\,Z\right\}$$

(for $t_0 = 0$) is monotonic in Z, but this does not necessarily hold for any arbitrary payoff $\Psi(S_T)$.[5] But when $\mathsf{Cov}(\hat{V}, V^-) < 0$, then the effect is

$$\mathsf{Var}(V_{\mathrm{AV}}) < \frac{1}{2}\mathsf{Var}(\hat{V}),$$

and V_{AV} is a better approximation than $\widehat{V}$. This approach at most doubles the costs. In comparison, an error reduction of this size by merely increasing N requires at least fourfold costs.

3.5.4.2 Application to GBM

Let the index k in V_k number an MC simulation of a GBM, $k = 1, \ldots, N$. For a payoff Ψ monotonic in Z draw $Z_k \sim \mathcal{N}(0,1)$ and calculate the pairs $\hat{V}_k, V_k^-$ and the antithetic variate $V_{\mathrm{AV},k}$ as follows:

$$\begin{aligned} \hat{V}_k &= \Psi\left(S_0 \exp\left\{(r - \tfrac{\sigma^2}{2})T + \sigma\sqrt{T}\,Z_k\right\}\right) \\ V_k^- &= \Psi\left(S_0 \exp\left\{(r - \tfrac{\sigma^2}{2})T - \sigma\sqrt{T}\,Z_k\right\}\right) \\ V_{\mathrm{AV},k} &= \tfrac{1}{2}(\hat{V}_k + V_k^-) \end{aligned} \tag{3.31}$$

For each k, $\hat{V}_k$ and V_k^- are dependent, but the independence of Z_k for $k = 1, \ldots, N$ makes the $V_{\mathrm{AV},k}$ independent, and MC is applied: The mean of the $V_{\mathrm{AV},k}$, discounted with the factor e^{-rT}, approximates V. Variance reduction by antithetic variates may not be too effective, but is implemented easily.

In Fig. 3.4 we simulate Example 3.7 again, now with antithetic variates. With this example and the chosen random number generator[6] the variance reaches small values already for small N. Compared to Fig. 3.2 the convergence is somewhat smoother. The accuracy the experiment shown in Fig. 3.2 reaches with $N = 6000$ is achieved already with $N = 2000$ in Fig. 3.4. But in the end, the error has not become really small. The main reason for the remaining significant error in the experiment reported by Fig. 3.4 is the bias due to the discretization error of the Euler scheme. To remove this source of error, we repeat the above experiments with the analytic solution of (1.71). The result is shown in Fig. 3.5 for MC with antithetic variates. Figure 3.5 better reflects the convergence behavior and the potential of Monte Carlo simulation. By the way, applying Milstein's scheme of Algorithm 3.5 does not improve the picture: No qualitative change is visible if we replace the Euler-generated simulations by their Milstein counterparts. This is explained by the fact that the weak convergence order of Milstein's method equals that of the

[5] It does hold for the standard put and call with monotonic Ψ, but not for a butterfly option.

[6] The simple generator of Algorithm 2.7.

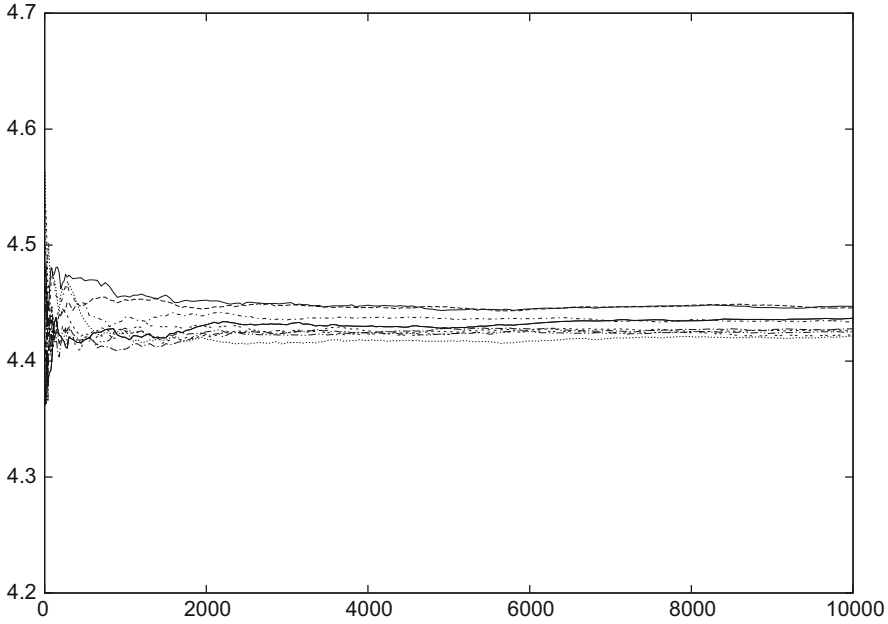

Fig. 3.4 Ten series of antithetic simulations on Example 3.7, using Euler steps (biased)

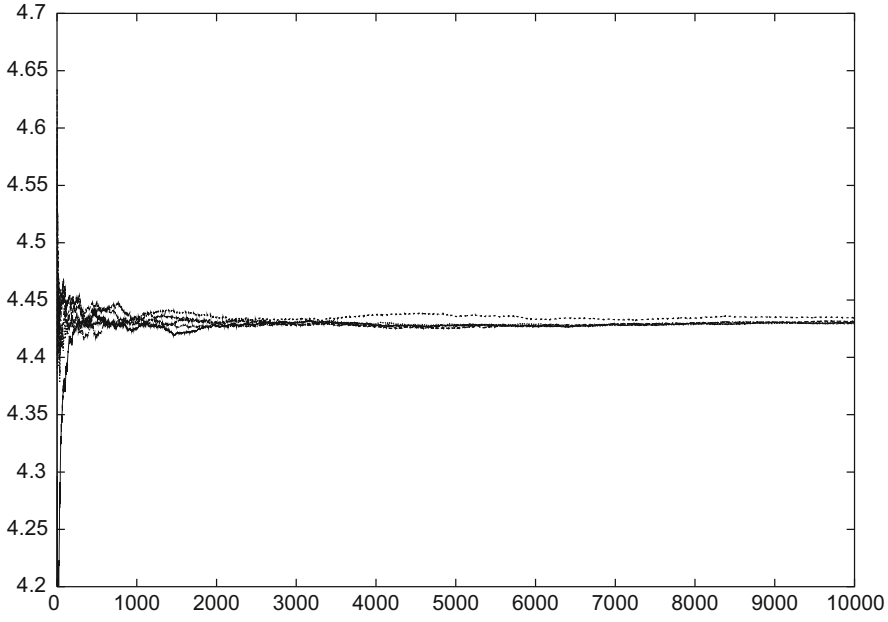

Fig. 3.5 Five series of unbiased Monte Carlo simulations on Example 3.7, unbiased using the analytic solution of the SDE and antithetic variates (3.29) (compare to Fig. 3.4)

Euler method. Recall that Example 3.7 is chosen merely for illustration; here other methods are more efficient than Monte Carlo approaches.

3.5.4.3 Control Variates

Again V denotes the exact value of the option and $\widehat{V}$ a Monte Carlo approximation. For error control, we calculate in parallel another option, which is closely related to the original option, and for which we know the exact value V^*. Let the Monte Carlo approximation of V^* be denoted $\widehat{V}^*$, so $V^* = \mathsf{E}(\widehat{V}^*)$. The variate $\widehat{V}^*$ serves as *control variate* with which we wish to "control" the error. The additional effort to calculate the control variate $\widehat{V}^*$ is small in case the simulations of the asset S are identical for both options. This situation arises, for example, when S_0, μ and σ are identical and only the payoff differs. When the two options are similar enough one may expect a strong positive correlation between them. So we expect relatively large values of $\mathsf{Cov}(V, V^*)$ or $\mathsf{Cov}(\widehat{V}, \widehat{V}^*)$, close to its upper bound,

$$\mathsf{Cov}(\widehat{V}, \widehat{V}^*) \approx \frac{1}{2}\mathsf{Var}(\widehat{V}) + \frac{1}{2}\mathsf{Var}(\widehat{V}^*)\,.$$

This leads us to define "closeness" between the options as sufficiently large covariance in the sense

$$\mathsf{Cov}(\widehat{V}, \widehat{V}^*) > \frac{1}{2}\mathsf{Var}(\widehat{V}^*)\,. \tag{3.32}$$

The method is motivated by the assumption that the unknown error $V - \widehat{V}$ has the same order of magnitude as the known error $V^* - \widehat{V}^*$. The bold anticipation $V - \widehat{V} \approx V^* - \widehat{V}^*$ leads to try

$$V_{\mathrm{CV}} := \widehat{V} + (V^* - \widehat{V}^*) \tag{3.33}$$

as another approximation (CV for *control variate*). We see from (B.6) (with $\beta = V^*$) and (B.7) that

$$\mathsf{Var}(V_{\mathrm{CV}}) = \mathsf{Var}(\widehat{V} - \widehat{V}^*) = \mathsf{Var}(\widehat{V}) + \mathsf{Var}(\widehat{V}^*) - 2\mathsf{Cov}(\widehat{V}, \widehat{V}^*)\,.$$

If (3.32) holds, then $\mathsf{Var}(V_{\mathrm{CV}}) < \mathsf{Var}(\widehat{V})$. In this sense $\mathsf{Var}(V_{\mathrm{CV}})$ is a better approximation than $\widehat{V}$.

To improve the approach, allow for a general linear relation between $V - \widehat{V}$ and $V^* - \widehat{V}^*$. This leads to define

$$V_{\mathrm{CV}}^{\alpha} := \widehat{V} + \alpha(V^* - \widehat{V}^*)\,, \tag{3.34}$$

instead of (3.33). The parameter α is chosen such that the variance $\mathsf{Var}(V_{\mathrm{CV}}^{\alpha})$ is minimal ($\longrightarrow$ Exercise 3.15). In [219] the control variate $\hat{V}^*$ in (3.33) itself is constructed by using MC on the same model, based on the same discretization scheme (Euler) but different step size. Further improvements have lead to multilevel Monte Carlo [151].

3.5.5 *Application to an Exotic Option*

As mentioned before, the error of Monte Carlo methods basically does not vary with the number of underlyings. As an example we choose a two-factor binary put to illustrate that MC can be applied as easily as in a one-factor situation.

Assume that two underlying assets $S_1(t), S_2(t)$ obey a two-dimensional GBM,

$$\begin{aligned} \mathrm{d}S_1 &= S_1\,(\mu_1\,\mathrm{d}t + \sigma_1\,\mathrm{d}W^{(1)}) \\ \mathrm{d}S_2 &= S_2\,(\mu_2\,\mathrm{d}t + \sigma_2\,(\rho\,\mathrm{d}W^{(1)} + \sqrt{1-\rho^2}\,\mathrm{d}W^{(2)}))\,. \end{aligned} \tag{3.35}$$

This makes use of Exercise 2.17: $W^{(1)}$ and $W^{(2)}$ are two uncorrelated Wiener processes, and the way they interact in (3.35) establishes a correlation ρ between S_1 and S_2. The analytic solution of (3.35) is given by

$$\begin{aligned} S_1(T) &= S_1(0)\exp\left((\mu_1 - \tfrac{1}{2}\sigma_1^2)T + \sigma_1 W^{(1)}(T)\right) \\ S_2(T) &= S_2(0)\exp\left((\mu_2 - \tfrac{1}{2}\sigma_2^2)T + \sigma_2(\rho W^{(1)}(T) + \sqrt{1-\rho^2}\,W^{(2)}(T))\right)\,, \end{aligned} \tag{3.36}$$

which generalizes (1.71).

Example 3.9 (2D European Binary Put) A two-asset cash-or-nothing put pays the fixed cash amount c in case

$$S_1(T) < K_1 \quad \text{and} \quad S_2(T) < K_2\,.$$

We choose the parameters $T = 1$, $K_1 = K_2 = 5$, $\sigma_1 = 0.2$, $\sigma_2 = 0.3$, $\rho = 0.3$, $c = 1$, $r = 0.1$; no dividends, so the "costs of carry" are taken as $\mu_1 = \mu_2 = r$. For $t = 0$ the value $V(S_1, S_2, 0)$ is to be evaluated at $S_1(0) = S_2(0) = 5$.
Figure 3.6 illustrates both the payoff of this exotic option and the Monte Carlo approach. The top figure depicts the box characterizing the payoff. Further, two paths starting at $S_1(0) = S_2(0) = 5$ are drawn. For $t = T$, one of the paths ends inside the box; accordingly the payoff value there is $V = c = 1$. The other path terminates "outside the strike," the payoff value is zero. Since we have the analytic solution (3.36), no paths need to be calculated. Rather, terminal points $(S_1(T), S_2(T))$ are evaluated by (3.36). The lower figure in Fig. 3.6 shows

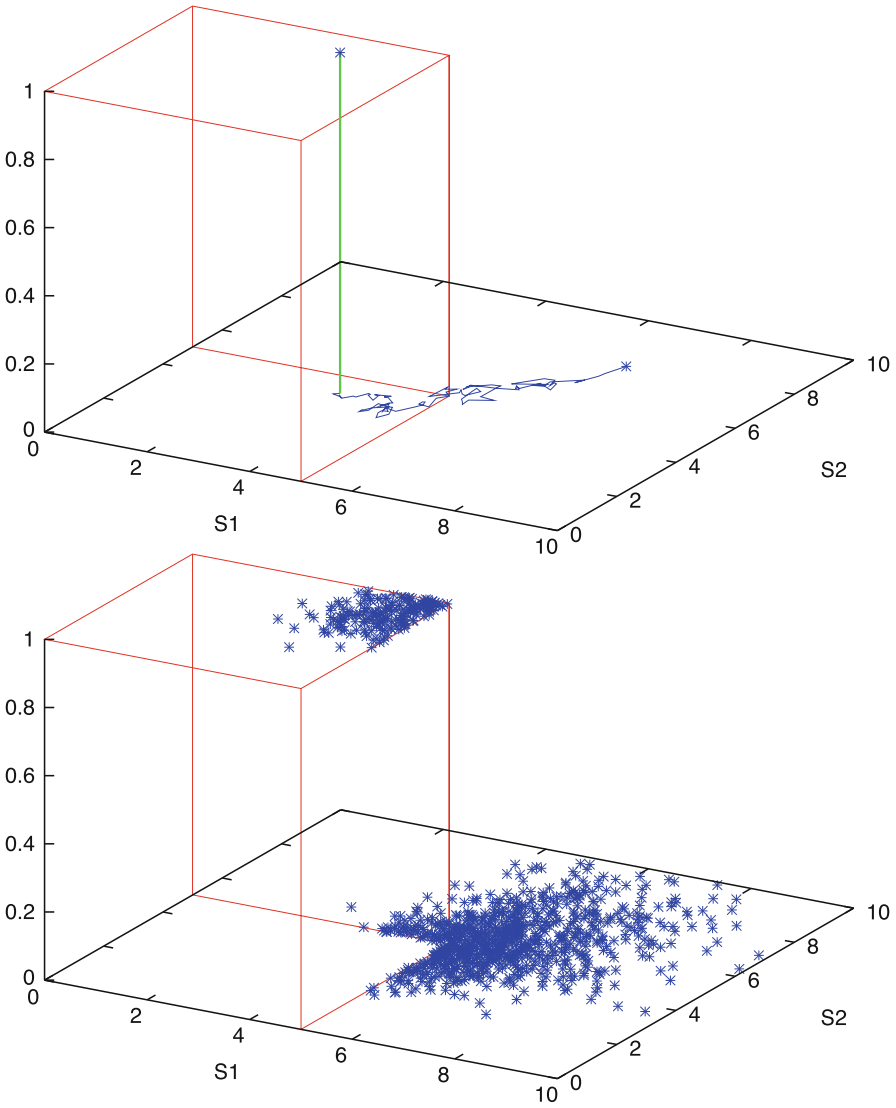

Fig. 3.6 Example 3.9, binary option on two assets. Horizontal: (S_1, S_2)-plane, vertical: $V(S_1, S_2)$; *top*: two paths starting at $S_1 = S_2 = 5$ with their payoff values; *bottom*: $N = 1000$ terminal points with their payoff values

1000 points calculated in this way. Taking the mean value and discounting as in Algorithm 3.6, yields approximations to $V(5, 5, 0)$. With $N = 10^5$ simulations we obtain

$$V(5, 5, 0) \approx 0.174\,,$$

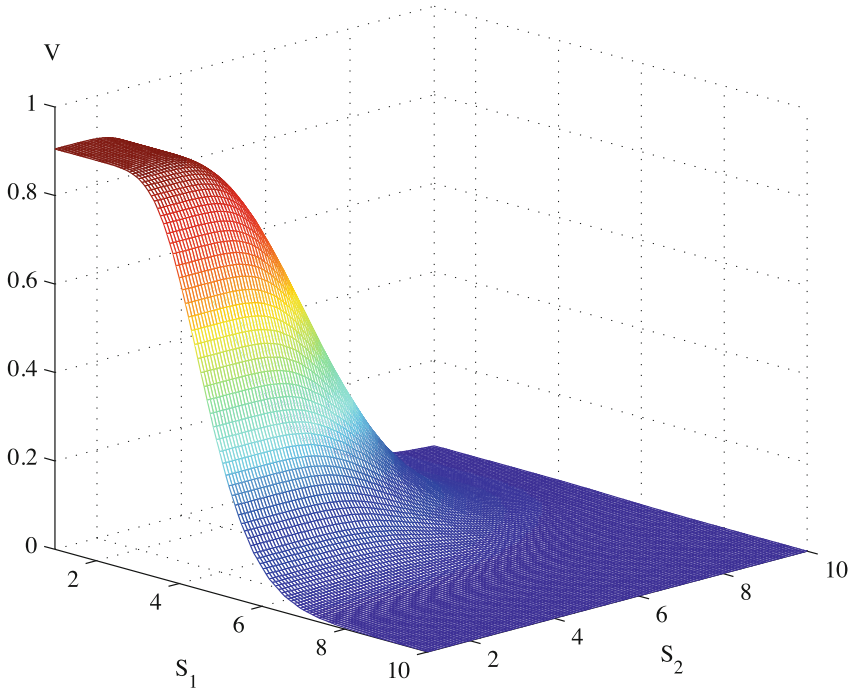

Fig. 3.7 Example 3.9: surface $V(S_1, S_2, 0)$ calculated by Algorithm 1.18. With kind permission of Sebastian Quecke

using random numbers based on the simple generator of Algorithm 2.7. The accuracy is almost three digits.[7] Using Euler's method rather than the analytic solution, Example 3.9 offers nice possibilities to conduct empirical studies in controlling either the bias or the sample error. We conclude Example 3.9 with Fig. 3.7, which depicts the entire surface $V(S_1, S_2, 0)$, calculated with Algorithm 1.18 [309].

3.5.6 Test with Halton Points

To complete this introduction into MC methods, we test the Monte Carlo simulation in a fully deterministic variant. To this end we insert the quasi-random two-dimensional Halton points into the polar Algorithm 2.14 and use the resulting quasi normal deviates to calculate solutions of the SDE. In this way, for Example 3.7 an acceptable accuracy is reached already with about 2000 paths, much better than what is shown in the experiments reported by Figs. 3.2 or 3.3.

[7] This example has an analytic solution based on bivariate distribution functions, see [172].

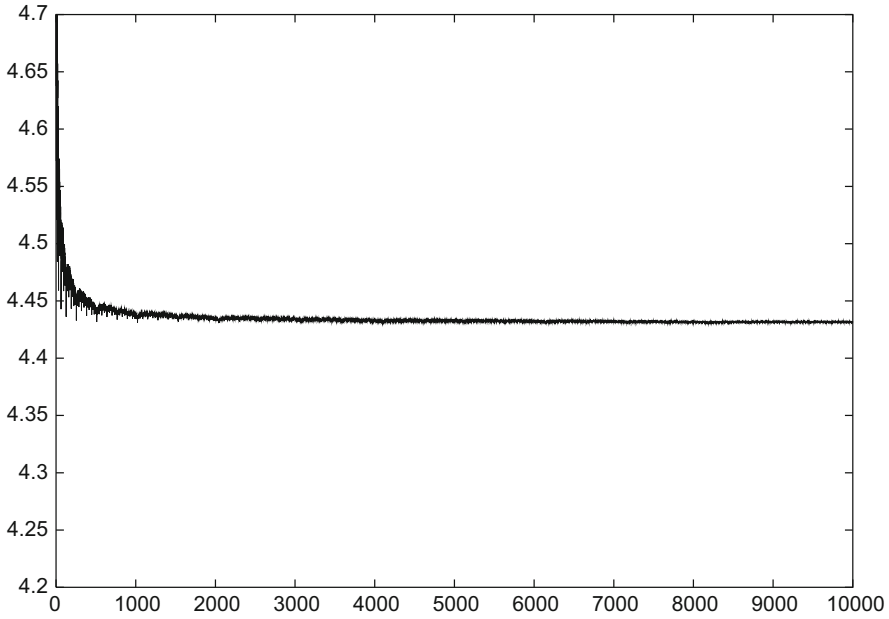

Fig. 3.8 Quasi Monte Carlo applied to Example 3.7

A closer investigation reveals that quasi normal deviates based on Box-Muller-Marsaglia (Algorithm 2.14) with two-dimensional Halton points lose the equidistributedness; the low discrepancy is not preserved. Apparently the quasi-random method does not simulate independence [147]. This sets the stage for the slightly faster inversion method [278] (⟶ Appendix E.2), based on one-dimensional low-discrepancy sequences. Figure 3.8 shows the result. The scaling of the figure is the same as before.

3.6 Monte Carlo Methods for American Options

The integral representation (3.25) can be generalized to American options. Similar as for European options, Monte Carlo applied to American options requires simulating paths S_t of the underlying model. Again, for ease of exposition, we think of the prototype example of the univariate Black–Scholes model where we integrate $\mathrm{d}S_t = rS_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t$ for $t \geq 0$. Whereas for European options it is clear to integrate until expiration, $t = T$, the American option requires to continuously investigate whether early exercise is advisable.

3.6.1 Stopping Time

For motivation, think of a price limit β of an asset, such that a stop-buy order is to be carried out at level β. This decision is prompted by the event that S_t reaches β for some "stopping time." Or, for the life of an American option, the decisive event is "early exercise," which amounts to a "stop" in holding the option. Such decisions are based on a specified event, triggered, for example, by reaching a threshold value. A rational decision follows a *stopping rule* that defines the *stopping time*, denoted τ.

Since a stopping time τ depends on the underlying process S_t, it is a stochastic process. Imagine we travel along the path of a specific realization of a stochastic process S_t and look up at the event that defines τ. In this way we get a realization of the random variable τ; for each path obtain another value of τ. If the event specified by the stopping rule does not happen, then set $\tau := \infty$. To mimic reality, one must take care that for any t the decision (on early exercise, for example) is *only based on the information that is known up to the present moment.* This situation suggests defining a stopping time to be not anticipating. A stopping time is not allowed to glimpse into the future. This characterization leads to an informal definition of τ:

Definition 3.10 (Stopping Time I) For any time t there must be certainty whether the event has happened—that is, whether $\tau \leq t$ or $\tau > t$.

A formal definition with the means of stochastics uses the concept of a filtration. A stochastic process S_t builds a natural filtration $\mathcal{F}_t$, which is interpreted as a model of the information available at time t ($\longrightarrow$ Appendix B.2). Accordingly, for a stopping time τ we require $\{\tau \leq t\} \in \mathcal{F}_t$ for all $t \geq 0$, where the set $\{\tau \leq t\}$ represents all decisions until time t. That is:

Definition 3.11 (Stopping Time II) A stopping time τ with respect to a filtration $\mathcal{F}_t$ is a random variable that is $\mathcal{F}_t$-measurable for all $t \geq 0$.

Example 3.12 (Hitting Time) For a value $\beta > S_0$, which fixes a level of S, define $\tau := \inf\{t > 0 \mid S_t \geq \beta\}$, and $\tau := \infty$ if such a t does not exist.
This example, illustrated in Fig. 3.9, fulfills the requirements of a stopping time. Setting τ amounts to set a flag as soon as S_t reaches β, and by checking the flag one

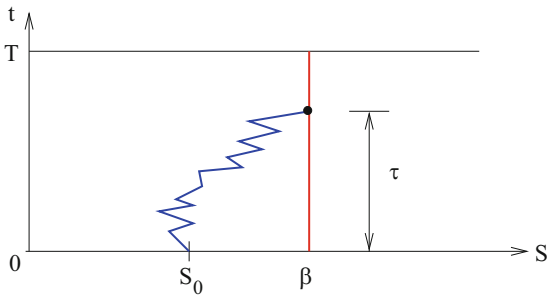

Fig. 3.9 Hitting time: the strategy of Example 3.12 to define a stopping time τ

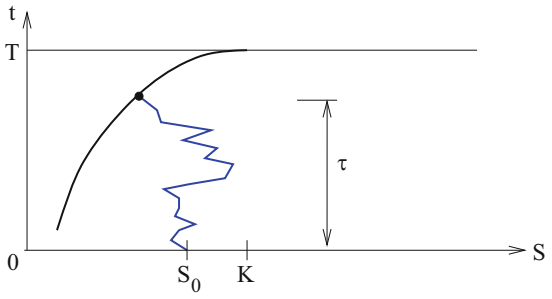

Fig. 3.10 The optimal stopping time τ of a vanilla put. The heavy curve is the early-exercise curve, and the zigzag symbolizes a path S_t

can verify at any time whether the event has happened.[8] This defines the stopping rule, "stop when S_t has reached β."

Two further examples should make the concept of a stopping time clearer.

(1) The example

$$t^* := \text{ moment when } S_t \text{ reaches its maximum over } 0 \le t \le T$$

is no stopping time, because for each $t < T$ it can not be decided whether $t^* \le t$ or $t^* > t$; it is not possible to decide whether to stop.

(2) For American-style options, define

$$\tau := \min\{\, t \le T \mid (t, S_t) \in \text{ stopping area} \,\}.$$

This example is similar to Example 3.12. The stopping time is defined by the moment when a path S_t hits the early-exercise curve, which is the boundary of the stopping region.[9]

This sets the stage for American options. Of all possible stopping times, the stopping at the early-exercise curve is optimal (illustrated in Fig. 3.10). Optimal stopping gives the American option its optimal value. From a practical point of view, the stopping at the early-exercise curve can not be established as simply as in Example 3.12, because the curve is not known initially.

The holder of an American option is free to decide when to stop, and hence picks the τ that maximizes the payoff. Or seen from the market view: Since one does not know when the holder of the option will exercise, its price reflects the worst possible case. This amounts to the following characterization of the value $V(S, 0)$ of

[8] For a more formal proof see [193, p. 42], or [340, p. 341].

[9] The early exercise curve will be the topic of Sect. 4.5. We briefly touched this in end of Sect. 1.4.4.

an American option:

$$V(S,0) = \sup_{0\le\tau\le T} \mathsf{E}_{\mathsf{Q}}\left(\mathrm{e}^{-r\tau}\,\Psi(S_\tau) \mid S_0 = S\right), \quad \text{where } \tau \text{ is a stopping time and } \Psi \text{ is the payoff.} \tag{3.37}$$

Thereby τ is stopping time with respect to the natural filtration of S_t. This result (3.37) is a special case for $t = 0$ of the general formula for $V(S,t)$, which is proved in [35]. Clearly, (3.37) includes the case of a European option for $\tau := T$, in which case taking the supremum is not effective.

3.6.2 Parametric Methods

A practical realization of (3.37) leads to calculating lower bounds $V^{\mathrm{low}}(S,0)$ and upper bounds $V^{\mathrm{up}}(S,0)$ such that

$$V^{\mathrm{low}}(S,0) \le V(S,0) \le V^{\mathrm{up}}(S,0)\,. \tag{3.38}$$

Since by (3.37) $V(S,0)$ is given by taking the supremum over *all* stopping times, a lower bound is obtained by taking a *specific* stopping rule. To illustrate the idea, choose the stopping rule of Example 3.12 with a level β, see Fig. 3.9. If we denote for each calculated path the resulting stopping time by $\tilde\tau = \tilde\tau(\beta)$, a lower bound to $V(S,0)$ is given by

$$V^{\mathrm{low}(\beta)}(S,0) := \mathsf{E}_{\mathsf{Q}}\left(\mathrm{e}^{-r\tilde\tau}\,\Psi(S_{\tilde\tau}) \mid S_0 = S\right). \tag{3.39}$$

This value depends on the parameter β, which is indicated by writing $V^{\mathrm{low}(\beta)}$. The bound is calculated by Monte Carlo simulation over a sample of N paths, where the paths are stopped according to the chosen stopping rule. Procedure and costs of such a simulation for one value of β are analogous as in Algorithm 3.6. Repeating the experiment for another value of β may produce a better (larger) value $V^{\mathrm{low}(\beta)}$.

It is difficult to get a tolerable accuracy when working with only a single parameter β. The situation can be slightly improved by choosing a finishing curve different from the line in Fig. 3.9. For example, think of a curve that approximates the early-exercise curve, as in Fig. 3.10. A simple but nicely working approximation uses a parabola in the (S,t)-domain with horizontal tangent at $t = T$. Hitting this curve defines a stopping time. Again this crude approach requires only one parameter β ($\longrightarrow$ Exercise 3.16). A result of this approach is illustrated in Fig. 3.11.

There are many examples how to obtain better lower bounds. For instance, the early-exercise curve can be approximated by pieces of curves or pieces of straight lines, which are defined by several parameters; β then symbolizes a vector of

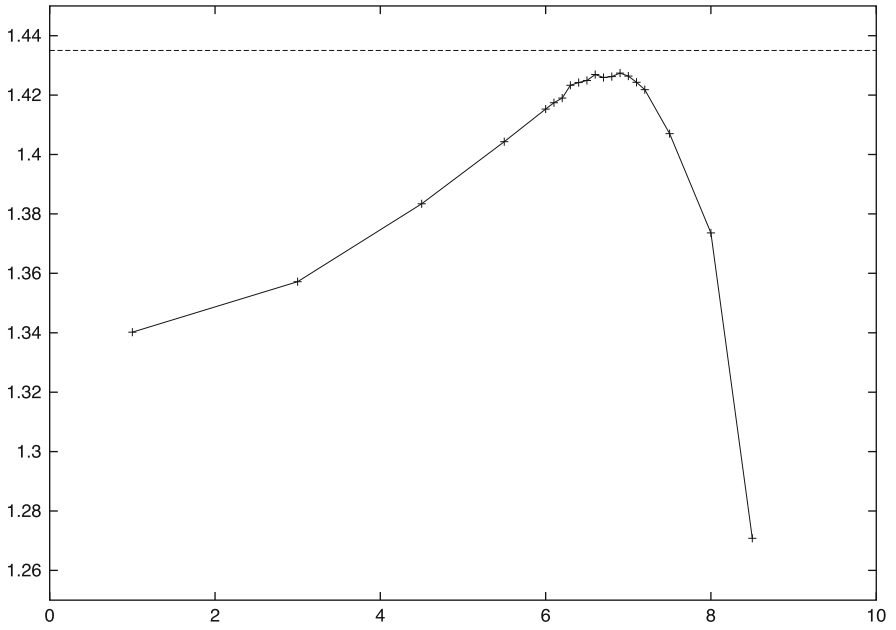

Fig. 3.11 Monte Carlo approximations $V^{\text{low}(\beta)}(S,0)$ (+) for several values of β and $S = 9$ (Exercise 3.16, random numbers calculated with the generator from [264]). The approximations are not close to the exact value $V(S,0)$, which is represented by the *dashed line*. A systematic error remains because a parabola cannot approximate the early-exercise curve well. Here the approximating parabola cuts the S-axis close to $\beta = 7$

parameters. The idea is to optimize in the chosen parameter space, trusting that

$$\sup_{\beta} V^{\text{low}(\beta)} \approx V.$$

As illustrated by Fig. 3.11, the corresponding surface to be maximized over β is not smooth. Accordingly, an optimization in the parameter space is costly, see Appendix C.4. Recall that each evaluation of $V^{\text{low}(\beta)}$ for one β is expensive.

What kind of parametric approximation, and what choice of the parameters can be considered "good" when $V(S,t)$ is still unknown? To this end, upper bounds V^{up} can be constructed, and one attempts to push the difference $V^{\text{up}} - V^{\text{low}}$ close to zero in order to improve the approximation provided by (3.38).[10] An upper bound can be obtained, for example, when one peers into the future. As a simple example, the entire path S_t for $0 \le t \le T$ can be simulated, and the option is "exercised" in

[10]Since the bounds are approximated by stochastic methods, it might happen that the true value $V(S,0)$ is not inside the calculated interval (3.38).

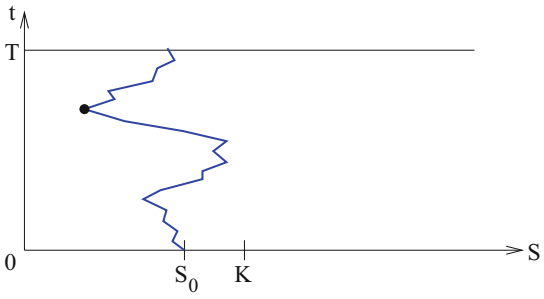

Fig. 3.12 No stopping time; maximizing the payoff of a *given* path

retrospect when

$$e^{-rt}\,\Psi(S_t)$$

is maximal. This is illustrated in Fig. 3.12. Pushing the lower bounds $V^{\mathrm{low}(\beta)}$ towards upper bounds amounts to search in the β-parameter space for a better combination of β-values. As a by-product of approximating $V(S,0)$, the corresponding parameters β provide an approximation of the early-exercise curve.

The above is just a crude strategy how Monte Carlo can be applied to approximate American options. In particular, the described simple approach to obtain upper bounds is not satisfactory. Consult [9] for a systematic way of constructing reasonable upper bounds. Typically, the upper bounds are more costly than the lower ones. Bounds are also provided by the stochastic grids of [59].

3.6.3 Regression Methods

One basic idea of regression methods is to approximate the American-style option by a Bermudan option. A Bermudan option restricts early exercise to specified discrete dates during its life. As in Sect. 1.8.4, the time instances with the right to exercise are created artificially by a finite set of discrete time instances t_j :

$$\Delta t := \frac{T}{M}\,, \quad t_j := j\,\Delta t \;\; (j = 0, \ldots, M)\,,$$

see the illustration of Fig. 3.13. The situation resembles the time discretization of the binomial method of Sect. 1.4. In that semidiscretized setting the value of the dynamic programming procedure of Eq. (1.22) generalizes to

$$V_j(S) = \max\{\,\Psi(S)\,,\; V_j^{\mathrm{cont}}(S)\,\}\,,$$

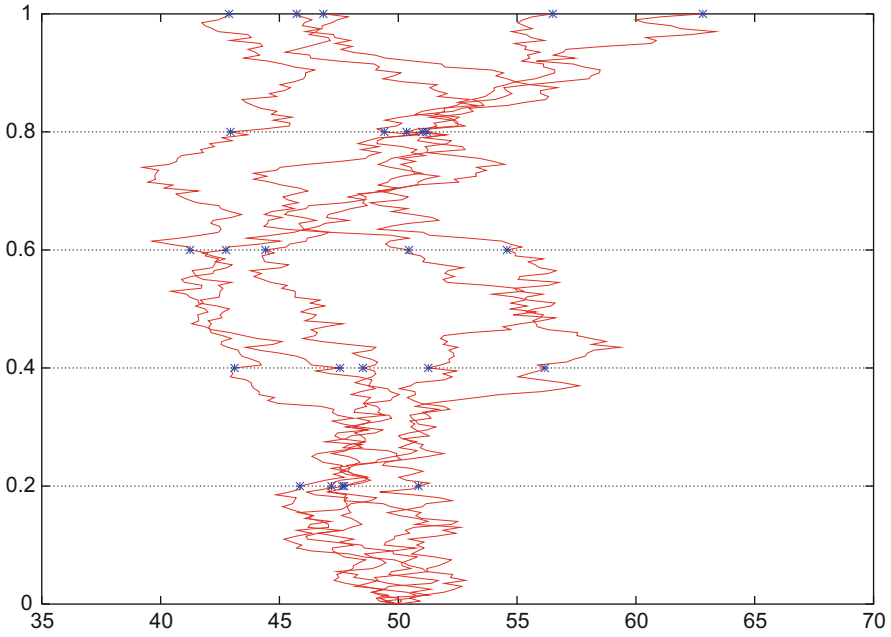

Fig. 3.13 Setting for a Bermudan option with one underlying; schematic illustration with five trajectories $S_1(t), \ldots, S_5(t)$ and $M = 5$ exercise times t_j with $S_{j,k} := S_k(t_j)$; data as in Fig. 3.1; horizontal axis: S, vertical axis: t. The points $(S_{j,k}, t_j)$ are marked

where the continuation value or *holding value* V_j^{cont} is defined by the conditional expectation

$$V_j^{\text{cont}}(S) := \mathrm{e}^{-r\Delta t}\, \mathsf{E}_{\mathsf{Q}}\left(V_{j+1}(S_{j+1}) \mid S_j = S \right).$$

[For the binomial tree, this is Eq. (1.21).] E_{Q} is calculated as before under the assumption of risk neutrality.

In the context of a Bermudan option, we define the continuation value

$$C_j(x) := \mathrm{e}^{-r\Delta t}\, \mathsf{E}_{\mathsf{Q}}\left(V(S_{t_{j+1}}, t_{j+1}) \mid S_{t_j} = x \right). \tag{3.40}$$

Here x is a vector with as many components as the number of underlyings. The function $C_j(x)$ needs to be approximated. If we can do it, then the general recursion is:

Algorithm 3.13 (Dynamic Programming)

Set $V_M(x) \equiv \Psi(x)$.
For $j = M-1, \ldots, 1$ *construct* $C_j(x)$ *for* $x > 0$ *and evaluate*
$V_j(x) := V(x, t_j) = \max\left\{\Psi(x),\ C_j(x)\right\}$ *for grid points* x.
$V_0 := V(S_0, 0) = \max\left\{\Psi(S_0),\ C_0(S_0)\right\}$

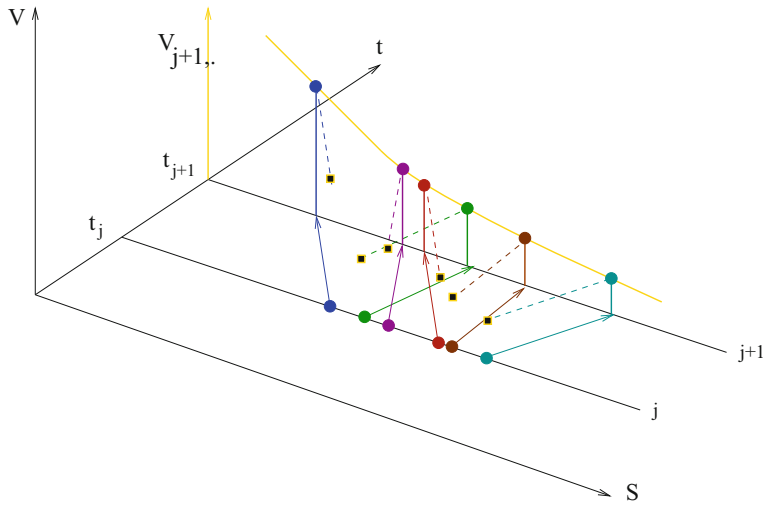

Fig. 3.14 Fictive illustration of the regression for $N = 6$, scalar case ($x \in \mathbb{R}$). The *circles* indicate the mapping $S_{j,k} \to V_{j+1,k}$ for $k = 1, \ldots, N$. Indicated by *dashed lines* and *little squares*, the discounting and the tuples $(S_{j,k}, \mathrm{e}^{-r\Delta t} V_{j+1,k})$ are indicated, which are the data for the computation of the function $\hat{C}_j(x)$ by regression

To calculate an approximation $\hat{C}_j(x)$ for $C_j(x)$, data are generated by running N simulations. All simulating paths are calculated starting from S_0, according to the underlying risk-neutral model. In this way, paths $S_1(t), \ldots, S_N(t)$ are created for $0 \le t \le T$ ($N = 5$ in Fig. 3.13). At the discrete t_j values, this establishes stochastic grid points $S_{j,k} := S_k(t_j)$ and assigns $(S_{j,k}, t_j)$ to $(S_{j+1,k}, t_{j+1})$ for $k = 1, \ldots, N$ and all i. Dropping the index k, this amounts to the transition $S_j \longrightarrow S_{j+1}$. On S_{j+1} a valuation V_{j+1} is calculated by the recursion. Hence N tuples $(S_j, \mathrm{e}^{-r\Delta t} V_{j+1})$ are provided for each j. These tuples match (3.40) and form the data basis on which $(x, C(x))$ is approximated by a suitable minimization method such as least squares, see also Fig. 3.14.[11] This sets up the basic principle of regression methods.

Algorithm 3.14 (Regression I)

(a) Simulate N paths $S_1(t), \ldots, S_N(t)$. Calculate and store the values

$$S_{j,k} := S_k(t_j)\,, \quad j = 1, \ldots, M,\ k = 1, \ldots, N\,.$$

(b) For $j = M$ set $V_{M,k} := \Psi(S_{M,k})$ for all k.

[11] For least squares see Appendix C.4.

(c) For $j = M-1, \ldots, 1$:

Approximate $C_j(x)$ using suitable basis functions $\phi_0, \ldots, \phi_L$ (monomials, for example)

$$C_j(x) \approx \sum_{l=0}^{L} a_l \phi_l(x) =: \hat{C}_j(x)$$

by least squares over the N points

$$(x_k,\, y_k) := (S_{j,k},\, \mathrm{e}^{-r\Delta t} V_{j+1,k})\,, \quad k = 1, \ldots, N,$$

and set

$$V_{j,k} := \max\left\{\Psi(S_{j,k}),\, \hat{C}_j(S_{j,k})\right\}\,.$$

(d) $\hat{C}_0 := \mathrm{e}^{-r\Delta t}\frac{1}{N}(V_{1,1} + \ldots + V_{1,N})\,, \quad V_0 = \max\left\{\Psi(S_0),\, \hat{C}_0\right\}\,.$

Again, step (a) is illustrated by Fig. 3.13, and the regression step (c) is explained by Fig. 3.14. The coefficients $a_0, \ldots, a_L$ of the approximation $\hat{C}$ result from a minimization. Step (d) is needed because (c) does not work for $j = 0$ since all $S_{0,k} = S_0$. Since in the multifactor case the S and the x are vectors, the minimization in the algorithm can become costly. Note that for convergence both N and L must be increased.

The above basic version of regression can be improved in several ways. Longstaff and Schwartz [247] has introduced a special version of the regression, incorporating as a subalgorithm the calculation of the stopping time of each path. Working with individual stopping times enables to set up an interleaving mechanism over the time levels for comparing cash flows. The central step in (c) changes to

$$V_{j,k} := \begin{cases} \Psi(S_{j,k}) & \text{for } \Psi(S_{j,k}) \geq \hat{C}_j(S_{j,k})\,, \\ V_{j+1,k} & \text{for } \Psi(S_{j,k}) < \hat{C}_j(S_{j,k})\,. \end{cases} \tag{3.41}$$

This requires to adapt steps (b), (c), (d). Points out-of-the-money do not enter the regression. To save storage, intermediate values can be filled in by using a bridging technique. Following [205], a significant speed-up is possible when working with a cash-flow vector g, and an integer stopping time vector τ (the integer factors k of $\tau_k = k\Delta t$). The resulting algorithm is:

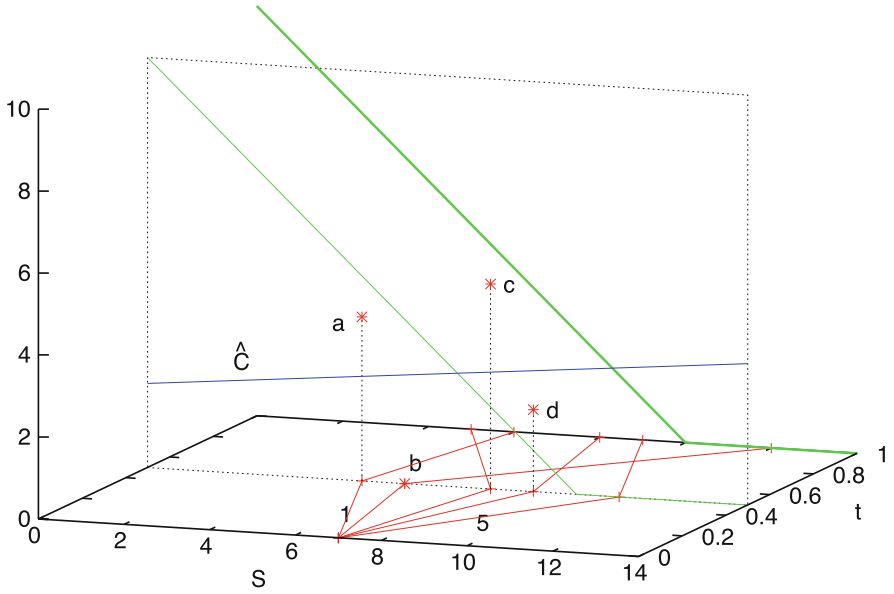

Fig. 3.15 Regression; illustration for a put with $r = 0$, $M = 2$, $K = 10$

Algorithm 3.15 (Regression II)

(a) Simulate N paths as in Algorithm 3.14.
(b) Set $g_k := \Psi(S_{M,k})$, $\tau_k = M$ *for* $k = 1, \ldots, N$.
(c) For $j = M-1, \ldots, 1$:
For the subset of in-the-money-points

$$(x_k,\, y_k) := (S_{j,k},\, \mathrm{e}^{-r(\tau_k - j)\Delta t} g_k)\,,$$

approximate $C_j(x)$ *by* $\hat{C}_j(x)$,
and for those k with $\Psi(S_{j,k}) \geq \hat{C}_j(S_{j,k})$: *update*

$$g_k := \Psi(S_{j,k}),\ \tau_k := j\,.$$

(d) $\hat{C}_0 := \dfrac{1}{N} \sum_{k=1}^{N} \mathrm{e}^{-r\tau_k \Delta t} g_k\,, \quad V_0 := \max\{\Psi(S_0),\, \hat{C}_0\}\,.$

Figure 3.15 shows a simple artificial setting as an attempt to illustrate the regression method, with strike $K = 10$, and $M = 2$, $N = 5$. For $j = 1$, four of the paths are in the money. Their continuation values $V_{j+1,k}$ are denoted a, b, c, d in Fig. 3.15. The heavy line is the regression $\hat{C}$, here a straight line because it is based only on the two regressors $\phi_0 = 1$, $\phi_1 = x$. The maximum $\max\{\Psi, \hat{C}\}$ is easy to check: for the points a and b the payoff is larger than $\hat{C}(S)$.

Recently, many refined Monte Carlo methods for the calculation of American options have been suggested. For an overview on related approaches, consult Chap. 8 in [155]. At current state, the robust regression of [206] appears to be a most efficient approach; it has priced options on baskets of up to 30 underlying assets. One basic ingredient of this method is to neglect outliers, with the effect of a remarkable reduction of the bias.

3.7 Sensitivity

A great computational challenge is to estimate how the price V changes when parameters or initial states change, see Sect. 1.4.6. A sensitivity analysis based on approximating partial derivatives amounts to calculating greeks, and can be used for calibration. Recall that for tree methods and for finite-difference methods there are easy ways to establish approximations to the greeks delta, gamma, and theta, without the need for any recalculations. For Monte Carlo methods, this task is more costly. When results are required for slightly changed parameter values, to set up difference quotients, it may be necessary to rerun Monte Carlo.

As an example we comment on approximating delta$= \frac{\partial V}{\partial S}$. A simple approach is to apply two runs of Monte Carlo simulation, one for S_0 and one for a close value $S_0 - \Delta S$. Then an approximation of delta is obtained by the difference quotient

$$\frac{V(S_0) - V(S_0 - \Delta S)}{\Delta S}\,. \tag{3.42}$$

The increment ΔS must be chosen carefully and not too small, because (B.6) in Appendix B.1 tells us that the variance of (3.42) for arbitrary numerator scales with $(\Delta S)^{-2}$. So it is important to investigate how the numerator depends on ΔS. Simulating the two terms $V(S_0)$ and $V(S_0 - \Delta S)$ using common random numbers improves the situation. Computing time can be saved by working with series of precalculated random numbers. The crude approach symbolized by (3.42) does not require additional programming, but the costs are prohibitive for multiasset options.

With some more sophistication, the effort can be reduced. For example, options are often priced for different maturities. When Monte Carlo is combined with a bridging technique, several such options can be priced effectively in a single run [316]. A general reference on estimating sensitivities is Chap. 7 in [155].

There are alternatives improving accuracy and saving computing time. For example, Malliavin calculus allows to shift the differencing to the density function, which leads via a kind of integration by parts to a different integral to be approximated by Monte Carlo. For references on this technique consult [137].

Another method that speeds up a sensitivity analysis significantly is the adjoint method developed by [152], which is described next.

3.7.1 Pathwise Sensitivities

Sensitivities can be approximated in a pathwise fashion. Similar as in (1.56) consider a system of autonomous SDEs

$$\mathrm{d}X_t = a(X_t)\,\mathrm{d}t + b(X_t)\,\mathrm{d}W_t \tag{3.43}$$

where $X_t \in \mathbb{R}^n$, and $W_t \in \mathbb{R}^m$ is a vector of independent Wiener processes. That is, $a(X)$ is an n-vector, and $b(X)$ is $(n \times m)$-matrix with elements b_{iv} and takes care of possible correlations ($\longrightarrow$ Exercise 3.17). For a standard discretization with M steps assume again $t_0 = 0$, $T = \Delta t \cdot M$, $t_j := j\Delta t$, $j = 0, \ldots, M$, and let $\Psi(X(T))$ denote the *discounted* payoff. The Euler discretization of (3.43) is

$$y(t_{j+1}) = y(t_j) + a(y(t_j))\Delta t + b(y(t_j))Z(t_j)\sqrt{\Delta t}\,. \tag{3.44}$$

We consider one calculated path X_t, $0 \le t \le T$, represented by $y(t_j)$, $0 \le j \le M$, and keep its random vectors $Z(t_j)$ available. The aim is to estimate the sensitivity vector

$$s(0)^{tr} := \frac{\partial \Psi(X(T))}{\partial X(0)}$$

(taken as a row vector). By the chain rule,

$$s(0)^{tr} = \frac{\partial \Psi(X(T))}{\partial X(T)}\,\frac{\partial X(T)}{\partial X(0)}\,. \tag{3.45}$$

The first factor is easily available. The endeavor is to approximate the matrix $\frac{\partial X(T)}{\partial X(0)}$. To this end, we use the dynamics as created by the Euler method (3.44), and calculate the approximation

$$\frac{\partial y(T)}{\partial y(0)}\,.$$

As outlined in [155, Sect. 7.2], we differentiate the ith component of the Euler formula (3.44) with respect to $y_k(t_0)$, which gives

$$\begin{aligned}\frac{\partial y_i(t_{j+1})}{\partial y_k(t_0)} = \frac{\partial y_i(t_j)}{\partial y_k(t_0)} &+ \sum_{l=1}^{n} \frac{\partial a_i(y(t_j))}{\partial y_l(t_j)}\,\frac{\partial y_l(t_j)}{\partial y_k(t_0)}\Delta t \\ &+ \frac{\partial}{\partial y_k(t_0)} \sum_{v=1}^{m} b_{iv}(y(t_j))Z_v(t_j)\sqrt{\Delta t}\end{aligned}$$

for all $i, k = 1, \ldots, n$. The last term is

$$\sum_{\nu=1}^{m} \sum_{l=1}^{n} \frac{\partial b_{i\nu}(y(t_j))}{\partial y_l(t_j)} \frac{\partial y_l(t_j)}{\partial y_k(t_0)} Z_\nu(t_j) \sqrt{\Delta t}\,.$$

With

$$\Delta_{ik}(j) := \frac{\partial y_i(t_j)}{\partial y_k(t_0)}$$

this is written

$$\begin{aligned} \Delta_{ik}(j+1) = \Delta_{ik}(j) &+ \sum_{l=1}^{n} \frac{\partial a_i(y(t_j))}{\partial y_l(t_j)} \Delta_{lk}(j) \Delta t \\ &+ \sum_{\nu=1}^{m} \sum_{l=1}^{n} \frac{\partial b_{i\nu}(y(t_j))}{\partial y_l(t_j)} \Delta_{lk}(j) Z_\nu(t_j) \sqrt{\Delta t}\,. \end{aligned} \tag{3.46}$$

The recursion (3.46) can be written in matrix notation. To this end, we use the definition (as in [152]) of the entries of $(n \times n)$-matrices $D(j)$

$$D_{ik}(j) := \delta_{ik} + \frac{\partial a_i(y(t_j))}{\partial y_k(t_j)} \Delta t + \sum_{\nu=1}^{m} \frac{\partial b_{i\nu}(y(t_j))}{\partial y_k(t_j)} Z_\nu(t_j) \sqrt{\Delta t}\,. \tag{3.47}$$

(Here $\delta_{ik} = 1$ for $k = i$, and $= 0$ for $k \neq i$, is the Kronecker symbol and no dividend yield.) The resulting recursion for the $(n \times n)$-matrices $\Delta(j)$ with elements $\Delta_{ik}(j)$ is

$$\Delta(j+1) = D(j)\Delta(j), \quad j = 0, \ldots, M-1, \quad \Delta(0) = I\,. \tag{3.48}$$

This summarizes the evolution of the path in a *forward fashion*. After M matrix products the final matrix $\Delta(M)$ is the estimate $\frac{\partial y(T)}{\partial y(0)}$ for $\frac{\partial X(T)}{\partial X(0)}$. Then an approximation $\bar{s}(0)^{tr}$ of the sensitivity vector $s(0)^{tr}$ is obtained via the product (3.45).

3.7.2 Adjoint Method

A backward view is possible too. To see this, rewrite the above as

$$\begin{aligned} \bar{s}(0)^{tr} :&= \frac{\partial \Psi(y(T))}{\partial y(T)} \frac{\partial y(T)}{\partial y(0)} = \frac{\partial \Psi(y(T))}{\partial y(T)} \Delta(M) \\ &= \frac{\partial \Psi(y(T))}{\partial y(T)} D(M-1) \cdot \cdots \cdot D(0) \end{aligned}$$

The observation of [152] is that $\bar{s}(0)$ can be calculated with a backward recursion, which operates n-vectors rather than $(n \times n)$-matrices. We start with the row vector

$$\bar{s}(M)^{tr} := \frac{\partial \Psi(y(T))}{\partial y(T)}$$

and obtain

$$(\bar{s}(M-1))^{tr} = \frac{\partial \Psi}{\partial y(T)} D(M-1)\,,$$

or

$$\bar{s}(M-1) = (D(M-1))^{tr}\,\bar{s}(M)\,.$$

The next row vector is

$$(\bar{s}(M-2))^{tr} = \frac{\partial \Psi}{\partial y(T)} D(M-1)D(M-2) = (\bar{s}(M-1))^{tr} D(M-2),$$

or

$$\bar{s}(M-2) = (D(M-2))^{tr}\,\bar{s}(M-1)\,,$$

and so on, which results in the recursion

$$\bar{s}(j) = (D(j))^{tr}\bar{s}(j+1), \quad j = M-1,\ldots,0, \quad \bar{s}(M) = \left(\frac{\partial \Psi}{\partial y(T)}\right)^{tr}. \tag{3.49}$$

This backward recursion updates the n components of the vector s for every j, whereas the forward recursion (3.48) updates the n^2 entries of Δ in each step. Hence the forward recursion (3.48) involves a factor of n more arithmetic operations than the backward recursion. Consequently, the backward recursion should be significantly faster for $n > 1$. But there is one drawback of the potentially fast backward recursion: Its implementation requires to store the entire path of the y-vectors with their Z-vectors in order to have the D-matrices available. For very small step sizes Δt (M large) this deteriorates the speed somewhat. And switching to another payoff Ψ requires to recalculate the backward recursion, whereas the forward recursion can use the previous $\Delta(M)$ again. Observing these two features, the backward recursion (3.49) ("adjoint method") can be highly advantageous. The above method approximates pathwise deltas. In a similar way, sensitivities with respect to parameters can be calculated, see [152].

Example 3.16 (Heston-Hull-White Model) Extending Heston's model (1.59) by an SDE for the interest rate r_t leads to the system

$$\begin{aligned} \mathrm{d}S_t &= r_t S_t \,\mathrm{d}t + \sqrt{v_t}\, S_t \,\mathrm{d}\tilde{W}_t^{(1)} \\ \mathrm{d}v_t &= \kappa(\theta - v_t)\,\mathrm{d}t + \sigma_2 \sqrt{v_t}\,\mathrm{d}\tilde{W}_t^{(2)} \\ \mathrm{d}r_t &= \alpha(R(t) - r_t)\,\mathrm{d}t + \sigma_3\,\mathrm{d}\tilde{W}_t^{(3)} \end{aligned} \tag{3.50}$$

The function R in the mean-reversion term for r_t can be chosen as to match the current term structure [163], here chosen as constant for simplicity:

$$R \equiv 0.06,\ \alpha = 0.1,\ \kappa = 3,\ \theta = 0.12,$$
$$\sigma_2 = 0.04,\ \sigma_3 = 0.01,\ T = 1,\ K = 100.$$

The mean reversion level $\theta = 0.12$ corresponds to a volatility of about 35%. The Brownian motions $\tilde{W}_t^{(1)}$, $\tilde{W}_t^{(2)}$, $\tilde{W}_t^{(3)}$ are assumed (partly) correlated:

$$\rho_{12} = 0.6,\ \rho_{13} = \rho_{23} = 0\,,$$

hence $\tilde{W}_t^{(3)}$ is not correlated with $\tilde{W}_t^{(1)}$, $\tilde{W}_t^{(2)}$. Accordingly, the Cholesky decomposition (Sect. 2.3.4) has a block structure, and Exercise 2.17 can be applied. To cast it into the framework of (1.56)/(1.57), observe $n = 3$,

$$X := \begin{pmatrix} S \\ v \\ r \end{pmatrix}, \quad a(X) = \begin{pmatrix} X_1 X_3 \\ \kappa(\theta - X_2) \\ \alpha(R - X_3) \end{pmatrix}$$

and

$$b(X)\,\mathrm{d}W_t = \begin{pmatrix} X_1\sqrt{X_2} & 0 & 0 \\ \sigma_2\sqrt{X_2}\,\rho_{12} & \sigma_2\sqrt{X_2}\sqrt{1-\rho_{12}^2} & 0 \\ 0 & 0 & \sigma_3 \end{pmatrix} \begin{pmatrix} \mathrm{d}W_t^{(1)} \\ \mathrm{d}W_t^{(2)} \\ \mathrm{d}W_t^{(3)} \end{pmatrix}$$

with independent Wiener processes $W^{(i)}$. In the discretization the Wiener process can be taken as

$$\sqrt{\Delta t}\, Z_1(t),\ \sqrt{\Delta t}\, Z_2(t),\ \sqrt{\Delta t}\, Z_3(t)$$

with $Z_i \sim \mathcal{N}(0,1)$. $\sqrt{\Delta t}\, b(X) Z$ is a vector, an its partial derivatives enter (3.47).

For a concrete example, we price a European call. Since the interest rate is variable, we discount each trajectory with its proper rate. Hence, the discounted payoff is

$$\exp\left(-\int_0^T r_t\,\mathrm{d}t\right)(S_T - K)^+\,.$$

We choose the starting point

$$S_0 = 95,\ v_0 = \theta,\ r_0 = R\,,$$

and approximate the discounting integral by the trapezoidal sum (C.2). Experiments resulted in $V(S_0, v_0, r_0, 0) \approx 13.1$. The reader is encouraged to set up the matrix $D(j)$ and test the adjoint method.

3.8 Notes and Comments

On Sects. 3.1 and 3.2

Under suitable assumptions it is possible to prove existence and uniqueness for strong solutions, see [225]. Usually the discretization error dominates other sources of error. We have neglected the sampling error (the difference between $\widehat{\epsilon}$ and ϵ), imperfections in the random number generator, and rounding errors. Typically these errors are likely to be less significant. Sect. 3.2 follows Sect. 5.1 of [225].

On Sect. 3.3

[225] discusses many methods for the approximation of paths of SDEs, and proves their convergence. An introduction is given in [301]. Possible orders of strongly converging schemes are integer multiples of $\frac{1}{2}$ whereas the orders of weakly converging methods are whole numbers. A weak scheme of order β needs all the multiple integrals of up and including β. Simple adaptions of deterministic schemes do not converge for SDEs. For the integration of *random ODEs* we refer to [160]. Maple routines for SDEs can be found in [93], and MATLAB routines in [180].

Linear stability is concerned with the long-time behavior of solutions of the test equation $\mathrm{d}X_t = \alpha X_t\,\mathrm{d}t + \beta X_t\,\mathrm{d}W_t$, where α is a complex number with negative real part. This situation does not appear relevant for applications in finance. The numerical stability in the case of a negative real part of α depends on the step size h and the relation among the three parameters α, β, h. For this topic and further references we refer to [180, 301, 327].

On Sect. 3.4

For Brownian bridges see, for instance, [155, 216, 225, 279, 291, 314, 340]. Other bridges than Brownian bridges are possible. For a Gamma process and a Gaussian

bridge this is shown in [315, 316]. For the effectiveness of Monte Carlo integration improved with bridging techniques, see [64]. The probability that a Brownian bridge passes a given barrier is found in [216], see also [155]. The maximum of a Wiener process tied down to $W_0 = 0$, $W_1 = a$ on $0 \le t \le 1$ has the distribution $F(x)$ of Exercise 2.8. And the time instant at which the maximum is attained is distributed with the distribution function

$$F(x) = \frac{2}{\pi} \arcsin(\sqrt{x}) \quad \text{for } 0 \le x \le 1 \,.$$

Another alternative to fill large gaps is to apply fractal interpolation [256].

On Sect. 3.5

Monte Carlo simulation is of great importance for general models where no specific assumptions (as those of Black, Merton and Scholes) have led to efficient approaches. For example, in case the interest rate r cannot be regarded as constant but is modeled by some SDE [such as Eq. (1.55)], then a system of SDEs must be integrated. Examples of stochastic volatility are provided by Example 1.15, or by the Heston model (1.59). In such cases, a Monte Carlo simulation can be the method of choice. Then the Algorithm 3.6 is adapted appropriately. Monte Carlo methods are especially attractive for multifactor models with high dimension.

In the literature the basic idea of the approach summarized by Eq. (3.24) is analyzed using martingale theory, compare the references in Chap. 1 and Appendix B.2. An early paper suggesting MC for the pricing of options is [46]. The calculation of risk indices such as *value at risk* is an important application of Monte Carlo methods, see the notes on Sect. 1.8. The equivalence of the Monte Carlo simulation [representation (3.23)/(3.24)] with the solution of the Black–Scholes equation is guaranteed by the theorem of Feynman and Kac [40, 216, 283, 291, 311, 340, 353]. Much effort is involved in estimating the bias [373]. Joshi [207] discusses how to apply MC to path-dependent options and greeks. For professional application, our brief introduction will not suffice. Consult, in particular, the standard reference on MC in finance [155].

Monte Carlo simulations can be parallelized in a trivial way: The single simulations can be distributed among the processors in a straightforward fashion because they are independent of each other. If m processors are available, the speed reduces by a factor of $1/m$. But the streams of random numbers in each processor must be independent. For related generators see [262]. In doubtful and sensitive cases Monte Carlo simulation should be repeated with other random-number generators, and with low-discrepancy numbers [200].

For a discussion of variance reduction and examples, consult Chap. 4 in [155]. For the variance-reduction method of *importance sampling*, see also [284]. Different variance-reduction techniques can be combined with each other. In particular, a

change of drift helps driving the underlying assets into "important" regions. An optimal drift is possible that reduces the variance significantly. Arouna [13] suggests a truncated version of the Robbins-Monro algorithm, and [206] reduces the number of insignificant paths for his robust regression with a deterministic method.

The demands for **accuracy** of Monte Carlo simulation should be kept on a modest level. In many cases an error of 0.1% (or even 1%) must suffice. Recall that it does not make sense to decrease the Monte Carlo sampling error significantly below the error of the time discretization of the underlying SDE (and vice versa). When the amount of available random numbers is too small or its quality poor, then no improvement of the error can be expected. The methods of variance reduction can save a significant amount of costs [47, 301, 331]. The efficiency of Monte Carlo simulations can be enhanced by suitably combining several discretizations with different levels of coarseness [151].

On Sect. 3.6

For Monte Carlo simulation on American options see also [47, 58, 143, 155, 234, 247, 319]. Note that for multivariate options of the American style the costs are increasing with the dimension of x more significantly than for European options. For parametric methods, the parameter vector β defines surfaces rather than curves. And for regression methods, the calculation of C or $\hat{C}$ is costly and does depend on the number of underlyings (dimension of x). A nice experiment with a parametric method is given in [181]. Significant savings are possible when the dimension is reduced by a principal component analysis ($\longrightarrow$ Exercise 2.16).

A first version of regression was introduced by [360], where the continuation value was approximated based on subsets of paths. This bundling technique was modified in [70] by an improved regression. As [360] points out, a single set of paths of an underlying asset can be generated and then used repeatedly to value many different derivatives. Lack of independence makes it difficult to prove convergence, or to set up confidence intervals. For these aspects, see [115], and [9] and the references therein.

3.9 Exercises

3.1 (Implementing Euler's Method)
Implement Algorithm 1.11. Start with a test version for one scalar SDE, then develop a version for a system of SDEs. Test examples:

(a) Perform the experiment of Fig. 1.18.
(b) Integrate the system of Example 1.15 for $\alpha = 0.3$, $\beta = 10$ and the initial values $S_0 = 50$, $\sigma_0 = 0.2$, $\xi_0 = 0.2$ for $0 \le t \le 1$.

Plot the calculated trajectories.

3.2 (Binary Random Variate) Let α, β, p with $0 < p < 1$ be given numbers. Design an algorithm that outputs α with probability p and β with probability $1-p$.

3.3 (Itô Integral in Eq. (3.11)) Let the interval $0 \le s \le t$ be partitioned into n subintervals, $0 = t_0 < t_1 < \ldots < t_n = t$. For a Wiener process W_t assume $W_{t_0} = 0$.

(a) Show $\displaystyle\sum_{j=0}^{n-1} W_{t_j}\left(W_{t_{j+1}} - W_{t_j}\right) = \frac{1}{2}W_t^2 - \frac{1}{2}\sum_{j=0}^{n-1}\left(W_{t_{j+1}} - W_{t_j}\right)^2$

(b) Use Lemma 1.17 to deduce Eq. (3.11).

3.4 (Integration by Parts for Itô Integrals)

(a) Show

$$\int_{t_0}^{t} s\,\mathrm{d}W_s = tW_t - t_0 W_{t_0} - \int_{t_0}^{t} W_s\,\mathrm{d}s$$

Hint: Start with the Wiener process $X_t = W_t$ and apply the Itô Lemma with the transformation $y = g(x,t) := tx$.

(b) Denote $\Delta Y := \int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}W_z\,\mathrm{d}s$ and $\Delta t := t - t_0$. Show by using (a) that

$$\int_{t_0}^{t}\int_{t_0}^{s} \mathrm{d}z\,\mathrm{d}W_s = \Delta W \Delta t - \Delta Y\,.$$

3.5 (Error of the Milstein Scheme)
To which formula does the Milstein scheme reduce for linear SDEs? Perform the experiment outlined in Example 3.2 using the Milstein scheme of Algorithm 3.5. Set up a table similar as in Table 3.1 to show

$$\widehat{\varepsilon}(h) \approx h$$

for Example 3.2.

3.6 (Transforming the CIR Equation)
For the CIR equation

$$\mathrm{d}v_t = \kappa(\theta - v_t)\,\mathrm{d}t + \sigma^{\mathrm{v}}\sqrt{v_t}\,\mathrm{d}W_t$$

with constant $\kappa, \theta, \sigma^{\mathrm{v}}$ find a transformation g such that the coefficient $\tilde{b}$ in the SDE of $y_t := g(v_t)$,

$$\mathrm{d}y_t = \tilde{a}(y_t)\,\mathrm{d}t + \tilde{b}(y_t)\,\mathrm{d}W_t\,,$$

is a constant.

3.7 (Drift-Implicit Scheme for CIR)
Show for the CIR-model in (3.13) that for $Y_t := \sqrt{X_t}$ the SDE

$$\mathrm{d}Y = \frac{4\kappa\theta - \sigma^2}{8}\,\frac{1}{Y}\,\mathrm{d}t - \frac{\kappa}{2}\,Y\,\mathrm{d}t + \frac{\sigma}{2}\,\mathrm{d}W$$

results. For this SDE analyze one step of the drift-implicit scheme

$$y_{j+1} = y_j + a(y_{j+1})\Delta t + b(y_j)\,\Delta W_j$$

and derive a quadratic equation for y_{j+1}. For which values of the parameters κ, σ, θ does the quadratic equation have a positive solution y_{j+1}?

3.8 (Moments of Itô Integrals for Weak Solutions)

(a) Use the Itô isometry

$$\mathsf{E}\left[\left(\int_a^b f(t,\omega)\,\mathrm{d}W_t\right)^2\right] = \int_a^b \mathsf{E}\left[f^2(t,\omega)\right]\,\mathrm{d}t$$

to show its generalization

$$\mathsf{E}\left[I(f)I(g)\right] = \int_a^b \mathsf{E}[fg]\,\mathrm{d}t\,, \qquad \text{where} \quad I(f) = \int_a^b f(t,\omega)\,\mathrm{d}W_t\,.$$

Hint: $4fg = (f+g)^2 - (f-g)^2$.

(b) For $\Delta Y := \int_{t_0}^t \int_{t_0}^s \mathrm{d}W_z\,\mathrm{d}s$ the moments are

$$\mathsf{E}[\Delta Y] = 0, \quad \mathsf{E}[\Delta Y^2] = \frac{\Delta t^3}{3}, \quad \mathsf{E}[\Delta Y \Delta W] = \frac{\Delta t^2}{2} \quad \text{and} \quad \mathsf{E}[\Delta Y \Delta W^2] = 0.$$

Show this by using (a) and $\mathsf{E}\left[\int_a^b f(t,\omega)\,\mathrm{d}W_t\right] = 0$.

3.9 (Simulation of $I_{(1,0)}$)
By transformation of two independent standard normally distributed random variables $Z_i \sim \mathcal{N}(0,1)$, $i = 1, 2$, two new random variables are obtained by

$$\widehat{\Delta W} := Z_1\sqrt{\Delta t}, \qquad \widehat{\Delta Y} := \frac{1}{2}(\Delta t)^{3/2}\left(Z_1 + \frac{1}{\sqrt{3}}Z_2\right).$$

Show that $\widehat{\Delta W}$ and $\widehat{\Delta Y}$ have the moments of (3.17).

3.10 (Auxiliary Variables)
In addition to (3.17) further moments are

$$\mathsf{E}(\Delta W) = \mathsf{E}(\Delta W^3) = \mathsf{E}(\Delta W^5) = 0, \quad \mathsf{E}(\Delta W^2) = \Delta t, \quad \mathsf{E}(\Delta W^4) = 3\Delta t^2.$$

Assume a new random variable $\widetilde{\Delta W}$ satisfying

$$\mathsf{P}\left(\widetilde{\Delta W} = \pm\sqrt{3\Delta t}\right) = \frac{1}{6}, \qquad \mathsf{P}\left(\widetilde{\Delta W} = 0\right) = \frac{2}{3}$$

and the additional random variable

$$\widetilde{\Delta Y} := \frac{1}{2}\widetilde{\Delta W}\Delta t\,.$$

Show that the random variables $\widetilde{\Delta W}$ and $\widetilde{\Delta Y}$ have up to terms of order $O(\Delta t^3)$ the same moments as ΔW and ΔY.

3.11 (Brownian Bridge)
For a Wiener process W_t consider

$$X_t := W_t - \frac{t}{T}W_T \quad \text{for } 0 \le t \le T\,.$$

Calculate $\mathsf{Var}(X_t)$ and show that

$$\sqrt{t\left(1-\frac{t}{T}\right)}\,Z \quad \text{with } Z \sim \mathcal{N}(0,1)$$

is a realization of X_t.

3.12 (Black–Scholes Formula)
For the value $V(S_t, t)$ of a European put with time to maturity $\tau := T - t$ prove that

$$\begin{aligned}&\mathrm{e}^{-r\tau}\int_0^\infty (K-S_T)^+ \frac{1}{S_T\sigma\sqrt{2\pi\tau}}\exp\left\{-\frac{[\ln(S_T/S_t)-(r-\frac{\sigma^2}{2})\tau]^2}{2\sigma^2\tau}\right\}\,\mathrm{d}S_T\\ &= \mathrm{e}^{-r\tau}KF(-d_2) - S_tF(-d_1)\,,\end{aligned}$$

where d_1 and d_2 are defined in (A.13)/(A.14).
Hints: The left-hand side collects the terms of (3.23). Use $(K-S_T)^+ = 0$ for $S_T > K$, and get two integrals.

3.13 (Monte Carlo for European Options)
Implement a Monte Carlo method for single-asset European options, based on the Black–Scholes model. Perform experiments with various values of N and a random number generator of your choice. Compare results obtained by using the analytic solution formula for S_t with results obtained by using Euler's discretization. For (c) B is the barrier such that the option expires worthless when $S_t \ge B$ for some t.
Input: S_0, number of simulations (trajectories) N, payoff function $\Psi(S)$, risk-neutral interest rate r, volatility σ, time to maturity T, strike K.

Payoffs:

(a) vanilla put, with $\Psi(S) = (K - S)^+$, $S_0 = 5, K = 10, r = 0.06, \sigma = 0.3$, $T = 1$.
(b) binary call, with $\Psi(S) = \mathbf{1}_{S>K}$, $S_0 = K = \sigma = T = 0.5, r = 0.1$
(c) up-and-out barrier: call with $S_0 = 5, K = 6, r = 0.05, \sigma = 0.3, T = 1, B = 8$.

Hint: Correct values are: (a) 4.43046 (b) 0.46220 [309] (c) 0.0983 [181]

3.14 (Error of Biased Monte Carlo)
Assume

$$\text{MSE} = \zeta(h, N) := \alpha_1^2 h^{2\beta} + \frac{\alpha_2}{N}$$

as error model of a Monte Carlo simulation with sample size N, based on a discretization of an SDE with stepsize h, where α_1, α_2 are two constants.

(a) Argue why for some constant α_3

$$C(h, N) := \alpha_3 \frac{N}{h}$$

is a reasonable model for the costs of the MC simulation.
(b) Minimize $\zeta(h, N)$ with respect to h, N subject to the side condition

$$\alpha_3 N/h = C$$

for given budget C.
(c) Show that for the optimal h, N

$$\sqrt{\text{MSE}} = \alpha_4 C^{-\frac{\beta}{1+2\beta}} .$$

3.15 (Control Variates)
Let $\widehat{V}, \widehat{V}^*$ be two random variables, and $V^* := \mathsf{E}(\widehat{V}^*)$. For a free parameter α define the controlled variable

$$V_{\text{CV}}^{\alpha} := \widehat{V} + \alpha(V^* - \widehat{V}^*) .$$

(a) Show

$$\mathsf{Var}(V_{\text{CV}}^{\alpha}) = \mathsf{Var}(\widehat{V}) + \alpha^2 \mathsf{Var}(\widehat{V}^*) - 2\alpha \mathsf{Cov}(\widehat{V}, \widehat{V}^*) .$$

(b) Determine the parameter α_0 for which $\mathsf{Var}(V_{\text{CV}}^{\alpha})$ is minimal.
(c) The optimal ratio of the variance of the controlled variable to that of the uncontrolled variable is $q_0 := \mathsf{Var}(V_{\text{CV}}^{\alpha_0})/\mathsf{Var}\widehat{V}$. How does q_0 depend on the

correlation $\rho_{\widehat{V},\widehat{V}^*}$ between $\widehat{V}$ and $\widehat{V}^*$? What is q_0 for $\rho = 0.95$, $\rho = 0.8$ and $\rho = 0.5$?

3.16 (Project: Monte Carlo Experiment)
Construct as hitting curve a parabola with horizontal tangent at $(S,t) = (K,T)$, similar as in Fig. 3.10. The parabola is defined by the intersection of its left branch with the S-axis, $(S,t) = (\beta, 0)$. Choose an American put with $K = 10$, $T = 1$, $r = 0.06$, $\sigma = 0.3$, and $S_0 = 9$ and simulate for several values of β the GBM $\mathrm{d}S = rS\,\mathrm{d}t + \sigma S\,\mathrm{d}W$ several thousand times, and calculate the hitting time for each trajectory. Estimate a lower bound to $V(S_0, 0)$ using (3.37). Decide whether an exact calculation of the hitting point makes sense. (Run experiments comparing such a strategy to implementing the hitting time restricted to the discrete time grid.) Think about how to implement upper bounds.

3.17 (SDE in Standard Form)
Let us denote (1.56) as "standard form" of a system of SDEs, with uncorrelated Wiener processes $W_t^{(1)}, \ldots, W_t^{(m)}$. What is the vector a and the matrix b for

(a) the example of Eq. (3.35),
(b) the Heston model of Eq. (1.59).

For the Heston model, first transform the unknown v_0 to the right-hand side by scaling $\tilde{v}_t := v_t / v_0$.

Chapter 4
Standard Methods for Standard Options

Now we enter the part of the book that is devoted to the numerical solution of equations of the Black–Scholes type. In this chapter, we discuss "standard" options in the sense as introduced in Sect. 1.1 and assume the scenario characterized by the Assumptions 1.2. In case of European vanilla options the value function $V(S,t)$ solves the Black–Scholes equation (1.5). It is not really our aim to solve this partial differential equation for vanilla payoff because it possesses an analytic solution ($\longrightarrow$ Appendix A.4). Ultimately our intention is to solve more general equations and inequalities. In particular, American options will be calculated numerically. But also European options without vanilla payoff are of interest; we encounter them for Bermudan options in Sect. 1.8.4, and for Asian options in Sect. 6.3.4. The goal is not only to calculate single values $V(S_0,0)$—for this purpose tree methods can be applied—but also to approximate the curve $V(S,0)$, or even the surface defined by the value function $V(S,t)$ on the half strip $S>0, 0\le t\le T$. Based on the surface of the value function, we collect information on early exercise and on the greeks (1.25), for example, on delta hedging by observing the derivative $\frac{\partial V}{\partial S}$.

American options obey *inequalities* of the type of the Black–Scholes equation (1.5). To allow for early exercise, the Assumptions 1.2 must be weakened. As a further generalization, the payment of dividends must be taken into account; otherwise early exercise does not make sense for American calls.

The main part of this chapter outlines a PDE approach based on finite differences. We begin with unrealistically simplified boundary conditions in order to keep the explanation of the discretization schemes transparent. Later sections will discuss appropriate boundary conditions, which turn out to be tricky in the case of American options. At the end of this chapter we will be able to implement a finite-difference algorithm for standard American (and European) options. Note that this approach assumes constant coefficients of the Black-Scholes equation. If we work carefully,

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_4

the resulting finite-difference computer program will yield correct approximations. But the finite-difference approach is not necessarily the most efficient one. Hints on other methods will be given at the end of this chapter. For nonstandard options we refer to Chap. 6.

The classic finite-difference methods will be explained in some detail because they are the most elementary approaches to approximate differential equations. As a side-effect, this chapter serves as introduction to several fundamental concepts of numerical mathematics. A trained reader may like to skip Sects. 4.2 and 4.3. The aim of this chapter is to introduce concepts, as well as a characterization of the free boundary (early-exercise curve), and of linear complementarity.

In addition to the finite-difference approach, "standard methods" include analytic methods, which to a significant part are based on nonnumerical analysis. The Sect. 4.8 will give an introduction to several such methods, including interpolation, a method of lines, and a method that solves an integral equation.

The broad field of available methods for pricing standard options calls for comparisons to judge on the relative merits of different approaches. Although such an endeavor goes beyond the scope of a text book, we offer some guidelines in Sect. 4.9.

4.1 Preparations

We allow for dividends paid with a continuous yield of constant level, because numerically this is a trivial extension from the case of no dividend. In case of a discrete dividend with, for example, one payment per year, a first remedy would be to convert the dividend to a continuous yield ($\longrightarrow$ Exercise 4.1).[1]

A continuous flow of dividends is modeled by a decrease of S in each time interval $\mathrm{d}t$ by the amount

$$\delta S \, \mathrm{d}t \,,$$

with a constant $\delta \geq 0$. This continuous dividend model can be easily built into the Black–Scholes framework. The standard model of a geometric Brownian motion represented by the SDE (1.47) is generalized to

$$\frac{\mathrm{d}S}{S} = (\mu - \delta)\,\mathrm{d}t + \sigma \,\mathrm{d}W \,,$$

[1] But the corresponding solutions $V(S,t)$ and their early-exercise structure will be different. The Notes and Comments summarize how to correctly compensate for a discrete dividend payment.

with $\mu = r$ according to Remark 1.14. This is the basis for this chapter. The corresponding Black–Scholes equation for the value function $V(S,t)$ is

$$\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + (r-\delta) S \frac{\partial V}{\partial S} - rV = 0\,. \tag{4.1}$$

For constant r, σ, δ, this equation is equivalent to the equation

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2} \tag{4.2}$$

for $y(x,\tau)$ with $0 \le \tau$, $x \in \mathbb{R}$. The equivalence is proved by means of the transformations

$$\begin{aligned} &S = K\mathrm{e}^x, \quad t = T - \tfrac{2\tau}{\sigma^2}\,, \quad q := \tfrac{2r}{\sigma^2}\,, \quad q_\delta := \tfrac{2(r-\delta)}{\sigma^2}\,, \\ &V(S,t) = V\left(K\mathrm{e}^x, T - \tfrac{2\tau}{\sigma^2}\right) =: v(x,\tau) \quad \text{and} \\ &v(x,\tau) =: K \exp\left\{-\tfrac{1}{2}(q_\delta - 1)x - \left(\tfrac{1}{4}(q_\delta - 1)^2 + q\right)\tau\right\} y(x,\tau)\,. \end{aligned} \tag{4.3}$$

For the case of no dividend payments ($\delta = 0$) the derivation was carried out earlier ($\longrightarrow$ Exercise 1.4). For Black–Scholes-type equations with variable $\sigma(S,t)$, see Appendix A.6.

The transformation $S = K\mathrm{e}^x$ is motivated by the observation that the Black–Scholes equation in the version (4.1) has variable coefficients S^j with powers matching the order of the derivative with respect to S. That is, the relevant terms in (4.1) are of the type

$$S^j \frac{\partial^j V}{\partial S^j}, \quad \text{for } j = 0, 1, 2\,.$$

The transformed version in Eq. (4.2) has constant coefficients ($= 1$), which makes implementing numerical algorithms easier.

In view of the time transformation in (4.3) the expiration time $t = T$ is determined in the "new" time by $\tau = 0$, and $t = 0$ is transformed to $\tau_{\max} := \frac{1}{2}\sigma^2 T$. Up to the scaling by $\frac{1}{2}\sigma^2$ the new time variable τ represents the remaining life time of the option. And the original domain of the half strip $S > 0$, $0 \le t \le T$ belonging to (4.1) becomes the strip

$$-\infty < x < \infty, \quad 0 \le \tau \le \frac{1}{2}\sigma^2 T\,, \tag{4.4}$$

on which we are going to approximate a solution $y(x,\tau)$ to (4.2). After that calculation we again apply the transformations of (4.3) to derive out of $y(x,\tau)$ the value of the option $V(S,t)$ in the original variables.

Under the transformations (4.3) the terminal conditions (1.1) and (1.2) become **initial conditions** for $y(x,0)$. A vanilla call, for example, satisfies

$$V(S,T) = \max\{S-K,0\} = K\cdot\max\{\mathrm{e}^x-1,0\}\,.$$

From (4.3) we find

$$V(S,T) = K\exp\left\{-\frac{x}{2}(q_\delta-1)\right\}y(x,0)\,,$$

and thus

$$\begin{aligned} y(x,0) &= \exp\left\{\frac{x}{2}(q_\delta-1)\right\}\max\{\mathrm{e}^x-1,0\} \\ &= \begin{cases} \exp\left\{\frac{x}{2}(q_\delta-1)\right\}(\mathrm{e}^x-1) & \text{for } x>0 \\ 0 & \text{for } x\le 0\,. \end{cases} \end{aligned}$$

Using

$$\exp\left\{\frac{x}{2}(q_\delta-1)\right\}(\mathrm{e}^x-1) = \exp\left\{\frac{x}{2}(q_\delta+1)\right\} - \exp\left\{\frac{x}{2}(q_\delta-1)\right\}$$

the initial conditions $y(x,0)$ for vanilla options in the new variables read

$$\text{call: } y(x,0) = \max\left\{\mathrm{e}^{\frac{x}{2}(q_\delta+1)} - \mathrm{e}^{\frac{x}{2}(q_\delta-1)},\ 0\right\}\,, \tag{4.5}$$

$$\text{put: } y(x,0) = \max\left\{\mathrm{e}^{\frac{x}{2}(q_\delta-1)} - \mathrm{e}^{\frac{x}{2}(q_\delta+1)},\ 0\right\}\,. \tag{4.6}$$

Insofar the PDE (4.2) on the strip (4.4) with initial condition (4.5) or (4.6) defines an initial-value problem. In Sect. 4.4 we shall discuss possible boundary conditions needed when the boundaries $x\to-\infty$ and $x\to+\infty$ are truncated.

The Eq. (4.2) is of the type of a parabolic partial differential equation and is the simplest diffusion or heat-conducting equation. Both Eqs. (4.1) and (4.2) are linear in the dependent variables V or y. The differential equation (4.2) is also written $y_\tau = y_{xx}$ or $\dot{y} = y''$. The diffusion term is y_{xx}.

In principle, the methods of this chapter can be applied directly to (4.1). But the equations and algorithms are easier to derive for the algebraically equivalent version (4.2). Note that numerically the two equations are *not* equivalent. A direct application of this chapter's methods to version (4.1) can cause severe difficulties. This will be discussed in Chap. 6. These difficulties will not occur for Eq. (4.2), which is well-suited for standard options with constant coefficients. The Eq. (4.2) is integrated in forward time—that is, for increasing τ starting from $\tau = 0$. This fact is important for stability investigations. For increasing τ the version (4.2) makes sense; this is equivalent to the well-posedness of (4.1) for *de*creasing t.

4.2 Foundations of Finite-Difference Methods

This section describes the basic ideas of finite differences as they are applied to the PDE (4.2).

4.2.1 Difference Approximation

Each two times continuously differentiable function f satisfies

$$f'(x) = \frac{f(x+h) - f(x)}{h} - \frac{h}{2} f''(\xi)\,,$$

where ξ is an intermediate number between x and $x+h$. The accurate position of ξ is usually unknown. Such expressions are derived by Taylor expansions. We discretize $x \in \mathbb{R}$ by introducing a one-dimensional grid of discrete points x_i with

$$\ldots < x_{i-1} < x_i < x_{i+1} < \ldots$$

For example, choose an equidistant grid with mesh size $h := x_{i+1} - x_i$. The x is discretized, but the function values $f_i := f(x_i)$ are not discrete, $f_i \in \mathbb{R}$. For $f \in \mathcal{C}^2$ the derivative f'' is bounded, and the term $-\frac{h}{2} f''(\zeta)$ can be conveniently written as $O(h)$. This leads to the practical notation

$$f'(x_i) = \frac{f_{i+1} - f_i}{h} + O(h)\,. \tag{4.7}$$

Analogous expressions hold for the partial derivatives of $y(x,\tau)$, which includes a discretization in τ. This suggests to replace the neutral notation h by either Δx or $\Delta\tau$, respectively. The fraction in (4.7) is the difference quotient that approximates the differential quotient $f'(x_i)$; the $O(h^p)$-term is the error. The one-sided (i.e. nonsymmetric) difference quotient (4.7) is of the order $p = 1$. Error orders of $p = 2$ are obtained by central differences

$$f'(x_i) = \frac{f_{i+1} - f_{i-1}}{2h} + O(h^2) \qquad (\text{for } f \in \mathcal{C}^3)\,, \tag{4.8}$$

$$f''(x_i) = \frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + O(h^2) \qquad (\text{for } f \in \mathcal{C}^4)\,, \tag{4.9}$$

or by one-sided differences that involve more terms, such as

$$f'(x_i) = \frac{-f_{i+2} + 4f_{i+1} - 3f_i}{2h} + O(h^2) \qquad (\text{for } f \in \mathcal{C}^3)\,. \tag{4.10}$$

Rearranging terms and indices of (4.10) provides the approximation formula

$$f_i \approx \frac{4}{3} f_{i-1} - \frac{1}{3} f_{i-2} + \frac{2}{3} h f'(x_i)\,. \tag{4.11}$$

The latter difference quotient is an example of a *backward differentiation formula* (BDF). Equidistant grids are advantageous in that algorithms are straightforward to implement, and error terms are easily derived by Taylor's expansion. This chapter works with equidistant grids.

4.2.2 The Grid

Either the x-axis, or the τ-axis, or both can be discretized. If only one of the two independent variables x or τ is discretized, one obtains a semidiscretization consisting of parallel lines. This is used in Exercise 4.3 and in Sect. 4.8.3. Here we perform a full discretization leading to a two-dimensional grid. A solution of the discretized problem will be different from the solution y of the initial-value problem on the strip (4.4). To emphasize the difference, we denote a solution of a discretized version w.

Let $\Delta\tau$ and Δx be the mesh sizes of the discretizations of τ and x. The step in τ is $\Delta\tau := \tau_{\max}/\nu_{\max}$ for $\tau_{\max} = \frac{1}{2}\sigma^2 T$ and a suitable integer $\nu_{\max}$. Selecting the x-discretization is more complicated. The infinite interval $-\infty < x < \infty$ must be replaced by a finite interval $x_{\min} \le x \le x_{\max}$, thereby the strip (4.4) changes to a rectangular domain for (x, τ). This truncation or *localization* will have an impact on the solutions w. The finite end values $a = x_{\min} < 0$ and $b = x_{\max} > 0$ must be chosen[2] such that for the corresponding $S_{\min} = K\mathrm{e}^a$ and $S_{\max} = K\mathrm{e}^b$ and the interval $S_{\min} \le S \le S_{\max}$ a sufficient quality of approximation is obtained, in the sense $w \approx y$. In addition, the interval $x_{\min} \le x \le x_{\max}$ must include the range of financial interest, namely, the x-values of S_0 and K. This requires

$$x_{\min} < \min\left\{0,\, \log\frac{S_0}{K}\right\}\,, \quad \max\left\{0,\, \log\frac{S_0}{K}\right\} < x_{\max}\,.$$

For simplicity, just think of $x_{\min} = -3$ and $x_{\max} = 3$. The localization will also need boundary conditions, which will be discussed in Sect. 4.4.

[2] Too large values of $|a|$ or b can lead to underflow or overflow when evaluating the exponential function.

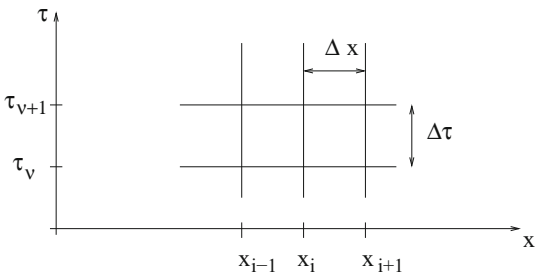

Fig. 4.1 Detail and notations of the grid

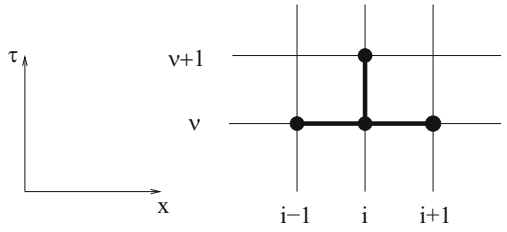

Fig. 4.2 Connection scheme (stencil) of the explicit method

For a suitable integer m the step length in x is defined by $\Delta x := (b-a)/m$. Additional notations for the grid are

$\tau_\nu := \nu \cdot \Delta\tau \quad$ for $\nu = 0, 1, \ldots, \nu_{\max}$
$x_i := a + i\Delta x \quad$ for $i = 0, 1, \ldots, m$
$y_{i,\nu} := y(x_i, \tau_\nu)$,
$w_{i,\nu}$ approximation to $y_{i,\nu}$.

This defines a two-dimensional uniform grid as illustrated in Fig. 4.1. Note that the equidistant grid in this chapter is defined in terms of x and τ, and not for S and t. Transforming the (x, τ)-grid via the transformation in (4.3) back to the (S, t)-plane, leads to a nonuniform grid with unequal distances of the grid lines $S = S_i = Ke^{x_i}$: The grid is increasingly dense close to $S_{\min}$. (This is not advantageous for the accuracy of the approximations of $V(S, t)$. We will come back to this in Sect. 5.2.) Figure 4.1 illustrates only a small part of the entire grid in the (x, τ)-strip. The grid lines $x = x_i$ and $\tau = \tau_\nu$ can be indicated by their indices (Fig. 4.2).

The points where the grid lines $\tau = \tau_\nu$ and $x = x_i$ intersect, are called *nodes*. In contrast to the theoretical solution $y(x, \tau)$, which is defined on a continuum, the $w_{i,\nu}$ are defined only for the nodes. The error $w_{i,\nu} - y_{i,\nu}$ depends on the choice of the discretization parameters $\nu_{\max}$, m, $x_{\min}$, $x_{\max}$. A priori we do not know which choice of the parameters matches a prespecified error tolerance. An example of the order of magnitude of these parameters is given by $x_{\min} = -5, x_{\max} = 5$, or smaller, and $\nu_{\max} = 100$, $m = 100$. Such a choice of $x_{\min}$, $x_{\max}$ has shown to be reasonable for a wide range of r, σ-values and accuracies. Then the actual error is essentially controlled via the numbers $\nu_{\max}$ and m of grid lines.

4.2.3 Explicit Method

Substituting the expressions from (4.7)/(4.10)

$$\frac{\partial y_{i,\nu}}{\partial \tau} := \frac{\partial y(x_i,\tau_\nu)}{\partial \tau} = \frac{y_{i,\nu+1}-y_{i,\nu}}{\Delta\tau} + O(\Delta\tau)$$
$$\frac{\partial^2 y_{i,\nu}}{\partial x^2} = \frac{y_{i+1,\nu}-2y_{i,\nu}+y_{i-1,\nu}}{\Delta x^2} + O(\Delta x^2)$$

into (4.2) and discarding the O-error terms leads to the difference equation

$$\frac{w_{i,\nu+1}-w_{i,\nu}}{\Delta\tau} = \frac{w_{i+1,\nu}-2w_{i,\nu}+w_{i-1,\nu}}{\Delta x^2}$$

for the approximation w. Solved for $w_{i,\nu+1}$ this is

$$w_{i,\nu+1} = w_{i,\nu} + \frac{\Delta\tau}{\Delta x^2}(w_{i+1,\nu}-2w_{i,\nu}+w_{i-1,\nu})\,.$$

With the abbreviation

$$\lambda := \frac{\Delta\tau}{\Delta x^2}$$

the result is written compactly

$$w_{i,\nu+1} = \lambda w_{i-1,\nu} + (1-2\lambda)w_{i,\nu} + \lambda w_{i+1,\nu}\,. \tag{4.12}$$

Figure 4.2 accentuates the nodes that are connected by this formula. Such a graphical scheme illustrating the structure of the equation, is called *stencil* (or molecule).

The Eq. (4.12) and the Fig. 4.2 suggest an evaluation organized by *time levels*. All nodes with the same index ν form the νth time level. For a fixed ν the values $w_{i,\nu+1}$ of the time level $\nu+1$ are calculated for all i. Then we advance to the next time level, $\nu \to \nu+1$. The formula (4.12) is an explicit expression for each of the $w_{i,\nu+1}$; the values w at level $\nu+1$ are not coupled. Since (4.12) provides an explicit formula for all $w_{i,\nu+1}$ $(i = 0,1,\ldots,m)$, this method is called *explicit method* or *forward-difference method*.

Start: For $\nu = 0$ the values of $w_{i,0}$ are given by the initial conditions

$$w_{i,0} = y(x_i,0) \quad \text{for } y \text{ from (4.5)/(4.6)},\ 0 \le i \le m.$$

Thereafter we proceed from $\nu = 0$ to $\nu = 1$, and so on. The $w_{0,\nu}$ and $w_{m,\nu}$ for $1 \le \nu \le \nu_{\max}$ are fixed by boundary conditions. For the next few pages, to simplify matters, we artificially set $w_{0,\nu} = w_{m,\nu} = 0$ for all ν. The correct boundary conditions are deferred to Sect. 4.4.

For the following analysis it is useful to collect all values w of the time level ν into a vector,

$$w^{(\nu)} := (w_{1,\nu}, \ldots, w_{m-1,\nu})^{tr} .$$

The next step towards a vector notation of the explicit method is to introduce the constant $(m-1) \times (m-1)$ tridiagonal matrix

$$A := A_{\text{expl}} := \begin{pmatrix} 1-2\lambda & \lambda & 0 & \cdots & 0 \\ \lambda & 1-2\lambda & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \lambda \\ 0 & \cdots & 0 & \lambda & 1-2\lambda \end{pmatrix} . \tag{4.13}$$

Now the explicit method in matrix-vector notation reads

$$w^{(\nu+1)} = A w^{(\nu)} \quad \text{for } \nu = 0, 1, 2, \ldots \tag{4.14}$$

The formulation with the matrix A of (4.13) and the iteration (4.14) is needed only for theoretical investigations. An actual computer program would rather use the version (4.12). In the vector notation of (4.14), the inner-loop index i does not occur explicitly.

To illustrate the behavior of the explicit method, we perform an experiment with an artificial example, where initial conditions and boundary conditions are not related to finance.

Example 4.1 (Instability) The PDE is $y_\tau = y_{xx}$, initial condition $y(x,0) = \sin \pi x$, $x_0 = 0$, $x_m = 1$, and boundary conditions $y(0,\tau) = y(1,\tau) = 0$ (that is, $w_{0,\nu} = w_{m,\nu} = 0$).

The aim is to calculate an approximation w for one (x,τ), for example, for $x = 0.2$, $\tau = 0.5$. The exact solution is $y(x,\tau) = \mathrm{e}^{-\pi^2 \tau} \sin \pi x$, such that $y(0.2, 0.5) = 0.004227\ldots$.

We carry out two calculations with the same $\Delta x = 0.1$ (hence $0.2 = x_2$), and two different $\Delta\tau$:

(a) $\Delta\tau = 0.0005 \Longrightarrow \lambda = 0.05$,
$0.5 = \tau_{1000}$, $w_{2,1000} \doteq 0.00435$

(b) $\Delta\tau = 0.01 \Longrightarrow \lambda = 1$,
$0.5 = \tau_{50}$, $w_{2,50} \doteq -1.5 * 10^8$ (the actual numbers depend on the computer)

It turns out that the choice of $\Delta\tau$ in (a) has led to a reasonable approximation, whereas the choice in (b) has caused a disaster. Here we have a stability problem!

4.2.4 Stability

Let us perform an error analysis of an iteration $w^{(\nu+1)} = Aw^{(\nu)} + d^{(\nu)}$. The iteration (4.14) is a special case, with matrix A_{expl}, and the vector $d^{(\nu)}$ vanishes for our preliminary boundary conditions $w_{0,\nu} = w_{m,\nu} = 0$. In general we use the same notation w for the theoretical definition of w and for the values of w obtained by numerical calculations in a computer. Since we now discuss rounding errors, we must distinguish between the two meanings. Let $w^{(\nu)}$ denote the vectors theoretically defined by the iteration. Hence, by definition, the $w^{(\nu)}$ are free of rounding errors. But in computational reality, rounding errors are inevitable. We denote the computer-calculated vector by $\overline{w}^{(\nu)}$ and the error vectors by

$$e^{(\nu)} := \overline{w}^{(\nu)} - w^{(\nu)} ,$$

for $\nu \geq 0$. The $\overline{w}$-result can be written

$$\overline{w}^{(\nu+1)} = A\overline{w}^{(\nu)} + d^{(\nu)} + r^{(\nu+1)} ,$$

where the vectors $r^{(\nu+1)}$ represent the rounding errors that occur during the calculation of $A\overline{w}^{(\nu)} + d^{(\nu)}$. Let us concentrate on the effect of the rounding errors that occur for an arbitrary ν, say for ν^*. We ask for the propagation of this error for increasing $\nu > \nu^*$. Without loss of generality we set $\nu^* = 0$, and for simplicity take $r^{(\nu)} = 0$ for $\nu > 1$. That is, we investigate the effect the initial rounding error $e^{(0)}$ has on the iteration. The initial error $e^{(0)}$ represents the rounding error during the evaluation of the initial condition (4.5)/(4.6), when $\overline{w}^{(0)}$ is calculated. According to this scenario, $\overline{w}^{(\nu+1)} = A\overline{w}^{(\nu)} + d^{(\nu)}$ for $\nu > 1$. The relation

$$Ae^{(\nu)} = A\overline{w}^{(\nu)} - Aw^{(\nu)} = \overline{w}^{(\nu+1)} - w^{(\nu+1)} = e^{(\nu+1)}$$

between consecutive errors is applied repeatedly and results in

$$e^{(\nu)} = A^{\nu} e^{(0)} . \tag{4.15}$$

For the method to be *stable*, previous errors must be damped. This leads to require $A^{\nu}e^{(0)} \to 0$ for $\nu \to \infty$. Elementwise this means $\lim_{\nu\to\infty}\{(A^{\nu})_{ij}\} = 0$ for $\nu \to \infty$ and for any pair of indices (i, j). The following lemma provides a criterion for this requirement.

Lemma 4.2

$$\begin{aligned} \rho(A) < 1 &\iff A^{\nu}z \to 0 \text{ for all } z \text{ and } \nu \to \infty \\ &\iff \lim_{\nu\to\infty}\{(A^{\nu})_{i,j}\} = 0 \end{aligned}$$

Here $\rho(A)$ is the *spectral radius* of A,

$$\rho(A) := \max_k |\mu_k^A| ,$$

where $\mu_1^A, \ldots, \mu_{m-1}^A$ denote the eigenvalues of A, labeled with index k. The proof can be found in text books on numerical analysis, for example, in [198]. As a consequence of Lemma 4.2 we require for stable behavior that $|\mu_k^A| < 1$ for all eigenvalues, here for $k = 1, \ldots, m-1$. To check the criterion of Lemma 4.2, the eigenvalues μ_k^A of A are needed. The matrix A can be written

$$A = I - \lambda \cdot \underbrace{\begin{pmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{pmatrix}}_{=:G} .$$

It remains to investigate the eigenvalues μ^A or μ^G of the tridiagonal matrices A or G.[3]

Lemma 4.3 *Let*

$$G = \begin{pmatrix} \alpha & \beta & & 0 \\ \gamma & \ddots & \ddots & \\ & \ddots & \ddots & \beta \\ 0 & & \gamma & \alpha \end{pmatrix}$$

be an N^2-matrix. The eigenvalues μ_k^G are

$$\mu_k^G = \alpha + 2\beta\sqrt{\frac{\gamma}{\beta}} \cos\frac{k\pi}{N+1} , \; k = 1, \ldots, N .$$

Proof The eigenvectors $v^{(k)}$ of G are

$$v^{(k)} = \left(\sqrt{\frac{\gamma}{\beta}} \sin\frac{k\pi}{N+1}, \left(\sqrt{\frac{\gamma}{\beta}}\right)^2 \sin\frac{2k\pi}{N+1}, \ldots, \left(\sqrt{\frac{\gamma}{\beta}}\right)^N \sin\frac{Nk\pi}{N+1} \right)^{tr} .$$

Substitute this into $Gv = \mu^G v$. □

[3] The zeros in the corner of the matrix G symbolize the triangular zero structure of (4.13).

To apply Lemma 4.3 to A or to G observe $N = m-1$, and for G $\alpha = 2, \beta = \gamma = -1$. Accordingly, the eigenvalues μ^G and the eigenvalues μ^A are

$$\mu_k^G = 2 - 2\cos\frac{k\pi}{m} = 4\sin^2\left(\frac{k\pi}{2m}\right),$$

$$\mu_k^A = 1 - \lambda\mu^G = 1 - 4\lambda\sin^2\frac{k\pi}{2m}.$$

Now we can state the stability requirement $|\mu_k^A| < 1$ as

$$\left|1 - 4\lambda\sin^2\frac{k\pi}{2m}\right| < 1, \quad k = 1, \ldots, m-1.$$

This implies the two inequalities $\lambda > 0$ and

$$-1 < 1 - 4\lambda\sin^2\frac{k\pi}{2m}, \quad \text{rewritten as} \quad \frac{1}{2} > \lambda\sin^2\frac{k\pi}{2m}.$$

The largest sin-term is $\sin\frac{(m-1)\pi}{2m}$; for increasing m this term grows monotonically approaching 1. In summary we have shown for (4.13)/(4.14)

$$\text{For } 0 < \lambda \le \frac{1}{2} \text{ the explicit method } w^{(\nu+1)} = Aw^{(\nu)} \text{ is stable.}$$

In view of $\lambda = \Delta\tau/\Delta x^2$ this stability criterion amounts to bounding the $\Delta\tau$ step size,

$$0 < \Delta\tau \le \frac{\Delta x^2}{2}. \tag{4.16}$$

This explains what happened with Example 4.1. The values of λ in the two cases of this example are

$$\text{(a)} \quad \lambda = 0.05 \le \frac{1}{2},$$

$$\text{(b)} \quad \lambda = 1 > \frac{1}{2}.$$

In case (b) the chosen $\Delta\tau$ and hence λ were too large, which led to an amplification of rounding errors resulting eventually in the "explosion" of the w-values.

The explicit method is stable only as long as (4.16) is satisfied. As a consequence, the parameters m and $\nu_{\max}$ of the grid resolution can not be chosen independent of each other. If the demands for accuracy are high, the step size Δx will be small, which in view of (4.16) bounds $\Delta\tau$ quadratically. This situation suggests searching for a method that is unconditionally stable.

4.2.5 An Implicit Method

Introducing the explicit method in Sect. 4.2.3, we have approximated the time derivative with a forward difference, "forward" as seen from the νth time level. Now we try the backward difference in

$$\frac{\partial y_{i,\nu}}{\partial \tau} = \frac{y_{i,\nu} - y_{i,\nu-1}}{\Delta\tau} + O(\Delta\tau)\,,$$

which yields the alternative to (4.12)

$$-\lambda w_{i+1,\nu} + (1+2\lambda) w_{i,\nu} - \lambda w_{i-1,\nu} = w_{i,\nu-1}\,. \tag{4.17}$$

The Eq. (4.17) relates the time level ν to the time level $\nu - 1$. For the transition from $\nu - 1$ to ν only the value $w_{i,\nu-1}$ on the right-hand side of (4.17) is known, whereas on the left-hand side of the equation three unknown values of w wait to be computed. Equation (4.17) couples three unknowns. The corresponding stencil is shown in Fig. 4.3. There is no simple explicit formula with which the unknowns can be obtained one after the other. Rather a system must be considered, all equations simultaneously. A vector notation reveals the structure of (4.17): With the matrix

$$A := A_{\text{impl}} := \begin{pmatrix} 1+2\lambda & -\lambda & & 0 \\ -\lambda & \ddots & \ddots & \\ & \ddots & \ddots & -\lambda \\ 0 & & -\lambda & 1+2\lambda \end{pmatrix} \tag{4.18}$$

the vector $w^{(\nu)}$ is implicitly defined as solution of the system of linear equations $Aw^{(\nu)} = w^{(\nu-1)}$. To have a consistent numbering, rewrite this as

$$Aw^{(\nu+1)} = w^{(\nu)} \quad \text{for } \nu = 0, \ldots, \nu_{\max} - 1\,. \tag{4.19}$$

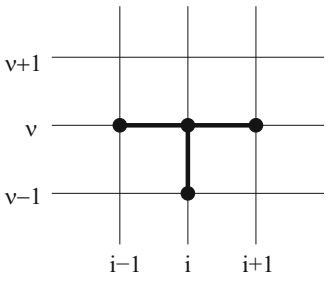

Fig. 4.3 Stencil of the backward-difference method (4.17)

(Again we set $w_{0,\nu} = w_{m,\nu} = 0$.) For each time level ν such a system of equations must be solved. This method is sometimes called *implicit method*. But to distinguish it from other implicit methods, we call it *fully implicit*, or *backward-difference method*, or more accurately *backward time centered space* scheme (BTCS). The method is unconditionally stable for all $\Delta\tau > 0$. This is shown analogously as in the explicit case (⟶ Exercise 4.2). The costs of this implicit method are low, because the matrix A is constant and tridiagonal. Initially, for $\nu = 0$, the LR-decomposition (⟶ Appendix C.1) is calculated once. Then the costs for each ν are only of the order $O(m)$.

4.3 Crank–Nicolson Method

For the methods of the previous section the discretizations of $\frac{\partial y}{\partial \tau}$ are of the order $O(\Delta\tau)$. It seems preferable to use a method where the time discretization of $\frac{\partial y}{\partial \tau}$ has the better order $O(\Delta\tau^2)$, and the stability is unconditional. Let us again consider Eq. (4.2), the equivalent to the Black–Scholes equation,

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2}.$$

Crank and Nicolson suggested to average the forward- and the backward difference method. For easy reference, we collect the underlying approaches from the above: forward for ν:

$$\frac{w_{i,\nu+1} - w_{i,\nu}}{\Delta\tau} = \frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2}$$

backward for $\nu + 1$:

$$\frac{w_{i,\nu+1} - w_{i,\nu}}{\Delta\tau} = \frac{w_{i+1,\nu+1} - 2w_{i,\nu+1} + w_{i-1,\nu+1}}{\Delta x^2}$$

Addition yields

$$\frac{w_{i,\nu+1} - w_{i,\nu}}{\Delta\tau} = \frac{1}{2\Delta x^2}(w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu} + w_{i+1,\nu+1} - 2w_{i,\nu+1} + w_{i-1,\nu+1}) \tag{4.20}$$

The Eq. (4.20) involves in each of the time levels ν and $\nu + 1$ three values of w (Fig. 4.4). This is the basis of an efficient method. Its features are summarized in Theorem 4.4.

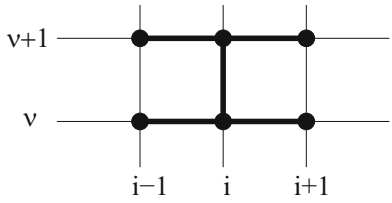

Fig. 4.4 Stencil of the Crank–Nicolson method (4.20)

Theorem 4.4 (Crank–Nicolson)

1.) For $y \in \mathcal{C}^4$ the order of the method is $O(\Delta\tau^2) + O(\Delta x^2)$.
2.) For each ν a linear system of a simple tridiagonal structure must be solved.
3.) Stability holds for all $\Delta\tau > 0$.

Proof

1.) order: A practical notation for the symmetric difference quotient of second order for y_{xx} is

$$\delta_{xx} w_{i,\nu} := \frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2} . \tag{4.21}$$

Apply the operator δ_{xx} to the exact solution y. Then by Taylor expansion for $y \in \mathcal{C}^4$ one shows

$$\delta_{xx} y_{i,\nu} = \frac{\partial^2}{\partial x^2} y_{i,\nu} + \frac{\Delta x^2}{12} \frac{\partial^4}{\partial x^4} y_{i,\nu} + O(\Delta x^4) .$$

The *local discretization error* ϵ describes how well the exact solution y of (4.2) satisfies the difference scheme,

$$\epsilon := \frac{y_{i,\nu+1} - y_{i,\nu}}{\Delta\tau} - \frac{1}{2}(\delta_{xx} y_{i,\nu} + \delta_{xx} y_{i,\nu+1}) .$$

Applying the operator δ_{xx} of (4.21) to the expansion of $y_{i,\nu+1}$ at τ_ν and observing $y_\tau = y_{xx}$ leads to

$$\epsilon = O(\Delta\tau^2) + O(\Delta x^2) .$$

($\longrightarrow$ Exercise 4.4)

2.) system of equations: With $\lambda := \frac{\Delta\tau}{\Delta x^2}$ the Eq. (4.20) is rewritten

$$\begin{aligned} &-\tfrac{\lambda}{2} w_{i-1,\nu+1} + (1+\lambda) w_{i,\nu+1} - \tfrac{\lambda}{2} w_{i+1,\nu+1} \\ &= \tfrac{\lambda}{2} w_{i-1,\nu} + (1-\lambda) w_{i,\nu} + \tfrac{\lambda}{2} w_{i+1,\nu} . \end{aligned} \tag{4.22}$$

The values of the new time level $\nu + 1$ are implicitly given by the system of Eqs. (4.22). For the simplest boundary conditions $w_{0,\nu} = w_{m,\nu} = 0$ Eq. (4.22) is

a system of $m-1$ equations. With matrices

$$A := A_{\mathrm{CN}} := \begin{pmatrix} 1+\lambda & -\frac{\lambda}{2} & & 0 \\ -\frac{\lambda}{2} & \ddots & \ddots & \\ & \ddots & \ddots & -\frac{\lambda}{2} \\ 0 & & -\frac{\lambda}{2} & 1+\lambda \end{pmatrix},$$
$$B := B_{\mathrm{CN}} := \begin{pmatrix} 1-\lambda & \frac{\lambda}{2} & & 0 \\ \frac{\lambda}{2} & \ddots & \ddots & \\ & \ddots & \ddots & \frac{\lambda}{2} \\ 0 & & \frac{\lambda}{2} & 1-\lambda \end{pmatrix} \tag{4.23}$$

the system (4.22) is rewritten

$$Aw^{(\nu+1)} = Bw^{(\nu)}. \tag{4.24}$$

The eigenvalues of A are real and lie between 1 and $1+2\lambda$ (follows from the Theorem of Gerschgorin, see Appendix C.1). This rules out a zero eigenvalue, and so A must be nonsingular and the solution $w^{(\nu+1)}$ of (4.24) is uniquely defined.

3.) stability: Both matrices A and B can be rewritten in terms of a constant tridiagonal matrix,

$$A = I + \frac{\lambda}{2}G, \quad G := \begin{pmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{pmatrix}, \quad B = I - \frac{\lambda}{2}G.$$

Now the Eq. (4.24) reads

$$\begin{aligned} \underbrace{(2I+\lambda G)}_{=:C} w^{(\nu+1)} &= (2I-\lambda G)w^{(\nu)} \\ &= (4I-2I-\lambda G)w^{(\nu)} \\ &= (4I-C)w^{(\nu)}, \end{aligned}$$

which leads to the formally explicit iteration

$$w^{(\nu+1)} = (4C^{-1}-I)w^{(\nu)}. \tag{4.25}$$

The eigenvalues μ_k^C of C for $k = 1, \ldots, m-1$ are known from Sect. 4.2.4,

$$\mu_k^C = 2 + \lambda \mu_k^G = 2 + \lambda(2 - 2\cos\frac{k\pi}{m}) = 2 + 4\lambda \sin^2 \frac{k\pi}{2m}\,.$$

In view of (4.25) we require for a stable method that for all k

$$\left| \frac{4}{\mu_k^C} - 1 \right| < 1\,.$$

This is guaranteed because of $\mu_k^C > 2$. Consequently, the Crank–Nicolson method (4.20)/(4.23)/(4.24) is unconditionally stable for all $\lambda > 0$ $(\Delta\tau > 0)$.

Although correct boundary conditions are still lacking, it makes sense to formulate the basic version of the Crank–Nicolson algorithm for the PDE (4.2).

Algorithm 4.5 (Crank-Nicolson)

Start: Choose m, $\nu_{\max}$; *calculate* $\Delta x, \Delta\tau$
$w_i^{(0)} = y(x_i, 0)$ *with* y *from (4.5) or (4.6),* $0 \le i \le m$.
Calculate the LR-decomposition of A.
loop: for $\nu = 0, 1, \ldots, \nu_{\max} - 1$:
Calculate $c := Bw^{(\nu)}$ *(preliminary).*
Solve $Ax = c$ *using e.g. the LR-decomposition—*
that is, solve $Lz = Bw^{(\nu)}$ *and* $Rx = z$.
$w^{(\nu+1)} := x$

The *LR*-decomposition is the symbol for the solution of the system of linear equations. Later we shall see when to replace it by the *RL*-decomposition. Obviously the matrices A and B are not stored in the computer. Next we show how the vector c in Algorithm 4.5 is modified to realize correct boundary conditions.

4.4 Boundary Conditions

On the unbounded domain $-\infty < x < \infty$ the initial-value problem $y_\tau = y_{xx}$ with initial condition (4.5)/(4.6) and $\tau \ge 0$ is well-posed. But the truncation to the interval $x_{\min} \le x \le x_{\max}$ changes the type of the problem. To make the PDE-problem well-posed in the finite-domain case, boundary conditions must be imposed artificially. They are not stated in the option's contract, and are not needed by Monte Carlo or tree methods. Boundary conditions are the price one has to pay when PDE-based approaches are applied. Since boundary conditions are often approximations of the reality, the "localized solution" on the finite domain $x_{\min} \le x \le x_{\max}$ in general is different from the solution of the pure initial-value problem. For simplicity, we neglect this difference, and denote the localized solution again by y. We need to formulate boundary conditions such that the localized solution is close

to the solution of the original problem. The choice of boundary conditions is not unique.

In the variety of possible boundary conditions there are two kinds so important and so frequent that they have names. For *Dirichlet conditions*, a value is assigned to y, whereas a *Neumann condition* assigns a value to the derivative $\mathrm{d}y/\mathrm{d}x$. For a call, for example, $y(x_{\min}) = 0$ is Dirichlet, and $\frac{\partial y(x_{\max})}{\partial x} = 1$ is Neumann. More generally, with x_{b} standing for $x_{\min}$ or $x_{\max}$,

$$y(x_{\mathrm{b}}, t) = \alpha(t)$$

for some function $\alpha(t)$ is an example of a Dirichlet condition. A discretized version is $w_{0,\nu} = \alpha(\tau_\nu)$. That is, our preliminary boundary conditions $w_{0,\nu} = w_{m,\nu} = 0$ have been of Dirichlet type. And a Neumann condition would be

$$\frac{\partial y(x_{\mathrm{b}}, t)}{\partial x} = \beta(t)$$

for some function $\beta(t)$. On our grid, a second-order approximation (4.8) for this Neumann condition is

$$w_{1,\nu} - w_{-1,\nu} = \beta(\tau_\nu)\, 2\Delta x\,,$$

which uses a fictive grid point x_{-1} outside the interval. The required information on $w_{-1,\nu}$ is provided by a discretized version of the PDE. Alternatively, the one-sided second-order difference quotient (4.10) can be applied. As a result, one or more entries of the matrix A would change, which makes a finite-difference realization of a Neumann condition a bit cumbersome. Dirichlet conditions are easier to cope with. Let us try to analyze the value function $V(S, t)$ for $S = 0$ and $S \to \infty$ in order to derive Dirichlet conditions for $S_{\min}$ and $S_{\max}$, and for

$$\begin{aligned} &y(x, \tau) \text{ for } x = x_{\min} \text{ and } x_{\max} \text{ and all } \tau\,, \quad \text{or} \\ &w_{0,\nu} \text{ and } w_{m,\nu} \text{ for } \nu = 1, \ldots, \nu_{\max}\,, \end{aligned}$$

all consistent with the Black-Scholes model.

Accordingly, we assume GBM paths satisfying (1.47). Then an initial value $S_0 = 0$ causes $S_t = 0$ for all t, and $S_0 \to \infty$ implies that S_t is arbitrarily large, at least larger than the strike K. The boundary conditions for the expiration time $t = T$ are obviously given by the payoff Ψ. This gives rise to the simplest cases of boundary conditions for $t < T$: As motivated by Figs. 1.1 and 1.2 and Eqs. (1.1), (1.2), the value V_{C} of a call and the value V_{P} of a put must satisfy

$$\begin{aligned} V_{\mathrm{C}}(S, t) &= 0 \quad \text{for } S = 0, \quad \text{and} \\ V_{\mathrm{P}}(S, t) &\to 0 \quad \text{for } S \to \infty \end{aligned} \tag{4.26}$$

also for all $t < T$. This follows, for example, from the integral representation (3.25), because discounting does not affect the value 0 of the payoff. Hence the value V_C for $S = 0$ can be predicted safely, as well as V_P for $S(0) \to \infty$. These arguments hold for European as well as for American options, with or without dividend payments.

The boundary conditions on each of the "other sides" of S, where $V \neq 0$, are more difficult. We postpone the boundary conditions for American options to the next section, and investigate European options in this section.

From (4.26) and the put-call parity (⟶ Exercise 1.1) we deduce the additional boundary conditions for European options. The result is

$$\begin{aligned} V_C(S,t) &= S - Ke^{-r(T-t)} \quad \text{for } S \to \infty \\ V_P(S,t) &= Ke^{-r(T-t)} - S \quad \text{for } S \to 0 \end{aligned} \tag{4.27}$$

(without dividend payment, $\delta = 0$). The lower bounds for European options (⟶ Appendix E.1) are attained at the boundaries. In (4.27) for $S \approx 0$ we do not discard the term S, because the realization of the transformation (4.3) requires $S_{\min} > 0$, see Sect. 4.2.2.[4] Boundary conditions analogous as in (4.27) hold for the case of a continuous flow of dividend payments ($\delta > 0$). We skip the derivation, which can be based on transformation (4.3) and the additional transformation $S = \overline{S}e^{\delta(T-t)}$ (⟶ Exercise 4.5). In summary, the asymptotic boundary conditions for European options in the (x,τ)-world are as follows:

Boundary Conditions 4.6 (European Options)

$$\begin{aligned} &y(x,\tau) = r_1(x,\tau) \textit{ for } x \to -\infty\,, \\ &y(x,\tau) = r_2(x,\tau) \textit{ for } x \to \infty\,, \quad \textit{with} \\ &\textit{call: } r_1(x,\tau) := 0\,, \\ &\qquad r_2(x,\tau) := \exp\left(\tfrac{1}{2}(q_\delta+1)x + \tfrac{1}{4}(q_\delta+1)^2\tau\right)\,, \\ &\textit{put: } r_1(x,\tau) := \exp\left(\tfrac{1}{2}(q_\delta-1)x + \tfrac{1}{4}(q_\delta-1)^2\tau\right)\,, \\ &\qquad r_2(x,\tau) := 0\,. \end{aligned} \tag{4.28}$$

Truncation What can we state about $S_{\min}$ and $S_{\max}$? We need boundary conditions for the finite interval

$$a := x_{\min} \le x \le x_{\max} =: b\,.$$

The probability that $S_T < K$ when $S_0 = S_{\min}$ can be estimated by the transition density (1.64). By the same argument the probability is known that $S_T > K$ when $S_0 = S_{\max}$. Both probabilities are large as long as $S_{\min}$ is small and $S_{\max}$ large enough. This situation suggests to apply the boundary conditions (4.26) and (4.27) also to the left-hand boundary $S_{\min}$ and to the right-hand boundary $S_{\max}$.

[4] For $S = 0$ the PDE is no longer parabolic.

Although (4.28) is valid only for $x \to -\infty$ and $x \to \infty$, we apply the dominant terms $r_1(x,\tau)$ and $r_2(x,\tau)$ to approximate boundary conditions at $x = a$ and $x = b$. This leads to the boundary conditions

$$\begin{aligned} w_{0,\nu} &= r_1(a,\tau_\nu) \\ w_{m,\nu} &= r_2(b,\tau_\nu) \end{aligned}$$

for all ν.

These approximations are explicit formulas and easy to implement. To this end return to the Crank–Nicolson equation (4.22), in which some of the terms on both sides of the equations are known by the boundary conditions. For the equation with $i = 1$ these are terms

$$\begin{aligned} &\text{from the left-hand side:} && -\frac{\lambda}{2} w_{0,\nu+1} = -\frac{\lambda}{2} r_1(a,\tau_{\nu+1}) \,, \\ &\text{from the right-hand side:} && \frac{\lambda}{2} w_{0,\nu} = \frac{\lambda}{2} r_1(a,\tau_\nu) \,, \end{aligned}$$

and for $i = m - 1$

$$\begin{aligned} &\text{from the left-hand side:} && -\frac{\lambda}{2} w_{m,\nu+1} = -\frac{\lambda}{2} r_2(b,\tau_{\nu+1}) \,, \\ &\text{from the right-hand side:} && \frac{\lambda}{2} w_{m,\nu} = \frac{\lambda}{2} r_2(b,\tau_\nu) \,. \end{aligned}$$

These known boundary values are collected on the right-hand side of system (4.22). So we finally arrive at

$$\begin{aligned} A w^{(\nu+1)} &= B w^{(\nu)} + d^{(\nu)} \\ d^{(\nu)} &:= \tfrac{\lambda}{2} \cdot \begin{pmatrix} r_1(a,\tau_{\nu+1}) + r_1(a,\tau_\nu) \\ 0 \\ \vdots \\ 0 \\ r_2(b,\tau_{\nu+1}) + r_2(b,\tau_\nu) \end{pmatrix} \end{aligned} \tag{4.29}$$

The preliminary version (4.24) is included as special case, with $d^{(\nu)} = 0$. The statement in Algorithm 4.5 that defines c is modified to the statement

$$\text{Calculate } c := B w^{(\nu)} + d^{(\nu)} \,.$$

The methods of Sect. 4.2 can be adapted by analogous formulas. The matrix A is not changed, and the stability is not affected by adding the vector d, which is constant with respect to w.

4.5 Early-Exercise Structure

In Sects. 4.1 through 4.3 we have considered tools for the Black–Scholes differential equation—that is, we have investigated European options. Now we turn our attention to American options. Recall that the value of an American option can never be smaller than the value of a European option,

$$V^{\mathrm{Am}} \geq V^{\mathrm{Eur}}.$$

In addition, an American option has at least the value of the payoff Ψ. So we have elementary lower bounds for the value of American options, but—as we shall see—additional numerical problems to cope with.

4.5.1 Early-Exercise Curve

A European option can have a value that is smaller than the payoff (compare, for example, Fig. 1.6). This can not happen with American options. Recall the arbitrage strategy: if for instance an American put would have a value $V_{\mathrm{P}}^{\mathrm{Am}} < (K-S)^+$, one would simultaneously purchase the asset and the put, and exercise immediately. An analogous arbitrage argument implies that for an American call the situation $V_{\mathrm{C}}^{\mathrm{Am}} < (S-K)^+$ can not prevail. Therefore the inequalities

$$\begin{aligned} V_{\mathrm{P}}^{\mathrm{Am}}(S,t) &\geq (K-S)^+ \quad \text{for all } (S,t) \\ V_{\mathrm{C}}^{\mathrm{Am}}(S,t) &\geq (S-K)^+ \quad \text{for all } (S,t) \end{aligned} \tag{4.30}$$

hold. For a put this is illustrated schematically in Fig. 4.5. The inequalities for V make the problem of calculating an American option nonlinear.

For American options we have stated in (4.26) the boundary conditions that prescribe $V = 0$. The boundary conditions at each of the other "ends" of the S-axis are still needed. In view of the inequalities (4.30) it is clear that the missing boundary conditions will be of a different kind than those for European options,

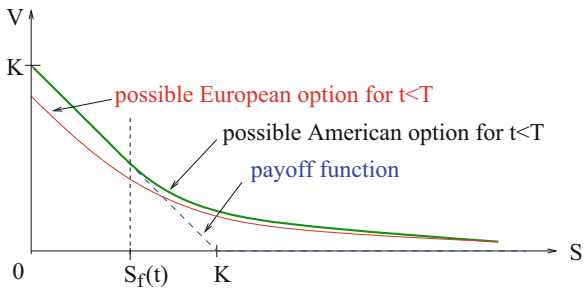

Fig. 4.5 $V(S,t)$ for a put and a $t < T$, schematically

which are listed in (4.27). Let us investigate the situation of an **American put**, which is illustrated in Fig. 4.5. First discuss the left-end part of the *curve* $V_P(S,t)$, for small $S > 0$, and some $t < T$. Without the possibility of early exercise the inequality $V_P^{Am}(S,t) = V_P^{Eur}(S,t) < K - S$ holds for $r > 0$ and sufficiently small S. But in view of (4.30) the American put should satisfy $V_P^{Am}(S,t) \equiv K - S$ at least for small S. To understand what happens for "medium" values of S, imagine to approach from the right-hand side, where $V_P^{Am}(S,t) > (K-S)^+$. Continuity and monotony of V_P suggest that the curve $V_P^{Am}(S,t)$ merges into the straight line $K - S$ of the payoff at some value S_f in the interval $0 < S_f < K$, see Fig. 4.5. This **contact point** S_f is defined by

$$\begin{aligned} V_P^{Am}(S,t) &> (K-S)^+ \quad \text{for } S > S_f(t), \\ V_P^{Am}(S,t) &= K - S \qquad \text{for } S \le S_f(t)\,. \end{aligned} \tag{4.31}$$

Convexity of $V(S,.)$ guarantees that there is only one contact point S_f for each t. For $S < S_f$ the value V_P^{Am} equals the straight line of the payoff and nothing needs to be calculated. For each t, the *curve* $V_P^{Am}(S,t)$ reaches its left boundary at $S_f(t)$.

The above situation holds for any $t < T$, and the contact point S_f varies with t, $S_f = S_f(t)$. For all $0 \le t < T$, the contact points $S_f(t)$ form a curve in the (S,t)-half strip. This curve S_f is the boundary separating the area with $V >$ payoff from the area with $V =$ payoff. The curve S_f of a put is illustrated in the left-hand diagram of Fig. 4.6. A priori the location of the boundary S_f is unknown, the curve is "free." This explains why the problem of calculating $V_P^{Am}(S,t)$ for $S > S_f(t)$ is called **free boundary problem**.

For **American calls** the situation is similar, except that the contact only occurs for dividend-paying assets, $\delta \neq 0$. This is seen from

$$V_C^{Am} \ge V_C^{Eur} \ge S - Ke^{-r(T-t)} > S - K$$

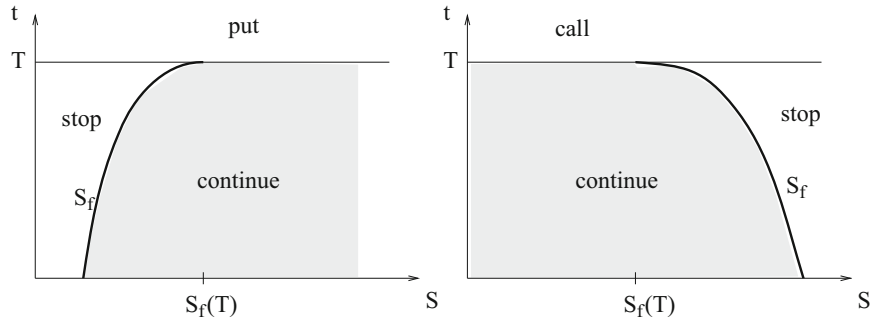

Fig. 4.6 Continuation region (*shaded*) and stopping region for American options: put (*left*) and call (*right*)

for $\delta = 0$, $r > 0$, $t < T$, compare Exercise 1.1. $V_C^{Am} > S - K$ for $\delta = 0$ implies that early-exercise does not pay. American and European calls on assets that pay no dividends are identical, $V_C^{Am} = V_C^{Eur}$. A typical curve $V_C^{Am}(S, t)$ for $\delta \neq 0$ contacting the payoff is shown in Fig. 4.9. And the free boundary S_f qualitatively looks like the right-hand diagram of Fig. 4.6.

The notation $S_f(t)$ for the free boundary is motivated by the process of solving PDEs. But the primary meaning of the curve S_f is economical. The free boundary S_f is the **early-exercise curve**. The time instance t_s when a price process S_t reaches the early-exercise curve is the optimal stopping time, compare also the illustration of Fig. 3.10. Let us explain this for the case of a put; for a call with dividend payment the argument is similar.

For a put, in case $S > S_f$, early-exercise causes an immediate loss, because (4.31) implies the exercise balance $-V + K - S < 0$. Receiving the strike price K does not compensate the loss of S and V. Accordingly, the rational holder of the option does not exercise when $S > S_f$. This explains why the area $S > S_f$ is called **continuation region**[5] (shaded in Fig. 4.6).

On the other side of the boundary curve S_f, characterized by $V = K - S$, each change of S is compensated by a corresponding move of V. Here the only way to create a profit is to exercise and invest the proceeds K at the risk-free rate r for the remaining time period $T - t$. The resulting profit will be

$$K\mathrm{e}^{r(T-t)} - K\,,$$

which relies on $r > 0$. (For $r = 0$ American and European put are identical.) To maximize the profit, the holder of the option will maximize $T - t$, and accordingly exercises as soon as $V \equiv K - S$ is reached. Hence, the boundary curve S_f is the early-exercise curve. And the area $S \leq S_f$ is called **stopping region**.[6]

Now that the curve S_f is recognized as having such a distinguished importance as early-exercise curve, we should make sure that the properties of S_f are as suggested by Figs. 4.6 and 4.7. In fact, the curves $S_f(t)$ are continuously differentiable in t, and monotone not decreasing/not increasing as illustrated. For more details see Appendix A.5. Here we confine ourselves to the bounds given by the limit $t \to T$ $(t < T)$:

$$\text{put:} \quad \lim_{t \to T^-} S_f(t) = \begin{cases} K & \text{for } 0 \leq \delta \leq r \\ \frac{r}{\delta} K & \text{for } r < \delta \end{cases} \tag{4.32}$$

$$\text{call:} \quad \lim_{t \to T^-} S_f(t) = \max(K, \frac{r}{\delta} K) \quad \text{for } \delta > 0 \tag{4.33}$$

[5] Of course, the holder may wish to sell the option.

[6] The final balance for a put after exercising is $K\mathrm{e}^{r(T-t)}$. The reader is encouraged to show that holding is less profitable ($S\mathrm{e}^{\delta(T-t)} < K\mathrm{e}^{r(T-t)}$), at least for small $r(T-t)$. When a discrete dividend is paid, the stopping region is not necessarily connected (⟶ Exercise 4.1b).

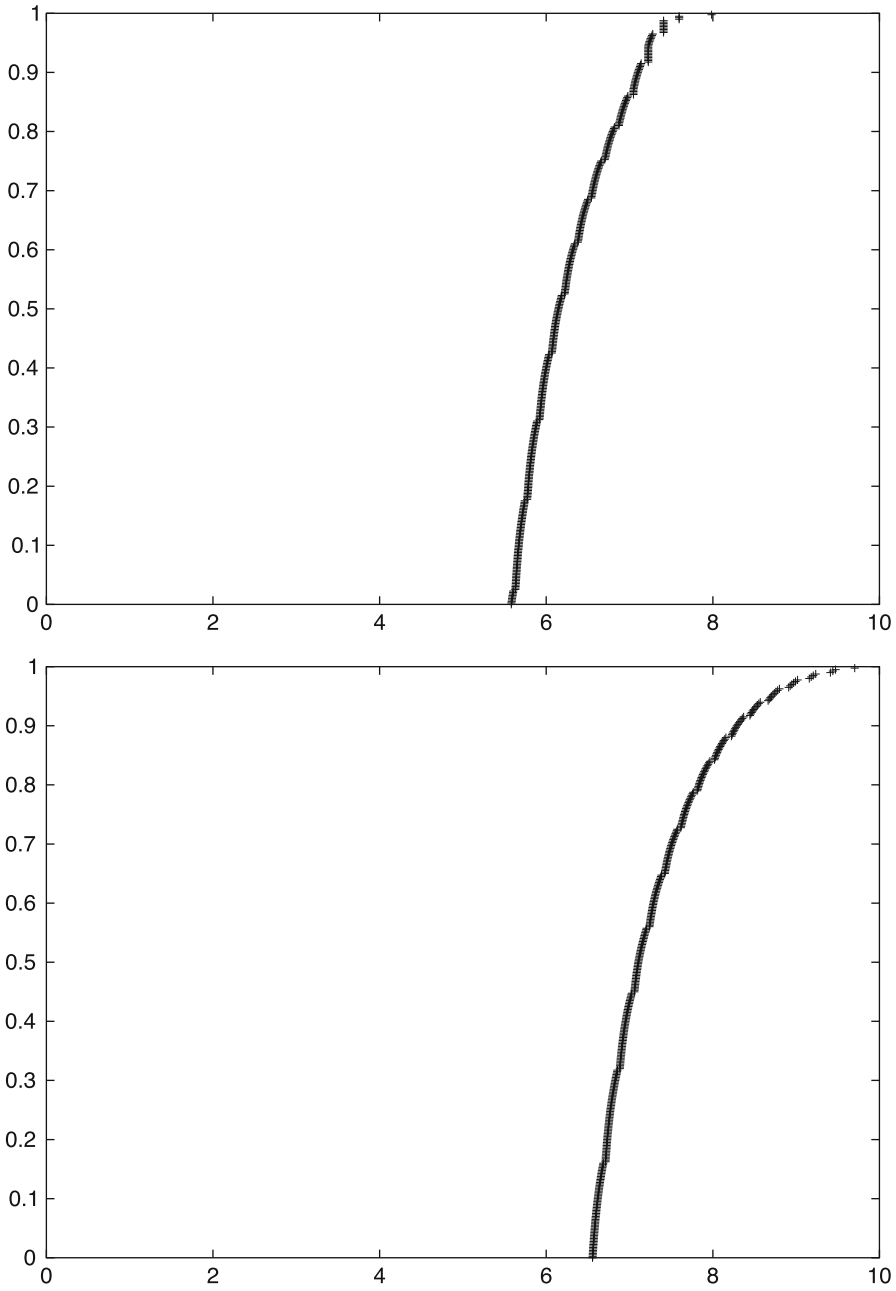

Fig. 4.7 Early-exercise curves of an American put in (S, t)-planes, $r = 0.06$, $\sigma = 0.3$, $K = 10$, and dividend rates $\delta = 0.08$ (*top*), $\delta = 0.04$ (*bottom*); raw data of a finite-difference calculation without interpolation or smoothing

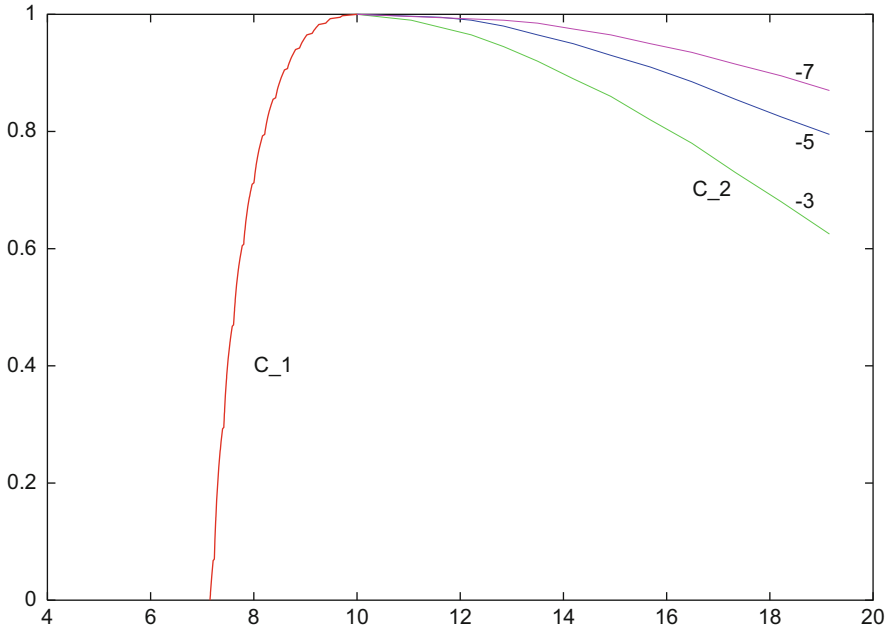

Fig. 4.8 (S, t)-plane, calculated curves of a put matching Figs. 1.4 and 1.5. C_1 is the curve S_f. The three curves C_2 have the meaning $V < 10^{-k}$ for $k = 3, 5, 7$

These bounds express a qualitatively different behavior of the early-exercise curve in the two situations $0 < \delta < r$ and $\delta > r$. This is illustrated in Fig. 4.7 for a put. For the chosen numbers, for all $\delta \leq 0.06$ the limit of (4.32) is the strike K (lower diagram). Compare to Figs. 1.4 and 1.5 to get a feeling for the geometrical importance of the curve as contact line where two surfaces merge. For large values of S the surface $V(S, t)$ approaches 0 in a way illustrated by Fig. 4.8.

4.5.2 Free-Boundary Problem

Again we start with a put. For the European option, the left-end boundary condition is formulated for $S = 0$. For the American option, the left-end boundary is given along the curve S_f (Fig. 4.5). In order to calculate the free boundary $S_f(t)$ one needs an additional condition. To this end consider the right-hand slope $\frac{\partial V}{\partial S}$ with which $V_P^{Am}(S, t)$ touches at $S_f(t)$ the straight line $K - S$, which has the constant slope -1. By geometrical reasons we can rule out the case $\frac{\partial V(S_f(t),t)}{\partial S} < -1$ for V_P^{Am}, because otherwise (4.30) and (4.31) would be violated. Using arbitrage arguments, the case $\frac{\partial V(S_f(t),t)}{\partial S} > -1$ can be ruled out as well ($\longrightarrow$ Exercise 4.6). It remains the condition $\partial V_P^{Am}(S_f(t), t)/\partial S = -1$. That is, $V(S, t)$ touches the payoff function *tangentially*.

This tangency condition is commonly called the *high-contact condition*, or *smooth pasting*. For the case of an option without maturity (*perpetual option*, $T = \infty$) the tangential touching can be calculated analytically (⟶ Exercise 4.7). In summary, *two* boundary conditions must hold at the contact point $S_f(t)$:

$$\begin{aligned} V_P^{Am}(S_f(t), t) &= K - S_f(t) \\ \frac{\partial V_P^{Am}(S_f(t), t)}{\partial S} &= -1 \end{aligned} \tag{4.34}$$

As before, the right-end boundary condition $V_P(S, t) \to 0$ must be observed for $S \to \infty$.

For **American calls** analogous boundary conditions can be formulated. For a call in case $\delta > 0$, $r > 0$ the free boundary conditions

$$\begin{aligned} V_C^{Am}(S_f(t), t) &= S_f(t) - K \\ \frac{\partial V_C^{Am}(S_f(t), t)}{\partial S} &= 1 \end{aligned} \tag{4.35}$$

must hold along the right-end boundary for $S_f(t) > K$. The left-end boundary condition at $S = 0$ remains unchanged. Figure 4.9 shows the situation of an American call on a dividend-paying asset. The high contact on the payoff is visible.

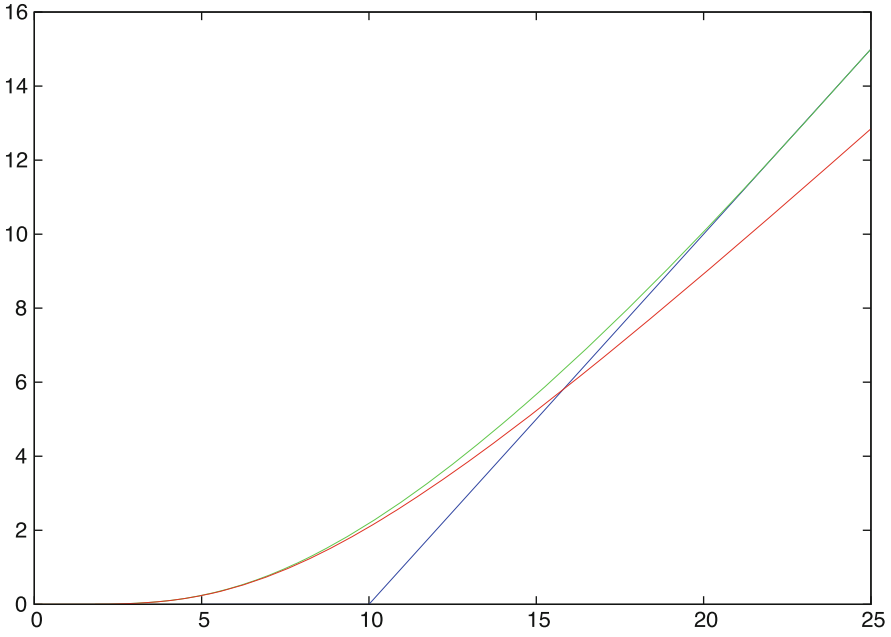

Fig. 4.9 Value $V(S, 0)$ of an American call (in *green*) with $K = 10$, $r = 0.25$, $\sigma = 0.6$, $T = 1$ and dividend flow $\delta = 0.2$. The corresponding curve of a European call in *red*; the payoff in *blue*. A special calculated value is $V(K, 0) = 2.18728$

We note in passing that the transformation $\zeta := S/S_f(t)$, $y(\zeta, t) := V(S, t)$ allows to set up a Black–Scholes-type PDE on a rectangle. In this way, the unknown front $S_f(t)$ is fixed at $\zeta = 1$, and is given implicitly by an ordinary differential equation as part of a nonlinear PDE ($\longrightarrow$ Exercise 4.8). Such a *front-fixing* approach is numerically relevant; see the Notes on Sect. 4.7.

4.5.3 Black–Scholes Inequality

The Black–Scholes equation (4.1) is valid on the continuation region (shaded areas in Fig. 4.6). For the numerical approach of the following Sect. 4.6 the computational domain will be the entire half strip $S > 0$, $0 \le t \le T$, *including* the stopping areas.[7] This will allow locating the early-exercise curve S_f. The approach requires to adapt the Black–Scholes equation in some way to the stopping areas.

To this end, define the Black–Scholes operator as

$$\mathcal{L}_{BS}(V) := \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - \delta) S \frac{\partial V}{\partial S} - rV\,.$$

With this notation the Black–Scholes equation reads

$$\frac{\partial V}{\partial t} + \mathcal{L}_{BS}(V) = 0\,.$$

What happens with this operator on the stopping regions? To this end substitute the payoff function Ψ into $\frac{\partial V}{\partial t} + \mathcal{L}_{BS}(V)$. In the case of a put, for $S \le S_f$, $V \equiv \Psi$ and

$$V = K - S\,,\quad \frac{\partial V}{\partial t} = 0\,,\quad \frac{\partial V}{\partial S} = -1\,,\quad \frac{\partial^2 V}{\partial S^2} = 0\,.$$

Hence

$$\frac{\partial V}{\partial t} + \mathcal{L}_{BS}(V) = -(r - \delta)S - r(K - S) = \delta S - rK\,.$$

Equation (4.32) implies the bound $\delta S < rK$, which leads to conclude

$$\frac{\partial V}{\partial t} + \mathcal{L}_{BS}(V) < 0\,.$$

[7] Up to localization.

That is, the Black–Scholes equation changes to an *inequality* on the stopping region. The same inequality holds for the call. (The reader may carry out the analysis for the case of a call.)

In summary, on the entire half strip $0 < S < \infty$, $0 < t < T$, American options must satisfy an inequality of the Black–Scholes type,

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r-\delta)S\frac{\partial V}{\partial S} - rV \le 0\,. \tag{4.36}$$

Both inequalities (4.30) and (4.36) hold for all (S,t). In case the strict inequality ">" holds in (4.30), equality holds in (4.36). The contact boundary S_f divides the half strip into the stopping region and the continuation region, each with appropriate version of V:

$$\begin{aligned}
\text{put:}\quad & V_\mathrm{P}^\mathrm{Am} = K - S \quad \text{for } S \le S_\mathrm{f} \quad \text{(stop)}\\
& V_\mathrm{P}^\mathrm{Am} \text{ solves (4.1)} \quad \text{for } S > S_\mathrm{f} \quad \text{(hold)}\\
\text{call:}\quad & V_\mathrm{C}^\mathrm{Am} = S - K \quad \text{for } S \ge S_\mathrm{f} \quad \text{(stop)}\\
& V_\mathrm{C}^\mathrm{Am} \text{ solves (4.1)} \quad \text{for } S < S_\mathrm{f} \quad \text{(hold)}
\end{aligned}$$

This shows that the Black–Scholes equation (4.1) must be solved also for American options, however, with special arrangements because of the free boundary. We have to look for methods that simultaneously calculate V along with the unknown S_f.

Notice that $\frac{\partial V}{\partial S}$ is continuous when S crosses S_f, but $\frac{\partial^2 V}{\partial S^2}$ and $\frac{\partial V}{\partial t}$ are not continuous. It must be expected that this lack of smoothness along the early-exercise curve S_f affects the accuracy of numerical approximations.

4.5.4 Penalty Formulation

In this subsection we outline an approach that allows for a unified treatment of stopping region and continuation region. The inequality (4.36) can be written as an equality by introducing a *penalty* term $p(V) \ge 0$, and requesting

$$\frac{\partial V}{\partial t} + \mathcal{L}_\mathrm{BS}(V) + p(V) = 0\,. \tag{4.37}$$

The penalty term p should be zero for the continuation region, and should be positive for the stopping area. When calculating an approximation V, the distance to S_f is not known, but the distance $V - \Psi$ of V to the payoff Ψ is available and serves as decisive building block of a penalty term. There are several possibilities to construct a penalty p. One classic approach will be described in Sect. 7.2. Another way to set

up a penalty can be accomplished by a term such as

$$p(V) := \frac{\epsilon}{V - \Psi} \quad \text{for a small } \epsilon > 0\,. \tag{4.38}$$

Let V^ϵ denote a solution of the penalty equation (4.37) with penalty function (4.38). Two extreme cases characterize the effect of the penalty term for (S, t) in the continuation area and in the stopping area:

- $V_\epsilon - \Psi \gg \epsilon$ implies $p \approx 0$. Then essentially the Black–Scholes equation results, and V_ϵ approximates the BS-solution.
- $0 < V_\epsilon - \Psi \ll \epsilon$ implies both $V_\epsilon \approx \Psi$ and a large value of p. The latter means that the BS-part of (4.37) is dominated by p; the BS equation is switched off.

The corresponding branches of the solution V_ϵ may be called the "continuation branch" ($p \approx 0$) and the "stopping branch" ($V_\epsilon \approx \Psi$). Obviously these two branches approximate the true solution V of the Black–Scholes problem. The intermediate range $V_\epsilon - \Psi \approx O(\epsilon)$ characterizes a boundary layer between the continuation branch and the stopping branch. In this layer around the early-exercise curve S_f the solution V_ϵ can be seen as a connection between the BS surface and the payoff plane.[8]

Notice that p and the resulting PDE are nonlinear in V, which complicates the numerical solution. The penalty formulation is advantageous especially in cases where an analysis of the early-exercise curve is difficult. See Sect. 6.7 for an exposition of the penalty approach in the two-dimensional situation. For the standard options of this chapter, we pursue another method, which effectively allows to preserve linearity.

4.5.5 *Obstacle Problem*

A brief digression into obstacle problems will motivate the procedure. We assume an "obstacle" $g(x)$, say with $g(x) > 0$ for a subinterval of $-1 < x < 1$, $g \in \mathcal{C}^2$, $g'' \le 0$ and $g(-1) < 0,\ g(1) < 0$, compare Fig. 4.10. Across the obstacle a function u with minimal length is stretched like a rubber thread. Between $x = \alpha$ and $x = \beta$ the curve u clings to the boundary of the obstacle. For α and β we encounter high-contact conditions $u(\alpha) = g(\alpha)$, $u'(\alpha) = g'(\alpha)$, and $u(\beta) = g(\beta)$, $u'(\beta) = g'(\beta)$. Initially, the two values $x = \alpha$ and $x = \beta$ are unknown. This obstacle problem is a simple free-boundary problem.

The aim is to reformulate the obstacle problem such that the free boundary conditions do not show up explicitly. This may promise computational advantages. The function u shown in Fig. 4.10 is characterized by the requirements $u \ge g$,

[8] This is illustrated in Topic 9 of the *Topics fCF*.

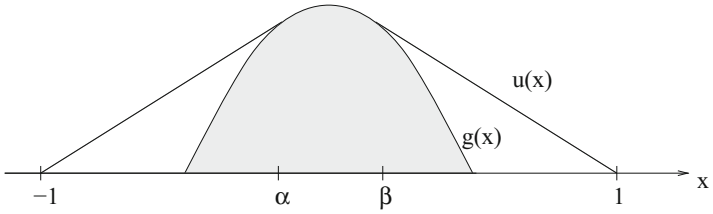

Fig. 4.10 Function $u(x)$ across an obstacle $g(x)$

$u(-1) = u(1) = 0$, $u \in \mathcal{C}^1$, and by: There is α, β such that

$$\begin{aligned} \text{for } -1 < x < \alpha: \quad & u'' = 0 \quad && (u > g) \\ \text{for } \alpha < x < \beta: \quad & u = g \quad && (u'' = g'' \le 0) \\ \text{for } \beta < x < 1: \quad & u'' = 0 \quad && (u > g)\,. \end{aligned}$$

The characterization of the two outer intervals is identical. This manifests a complementarity in the sense

$$\begin{aligned} &\text{if } u - g > 0\ , \text{ then } u'' = 0\,; \\ &\text{if } u - g = 0\ , \text{ then } u'' \le 0\,. \end{aligned}$$

In retrospect it is clear that American options are complementary in an analogous way:

$$\begin{aligned} &\text{if} \quad V - \Psi > 0, \text{ then Black–Scholes } \textit{equation} \ \frac{\partial V}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V) = 0\,; \\ &\text{if} \quad V - \Psi = 0, \text{ then Black–Scholes } \textit{inequality} \ \frac{\partial V}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V) \le 0\,. \end{aligned}$$

This analogy motivates searching for a solution of the obstacle problem. The obstacle problem can be reformulated as

$$\begin{aligned} &\text{find a function } u \text{ such that} \\ &u''(u-g) = 0\,, \quad -u'' \ge 0\,, \quad u - g \ge 0\,, \\ &u(-1) = u(1) = 0\,, \quad u \in \mathcal{C}^1[-1,1]\,. \end{aligned} \tag{4.39}$$

The key line (4.39) is a **linear complementarity problem** (LCP). This formulation does not mention the free boundary conditions at $x = \alpha$ and $x = \beta$ explicitly. This will be advantageous because α and β are unknown. After a solution to (4.39) is calculated, the values α and β are read off from the solution. To this end we

construct a numerical solution procedure for the complementarity version (4.39) of the obstacle problem.

Discretization of the Obstacle Problem
A finite-difference approximation for u'' on the grid $x_i = -1 + i\Delta x$, with $\Delta x = \frac{2}{m}$, $g_i := g(x_i)$ leads to

$$(w_{i-1} - 2w_i + w_{i+1})(w_i - g_i) = 0,$$
$$-w_{i-1} + 2w_i - w_{i+1} \geq 0, \quad w_i \geq g_i$$

for $0 < i < m$ and $w_0 = w_m = 0$. The w_i are approximations to $u(x_i)$. In view of the signs of the factors in the first line in this discretization scheme it can be written using a scalar product. To see this, define a vector notation using

$$G := \begin{pmatrix} 2 & -1 & & 0 \\ -1 & \ddots & \ddots & \\ & \ddots & \ddots & -1 \\ 0 & & -1 & 2 \end{pmatrix} \text{ and } w := \begin{pmatrix} w_1 \\ \vdots \\ w_{m-1} \end{pmatrix}, \; g := \begin{pmatrix} g_1 \\ \vdots \\ g_{m-1} \end{pmatrix}.$$

Then the discretized complementarity problem is rewritten in the form

$$(w - g)^{tr} Gw = 0\,, \quad Gw \geq 0\,, \quad w \geq g\,. \tag{4.40}$$

To calculate a solution of (4.40) one solves $Gw = 0$ under the side condition $w \geq g$. This will be explained in Sect. 4.6.4. In Sect. 5.3 we will return to the obstacle problem with a version as variational problem.

4.5.6 *Linear Complementarity for American Put Options*

In analogy to the simple obstacle problem described above we now derive a linear complementarity problem for American options. Here we confine ourselves to American puts without dividends ($\delta = 0$); the general case will be listed in Sect. 4.6. The transformations (4.3) lead to

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2} \text{ as long as } V_{\mathrm{P}}^{\mathrm{Am}} > (K - S)^+\,.$$

Also the side condition (4.30) is transformed: The relation

$$V_{\mathrm{P}}^{\mathrm{Am}}(S,t) \geq (K-S)^{+} = K \max\{\,1-\mathrm{e}^{x},\,0\,\}$$

leads to the inequality

$$\begin{aligned} y(x,\tau) &\geq \exp\{\frac{1}{2}(q-1)x + \frac{1}{4}(q+1)^2\tau\}\max\{1-\mathrm{e}^{x},\,0\} \\ &= \exp\{\frac{1}{4}(q+1)^2\tau\}\max\{(1-\mathrm{e}^{x})\mathrm{e}^{\frac{1}{2}(q-1)x},\,0\} \\ &= \exp\{\frac{1}{4}(q+1)^2\tau\}\max\{\mathrm{e}^{\frac{1}{2}(q-1)x} - \mathrm{e}^{\frac{1}{2}(q+1)x},\,0\} \\ &=: g(x,\tau)\,. \end{aligned}$$

This function g allows to write the initial condition (4.5)/(4.6) as $y(x,0) = g(x,0)$. In summary, we require $y_\tau = y_{xx}$ as well as

$$y(x,0) = g(x,0) \;\text{ and }\; y(x,\tau) \geq g(x,\tau)\,,$$

and, in addition, boundary conditions, and $y \in \mathcal{C}^1$ with respect to x. For $x \to \infty$ the function g vanishes, $g(x,\tau) = 0$, so the boundary condition $y(x,\tau) \to 0$ for $x \to \infty$ can be written

$$y(x,\tau) = g(x,\tau) \;\text{ for }\; x \to \infty\,.$$

The same holds for $x \to -\infty$ ($\longrightarrow$ Exercise 4.9). In the localizing practice, the boundary conditions are formulated for $x_{\min}$ and $x_{\max}$. Collecting all expressions, the American put is formulated as linear complementarity problem:

$$\begin{aligned} &\left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2}\right)(y-g) = 0\,, \\ &\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0\,, \qquad y - g \geq 0\,, \\ &y(x,0) = g(x,0), \quad y(x_{\min},\tau) = g(x_{\min},\tau)\,, \\ &y(x_{\max},\tau) = g(x_{\max},\tau)\,,\; y \in \mathcal{C}^1 \text{ with respect to } x\,. \end{aligned}$$

The exercise boundary is automatically captured by this formulation. An analogous formulation holds for the American call. Both of the formulations are comprised by Problem 4.7 below.

4.6 Computation of American Options

In the previous sections we have derived a linear complimentarity problem for both put and call of an American-style option. We summarize the results into Problem 4.7. This assumes for a put $r > 0$, and for a call $\delta > 0$; otherwise the American option is not distinct from the European counterpart.

Problem 4.7 (Linear Complementarity Problem)

Notations of (4.3), including

$$q = \frac{2r}{\sigma^2}\,, \quad q_\delta = \frac{2(r-\delta)}{\sigma^2}\,,$$

$$put: \; g(x,\tau) := \exp\{\frac{\tau}{4}((q_\delta - 1)^2 + 4q)\} \max\{e^{\frac{x}{2}(q_\delta - 1)} - e^{\frac{x}{2}(q_\delta + 1)},\, 0\}$$

$$call: \; g(x,\tau) := \exp\{\frac{\tau}{4}((q_\delta - 1)^2 + 4q)\} \max\{e^{\frac{x}{2}(q_\delta + 1)} - e^{\frac{x}{2}(q_\delta - 1)},\, 0\}$$

$$\left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2}\right)(y - g) = 0$$

$$\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0\,, \quad y - g \geq 0$$

$$x_{\min} \leq x \leq x_{\max}\,, \quad 0 \leq \tau \leq \frac{1}{2}\sigma^2 T$$

$$y(x,0) = g(x,0)$$

$$y(x_{\min},\tau) = g(x_{\min},\tau)\,, \quad y(x_{\max},\tau) = g(x_{\max},\tau)$$

As outlined in Sect. 4.5, the free boundary problem of American options is described in Problem 4.7 such that the free boundary condition does not show up explicitly. We now enter the discussion of how to solve Problem 4.7 numerically.

4.6.1 Discretization with Finite Differences

We use the same grid as in Sect. 4.2.2, with $w_{i,\nu}$ denoting an approximation to $y(x_i, \tau_\nu)$, and $g_{i,\nu} := g(x_i, \tau_\nu)$ for $0 \leq i \leq m$, $0 \leq \nu \leq \nu_{\max}$. The backward difference, the explicit, and the Crank–Nicolson method can be combined into one formula,

$$\frac{w_{i,\nu+1} - w_{i,\nu}}{\Delta\tau} = \theta \frac{w_{i+1,\nu+1} - 2w_{i,\nu+1} + w_{i-1,\nu+1}}{\Delta x^2} + (1-\theta)\frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2}\,,$$

with the choices $\theta = 0$ (explicit), $\theta = \frac{1}{2}$ (Crank–Nicolson), $\theta = 1$ (backward-difference method). This family of numerical schemes parameterized by θ is often called *θ-method*.

The differential inequality $\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0$ becomes the discrete version

$$\begin{aligned} & w_{i,\nu+1} - \lambda\theta(w_{i+1,\nu+1} - 2w_{i,\nu+1} + w_{i-1,\nu+1}) \\ & \quad - w_{i,\nu} - \lambda(1-\theta)(w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}) \geq 0\,, \end{aligned} \tag{4.41}$$

again with the abbreviation $\lambda := \frac{\Delta\tau}{\Delta x^2}$. With the notations

$$\begin{aligned} & b_{i,\nu} := w_{i,\nu} + \lambda(1-\theta)(w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu})\,,\ i = 2, \ldots, m-2 \\ & b_{1,\nu} \text{ and } b_{m-1,\nu} \text{ incorporate the boundary conditions} \\ & b^{(\nu)} := (b_{1,\nu}, \ldots, b_{m-1,\nu})^{tr} \\ & w^{(\nu)} := (w_{1,\nu}, \ldots, w_{m-1,\nu})^{tr} \\ & g^{(\nu)} := (g_{1,\nu}, \ldots, g_{m-1,\nu})^{tr} \end{aligned}$$

and

$$A := \begin{pmatrix} 1+2\lambda\theta & -\lambda\theta & & 0 \\ -\lambda\theta & \ddots & \ddots & \\ & \ddots & \ddots & \ddots \\ 0 & & \ddots & \ddots \end{pmatrix} \in \mathbb{R}^{(m-1)\times(m-1)} \tag{4.42}$$

(4.41) is rewritten in vector form as

$$Aw^{(\nu+1)} \geq b^{(\nu)} \quad \text{for all } \nu\,.$$

Such inequalities for vectors are understood componentwise. The inequality $y - g \geq 0$ leads to

$$w^{(\nu)} \geq g^{(\nu)}\,,$$

and $\left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2}\right)(y-g) = 0$ becomes

$$\left(Aw^{(\nu+1)} - b^{(\nu)}\right)^{tr}\left(w^{(\nu+1)} - g^{(\nu+1)}\right) = 0\,.$$

The initial and boundary conditions are

$$\begin{aligned} & w_{i,0} = g_{i,0}\,, \quad i = 1, \ldots, m-1\,, \quad (w^{(0)} = g^{(0)})\,; \\ & w_{0,\nu} = g_{0,\nu}\,, \quad w_{m,\nu} = g_{m,\nu}\,, \quad \nu \geq 1\,. \end{aligned}$$

Boundary conditions are realized in the vectors $b^{(\nu)}$ as follows:

$$\begin{aligned} &b_{2,\nu}, \ldots, b_{m-2,\nu} \quad \text{as defined above,} \\ &b_{1,\nu} = w_{1,\nu} + \lambda(1-\theta)(w_{2,\nu} - 2w_{1,\nu} + g_{0,\nu}) + \lambda\theta g_{0,\nu+1} \\ &b_{m-1,\nu} = w_{m-1,\nu} + \lambda(1-\theta)(g_{m,\nu} - 2w_{m-1,\nu} + w_{m-2,\nu}) + \lambda\theta g_{m,\nu+1} \end{aligned} \tag{4.43}$$

We summarize the discrete version of the Problem 4.7 into an Algorithm:

Algorithm 4.8 (Computation of American Options)

For $\nu = 0, 1, \ldots, \nu_{\max} - 1$:

Calculate the vectors $g := g^{(\nu+1)}$,

$b := b^{(\nu)}$ *from* (4.42), (4.43).

Calculate the vector w *as solution of the problem*

$$Aw - b \geq 0\,, \quad w \geq g\,, \quad (Aw-b)^{tr}(w-g) = 0\,. \tag{4.44}$$

$w^{(\nu+1)} := w$

This completes the chosen finite-difference discretization.

The remaining problem is to solve the complementarity problem in matrix-vector form (4.44). In principle, how to solve (4.44) is a new topic independent of the discretization background. But accuracy and efficiency will depend on the context of selected methods. We pause for a moment to become aware how broad the range of possible finite-difference methods is.

There are possible sources of inaccuracies. The payoff is not smooth. And recall from Sect. 4.5.3 that $V(S,t)$ is not C^2-smooth over the free boundary S_f. Second-order convergence of the basic Crank–Nicolson scheme must be expected to be deteriorated. The effect caused by lacking smoothness depends on the choice of several items, namely, the

(1) kind of transformation/PDE [from no transformation over a mere $\tau := T - t$ to the transformation (4.3)],
(2) kind of discretization (from backward-difference over Crank–Nicolson to more refined schemes like BDF2),
(3) method of solution for (4.44).

The latter can be a direct elimination method, or an iteratively working indirect method. Large systems as they occur in PDE context are frequently solved iteratively, in particular in high-dimensional spaces. Such approaches sometimes benefit from smoothing properties. Both an iterative procedure (following [376]) and a direct approach (following [52]) will be discussed below. It turns out that in the one-dimensional scenario of this chapter (one underlying asset), the direct approach is faster.

4.6.2 Reformulation and Analysis of the LCP

In each time level ν in Algorithm 4.8, a linear complementarity problem (4.44) must be solved. This is the bulk of work in Algorithm 4.8. Before entering a numerical solution, we analyze the LCP. Since this subsection is general numerical analysis independent of the finance framework, we momentarily use vectors x, y, r freely in other context.[9] For the analysis transform problem (4.44) from the w-world into an x-world with

$$\begin{aligned} x &:= w - g\,, \\ y &:= Aw - b\,. \end{aligned} \tag{4.45}$$

Then it is easy to see (the reader may check) that the task of calculating a solution w for (4.44) is equivalent to the following problem:

Problem 4.9 (Cryer) *Compute vectors x and y such that for $\hat{b} := b - Ag$*

$$Ax - y = \hat{b}\,, \quad x \geq 0\,, \quad y \geq 0\,, \quad x^{tr} y = 0\,. \tag{4.46}$$

First we make sure that the above problem has a unique solution. To this end one shows the equivalence of Problem 4.9 with a minimization problem.

Lemma 4.10 *Problem 4.9 is equivalent to the minimization problem*

$$\min_{x \geq 0} G(x), \quad \text{with } G(x) := \frac{1}{2}(x^{tr} Ax) - \hat{b}^{tr} x\,, \tag{4.47}$$

where G is strictly convex.

Proof The derivatives of G are $G_x = Ax - \hat{b}$ and $G_{xx} = A$. Lemma 4.3 implies that A has positive eigenvalues. Hence the Hessian matrix G_{xx} is symmetric and positive definite. So G is strictly convex, and has a unique minimum on each convex set in $\mathbb{R}^n$, for example on $x \geq 0$. The Theorem of Karush, Kuhn and Tucker minimizes G under $H_i(x) \leq 0$, $i = 1, \ldots, m$. According to this theorem,[10] a vector x_0 to be a minimum is equivalent to the existence of a Lagrange multiplier $y \geq 0$ with

$$\operatorname{grad} G(x_0) + \left(\frac{\partial H(x_0)}{\partial x} \right)^{tr} y = 0\,, \quad y^{tr} H(x_0) = 0\,.$$

[9] Notation: In this Sect. 4.6.2, x does not have the meaning of transformation (4.3), and r not that of an interest rate, and y is no PDE solution. Here, $x, y \in \mathbb{R}^{m-1}$.

[10] For the KKT (Karush-Kuhn-Tucker or Kuhn-Tucker) theory we refer to [348, 350]. In our context, $m - 1$.

The set $x \geq 0$ leads to define $H(x) := -x$. Hence the KKT condition is $Ax - \hat{b} + (-I)^{tr} y = 0$, $y^{tr} x = 0$, and we have reached Eq. (4.46). □

4.6.3 Iterative Procedure for the LCP

An iterative procedure can be derived from the minimization problem stated in Lemma 4.10. This algorithm is based on the SOR[11] method [92]. Note that (4.44) is not in the easy form of equation $Ax = b$ discussed in Appendix C.2; a modification of the standard SOR will be necessary. The iteration of the SOR method for $Ax = \hat{b} = b - Ag$ is written componentwise (⟶ Exercise 4.10) as iteration for the correction vector $x^{(k)} - x^{(k-1)}$:

$$r_i^{(k)} := \hat{b}_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - a_{ii} x_i^{(k-1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k-1)} , \tag{4.48}$$

$$x_i^{(k)} = x_i^{(k-1)} + \omega_{\mathrm{R}} \frac{r_i^{(k)}}{a_{ii}} . \tag{4.49}$$

Here k denotes the number of the iteration, $n = m-1$, and a_{ij} is element of the matrix A. In the cases $i = 1$, $i = m - 1$ one of the sums in (4.48) is empty. The *relaxation parameter* ω_{R} is a factor chosen in a way that should improve the convergence of the iteration. The "projected" SOR method for solving (4.46) starts from a vector $x^{(0)} \geq 0$ and is identical to the SOR method up to a modification on (4.49) serving for $x_i^{(k)} \geq 0$.

Algorithm 4.11 (PSOR, Projected SOR for Problem 4.9)

$$\begin{aligned}
&\textit{outer loop}: \ k = 1, 2, \ldots \\
&\textit{inner loop}: \ i = 1, \ldots, m-1 \\
&\qquad r_i^{(k)} \ \textit{as in (4.48)}, \\
&\qquad x_i^{(k)} = \max \left\{ 0, \ x_i^{(k-1)} + \omega_{\mathrm{R}} \frac{r_i^{(k)}}{a_{ii}} \right\} , \\
&\qquad y_i^{(k)} = -r_i^{(k)} + a_{ii} \left(x_i^{(k)} - x_i^{(k-1)} \right) .
\end{aligned}$$

[11] Successive overrelaxation, SOR. For an introduction to classic iterative methods for the solution of systems of linear equations $Ax = b$ we refer to Appendix C.2.

This algorithm solves $Ax = \hat{b}$ for $\hat{b} = b - Ag$ iteratively by *componentwise* considering $x^{(k)} \geq 0$. The vector y or the components $y_i^{(k)}$ converging to y_i, are not used explicitly for the algorithm. But since $y \geq 0$ is shown ($Aw \geq b$), the vector y plays an important role in the proof of convergence. Transformed back into the w-world of problem (4.44) by means of (4.45), the Algorithm 4.11 solves (4.44).

A proof of the convergence of Algorithm 4.11 is based on Lemma 4.10. One shows that the sequence defined in Algorithm 4.11 minimizes G. The main steps of the argumentation are sketched as follows:

For $0 < \omega_R < 2$ the sequence $G(x^{(k)})$ is decreasing monotonically;
Show $x^{(k+1)} - x^{(k)} \to 0$ for $k \to \infty$;
The limit exists because $x^{(k)}$ moves in a compact set $\{x \mid G(x) \leq G(x^{(0)})\}$;
The vector r from (4.48) converges toward $-y$;
Assuming $r \geq 0$ and $r^T x \neq 0$ leads to a contradiction to $x^{(k+1)} - x^{(k)} \to 0$. (For the proof see [92].)

4.6.4 Direct Method for the LCP

Another formulation has shown to be a basis for a direct solution by elimination:

Problem 4.12 (Cryer's Problem Restated)

Solve $Aw = b$ componentwise such that
the side condition $w \geq g$ is obeyed.

An implementation must be done carefully such that the boundary conditions and all the LCP requirements in (4.46) are met. The structure of Problem 4.12 is different from the system $Aw = b$ without side condition [201].

Recall that a direct method to solve a system $Aw = b$ of linear equations establishes in a first phase an equivalent system $\tilde{A}w = \tilde{b}$ with a triangular matrix $\tilde{A}$ (here bidiagonal since A is tridiagonal). After $\tilde{A}, \tilde{b}$ are calculated, the second phase (the solution of $\tilde{A}w = \tilde{b}$) is established in a single loop. For an upper (right) triangular matrix $\tilde{A}$ the fist phase is a forward loop, and the subsequent second phase is backward. This is the familiar form of Gauss elimination, which in this context may be called "forward-backward method."

Less familiar is the opposite procedure: In order to establish $\tilde{A}$ as *lower* triangular matrix, the first phase creates zeroes above the diagonal, and hence is done in the backward fashion. The second phase then solves $\tilde{A}w = \tilde{b}$ in a forward loop. This is the backward-forward version of Gauss elimination. In our context of solving Problem 4.12 both versions are not equivalent; renumbering the equations and variables does not help. This is caused by the side condition $w \geq g$, which adds a nonlinearity, with different conditions on w_1 and w_m.

When in the second phase in the ith step of the solution loop $\tilde{w}_i$ is a component of the solution of $\tilde{A}w = \tilde{b}$, then $w_i := \max\{\tilde{w}_i, g_i\}$ might appear the correct value.

But w depends on the orientation of the loop, backward or forward. Only one direction works. An implementation must make sure that the characteristic order of the underlying option is preserved. For a **put** this means:

We denote i_f the index of the node S_i, which is closest to the contact point.[12] The index i_f marks the location of the free boundary. More precisely,
$w_i = g_i$ for $1 \le i \le i_\mathrm{f}$, and
$w_i > g_i$ for $i_\mathrm{f} < i \le m$.

This structure is characteristic for a put, but the index i_f is unknown. For a put the start is $w_1 = g_1$, and the $w_i := \max\{\tilde{w}_i, g_i\}$-loop must be *forward*. Accordingly, for a put, $\tilde{A}$ must be a lower triangular matrix, and hence the backward-forward variant of Gauss elimination is applied. This amounts to an RL-decomposition of A ($\longrightarrow$ Appendix C.1). The lower triangle $\tilde{A} := L$ is established, and the vector $\tilde{b}$ obtained by solving $R\tilde{b} = b$.

Algorithm 4.13 (American Put)

first phase:
Calculate the RL-decomposition of A.
Then set $\tilde{A} = L$ *and calculate* $\tilde{b}$ *from* $R\tilde{b} = b$ *(backward loop).*
second phase: forward loop for growing i:
Start with $i = 1$. *Calculate the next component of* $\tilde{A}w = \tilde{b}$; *denote it* $\tilde{w}_i$.
Set $w_i := \max\{\tilde{w}_i, g_i\}$.

This procedure was suggested by Brennan and Schwartz [52]. Since the matrix A from (4.42) is tridiagonal, the costs are low. In this way, a direct method for solving Problem 4.12 is established, which is as efficient as solving a standard system of linear equations. ($\longrightarrow$ Exercise 4.11) The elegant approach of Algorithm 4.13 allows to treat the nonlinear problem of valuing an American option as if it were linear.

For a call $w_i = g_i$ holds for large indices i, and the elimination phase runs in a backward loop. This requires the traditional upper triangular matrix $\tilde{A}$ as calculated by the LR-decomposition ($\longrightarrow$ Exercise 4.12). For both put and call there is only one index i_f separating the components with $w_i = g_i$ from those with $w_i > g_i$.

4.6.5 An Algorithm for Calculating American Options

We return to the original meaning of the variables x, y, r, as used for instance in (4.2), (4.3). It remains to substitute a proper algorithm for solving (4.44) in Algorithm 4.8. From the analysis of Sect. 4.6.2, we either apply the iterative Algorithm 4.11 ($\longrightarrow$ Exercise 4.13), or implement the fast direct method of

[12] The S-interval must be large enough, $S_1 < S_\mathrm{f}$.

Algorithm 4.13. The resulting algorithm is formulated in Algorithm 4.14 with an LCP-solving module. The implementation of the direct version is left to the reader (⟶ Exercise 4.11). Recall $g_{i,\nu} := g(x_i, \tau_\nu)$ $(0 \le i \le m)$ and $g^{(\nu)} := (g_{1,\nu}, \ldots, g_{m-1,\nu})^{tr}$. Figure 4.11 depicts a result of Algorithm 4.14 for Example 1.6. Here we obtain the contact point with value $S_f(0) = 36.16$ (with $m = \nu_{\max} = 1600$). Figure 4.13 shows the American put that corresponds to the call in Fig. 4.9.

Algorithm 4.14 (Prototype Core Algorithm)

Set up the function $g(x, \tau)$ listed in Problem 4.7.

Choose θ ($\theta = 1/2$ for Crank–Nicolson).

For PSOR: choose $1 \le \omega_R < 2$ (for example, $\omega_R = 1$),

fix an error bound ε (for example, $\varepsilon = 10^{-5}$).

Fix the discretization by choosing $x_{\min}$, $x_{\max}$, m, $\nu_{\max}$

(for example, $x_{\min} = -5$, $x_{\max} = 5$ or 3, $\nu_{\max} = m = 100$).

Calculate $\Delta x := (x_{\max} - x_{\min})/m$,

$$\Delta\tau := \frac{1}{2}\sigma^2 T/\nu_{\max}\,,$$

$$x_i := x_{\min} + i\Delta x \text{ for } i = 0, \ldots, m\,.$$

Initialize the iteration vector w with

$$g^{(0)} = (g(x_1, 0), \ldots, g(x_{m-1}, 0)).$$

Calculate $\lambda := \Delta\tau/\Delta x^2$ and $\alpha := \lambda\theta$.

(Now all elements of matrix A from (4.42) are defined.)

τ *– loop :* *for* $\nu = 0, 1, \ldots, \nu_{\max} - 1$:

$$\tau_\nu := \nu\Delta\tau$$

$$b_i := w_i + \lambda(1-\theta)(w_{i+1} - 2w_i + w_{i-1}) \text{ for } 2 \le i \le m-2$$

$$b_1 := w_1 + \lambda(1-\theta)(w_2 - 2w_1 + g_{0,\nu}) + \alpha g_{0,\nu+1}$$

$$b_{m-1} := w_{m-1} + \lambda(1-\theta)(g_{m,\nu} - 2w_{m-1} + w_{m-2}) + \alpha g_{m,\nu+1}$$

Module: Calculate the LCP solution w of Problem 4.12,

preferably by direct elimination as Algorithm 4.13, Exercise 4.11, or

alternatively by implementing an iterative method as Algorithm 4.11.

$$w^{(\nu+1)} = w\,.$$

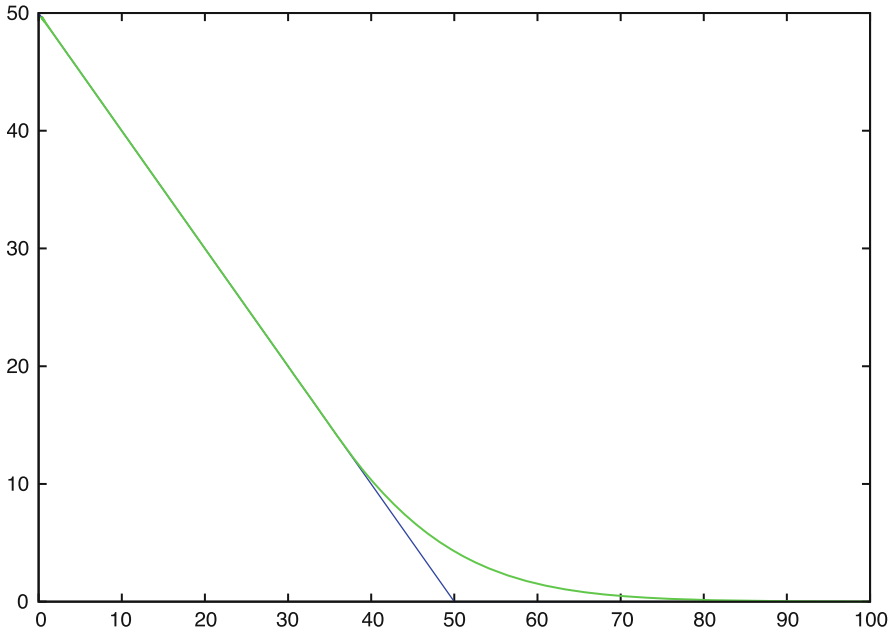

Fig. 4.11 Example 1.6: American put, $K = 50$, $r = 0.1$, $\sigma = 0.4$, $T = \frac{5}{12}$. $V(S, 0)$ (*green curve*) and payoff $V(S, T)$ (*blue*). Special calculated value: $V(K, 0) = 4.2842$

4.6.5.1 Valuing Options

For completeness we mention that it is possible to calculate European options with Algorithm 4.14 after simple modifications, which recover standard methods for solving $Aw = b$ (without $w \geq g$). If in addition the boundary conditions are adapted, then the computer program resulting from Algorithm 4.14 can be applied to European options. Of course, applying the analytic solution formula (A.15) or (A.17) should be most economical, when the entire surface $V(S, t)$ is not required. But for the purpose of testing Algorithm 4.14 it is recommendable to compare its results to something "known."

Back to American options, we complete the analysis, summarizing how a concrete financial task is solved with the core Algorithm 4.14, which is formulated in artificial variables such as $x_i, g_{i,\nu}, w_i$ and not in financial variables. This requires an interface between the real world and the core algorithm. The interface is provided by the transformations in (4.3). This important ingredient must be included for completeness. Let us formulate the required transition between the real world and

the numerical machinery of Algorithm 4.14 as another algorithm:

Algorithm 4.15 (American Options)

Input: strike K, *time to expiration* T, *spot price* S_0, r, δ, σ .

Perform the core Algorithm 4.14.

(*The* τ*-loop ends at* $\tau_{\text{end}} = \frac{1}{2}\sigma^2 T$.)

For $i = 1, \ldots, m-1$:

w_i *approximates* $y(x_i, \frac{1}{2}\sigma^2 T)$,

$S_i = K \exp\{x_i\}$

$V(S_i, 0) = K w_i \exp\{-\frac{x_i}{2}(q_\delta - 1)\} \exp\{-\tau_{\text{end}}(\frac{1}{4}(q_\delta - 1)^2 + q)\}$

Test for early exercise: Approximate $S_{\text{f}}(0)$ *and compare to* S_0 .

For the direct method, an approximation for $S_{\text{f}}(0)$ is readily available via $S_{i_{\text{f}}}$. An indirect method checks the closeness of V to the payoff:

$$\text{Choose a small } \varepsilon^* > 0, \text{ for example, } \varepsilon^* = K \cdot 10^{-5}.$$

$$i_{\text{f}} := \max\{\, i \mid \; |V(S_i, 0) + S_i - K| < \varepsilon^* \,\} \text{ for a put,}$$

$$i_{\text{f}} := \min\{\, i \mid \; |K - S_i + V(S_i, 0)| < \varepsilon^* \,\} \text{ for a call.}$$

Criterion $S_0 < S_{i_{\text{f}}}$ indicates the stopping region for a put; for a call, this indication is $S_0 > S_{i_{\text{f}}}$.

Algorithm 4.15 evaluates the data at the final time level τ_{end}, which corresponds to $t = 0$. The computed information for the intermediate time levels can be evaluated analogously. In this way, the locations of $S_{i_{\text{f}}}$ can be put together to form an approximation of the free-boundary or stopping-time curve $S_{\text{f}}(t)$. But note that this approximation will be a crude step function. It requires some effort to calculate the curve $S_{\text{f}}(t)$ with reasonable accuracy, see the illustration of curve c_1 in Fig. 4.8 ($\longrightarrow$ Exercise 4.14).

4.6.5.2 Modifications

The above Algorithm 4.14 (along with Algorithm 4.15) is the prototype of a finite-difference algorithm. Improvements are possible. For example, the equidistant time step $\Delta\tau$ can be given up in favor of a variable time stepping. A few very small time steps initially will help to quickly damp the influence of the nonsmooth payoff. The effect of the kink of the payoff at the strike K is illustrated by Fig. 4.12. The turmoil at the corner is seen, but also the relatively rapid smoothing within a few

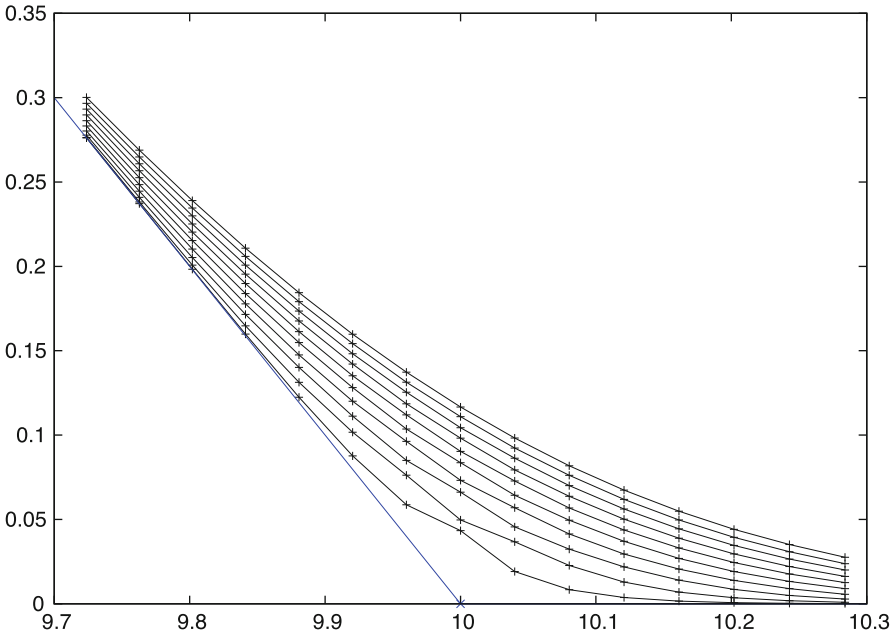

Fig. 4.12 Finite differences, Crank–Nicolson; American put with $r = 0.06$, $\sigma = 0.3$, $T = 1$, $K = 10$; $M = 1000$, $x_{\min} = -2$, $x_{\max} = 2$, $\Delta x = 1/250$, $\Delta t = 1/1000$, payoff (in *blue*) and $V(S, t_\nu)$ for $t_\nu = 1 - \nu\Delta t$, $\nu = 1, \ldots, 10$

time steps. Figure 4.12 shows explicitly the dependence of V on S; implicit in the figure is the dependence on t with corresponding oscillations. The effect of the lack of smoothness is heavier in case the payoff is discontinuous (binary option). In such a context it is advisable to start with a few fully implicit backward time steps ($\theta = 1$) before switching to Crank–Nicolson ($\theta = 1/2$). Such a procedure is called Rannacher stepping, see [305, 310], and the Notes on Sect. 4.3. After one run of the algorithm it is advisable to refine the initial grid to have a possibility to control the error. This simple strategy will be discussed in some more detail in Sect. 4.7.

Practical experience with boundary conditions (4.27) suggests working with $S_{\min} = 0.05$ and $S_{\max} = 5K$. For the transformation (4.3) $S = Ke^x$ this amounts to $x_{\min} = -3 - \log K$, $x_{\max} = 1.6$. This is to be modified for other transformations, see for instance the choice in Fig. 7.4.

4.6.5.3 Sensitivities

The greeks delta, gamma, theta are easily obtained by difference quotients. These approximations are formed by the V-values that were calculated on the finite-difference grid. For vega and rho, a recalculation is necessary, see Sect. 1.4.6.

In general, the comparably expensive solving of appropriate PDEs will not be necessary (⟶ Exercise 4.16).

4.7 On the Accuracy

Necessarily, each result obtained with the means of this chapter is subjected to errors in several ways. The most important errors have been mentioned earlier; in this section we collect them. Let us emphasize again that in general the *existence* of errors must be accepted, but not their magnitude. By investing sufficient effort, many of the errors can be kept at a tolerable level.

(a) **modeling error**
The assumptions defining the underlying financial model are restrictive. The Assumptions 1.2, for example, will not exactly match the reality of a financial market. Similarly this holds for other models. And the parameters of the models (such as volatility σ) are unknown and must be estimated. Hence the equations of the model are only crude approximations of "reality."

(b) **discretization errors**
Under the heading "discretization error" we summarize several errors that are introduced when the continuous PDE is replaced by a set of approximating equations defined on a grid. An essential portion of the discretization error is the difference between differential quotients and difference quotients. For example, a Crank–Nicolson discretization error is of the order $O(\Delta^2)$, if Δ is a measure of the grid size, and if the solution function is sufficiently smooth. Other discretization errors include the localization error caused by truncating the infinite interval $-\infty < x < \infty$ to a finite interval, or the implementation of the boundary conditions, or a quantification error when the strike ($x = 0$) is not part of the grid. In passing we recommend that the strike be one of the grid points, $x_k = 0$ for one k.

(c) **error from solving the linear equation**
An iterative solution of the linear systems of equation $Aw = b$ means that the error approaches 0 when $k \to \infty$, where k counts the number of iterations. By practical reasons the iteration must be terminated at a finite $k_{\max}$ such that the effort is bounded. Hence an error remains from the linear equations. The error tends to be neglectable for direct elimination methods.

(d) **rounding error**
The finite number of digits l of the mantissa is the reason for rounding errors.

In general, one has no *accurate* information on the size of these errors. Typically, the modeling errors are much larger than the discretization errors. In practice, in view of the uncertainties of modeling, it would be questionable to strive for an extremely small discretization error. For a stable method, the rounding errors are the least problem. The numerical analyst, as a rule, has limited potential in manipulating the modeling error. So the numerical analyst concentrates on the other

errors, especially on discretization errors. To this end we may use the qualitative assertion of Theorem 4.4. But such an a priori result is only a basic step toward our ultimate goal formulated in Problem 4.16.

4.7.1 Elementary Error Control

Here we neglect modeling errors and try to solve the a posteriori error problem:

Problem 4.16 (Principle of an Error Control) *Let the exact result of a solution of the continuous equations be denoted η^*. The approximation η calculated by a given algorithm depends on a representative grid size Δ, on $k_{\max}$, on the word length l of the computer, and maybe on several additional parameters, symbolically written*

$$\eta = \eta(\Delta, k_{\max}, l)\,.$$

Choose $\Delta, k_{\max}, l$ such that the absolute error of η does not exceed a prescribed error tolerance ϵ,

$$|\eta - \eta^*| < \epsilon\,.$$

This problem is difficult to solve, because we implicitly assume an *efficient* approximation avoiding an overkill with extremely small values of Δ or large values of $k_{\max}$ or l. Time counts in real-time application. So we try to avoid unnecessary effort of achieving a tiny error $|\eta - \eta^*| \ll \epsilon$. The exact size of the error is unknown. But its order of magnitude can be estimated as follows.

Let us assume the method is of order p. We simplify this statement to

$$\eta(\Delta) - \eta^* = \gamma\Delta^p\,. \tag{4.50}$$

Here γ is a priori unknown. By calculating two approximations, say for grid sizes Δ_1 and Δ_2, the constant γ can be calculated. To this end subtract the two calculated approximations η_1 and η_2,

$$\eta_1 := \eta(\Delta_1) = \gamma\Delta_1^p + \eta^*$$
$$\eta_2 := \eta(\Delta_2) = \gamma\Delta_2^p + \eta^*$$

to obtain

$$\gamma = \frac{\eta_1 - \eta_2}{\Delta_1^p - \Delta_2^p}\,.$$

A simple choice of the grid size Δ_2 for the second approximation is the refinement $\Delta_2 = \frac{1}{2}\Delta_1$. This leads to

$$\gamma \left(\frac{\Delta_1}{2}\right)^p = \frac{\eta_1 - \eta_2}{2^p - 1}. \tag{4.51}$$

Especially for $p = 2$ the relation

$$\gamma \Delta_1^2 = \frac{4}{3}(\eta_1 - \eta_2)$$

results. In view of the scenario (4.50) the absolute error of the approximation η_1 is given by

$$\frac{4}{3}|\eta_1 - \eta_2|$$

and the error of η_2 by (4.51).

The above procedure does not guarantee that the error η is bounded by ϵ. This flaw is explained by the simplification in (4.50), and by neglecting the other type of errors of the above list (b)–(c). Here we have assumed γ constant, which in reality depends on the parameters of the model, for example, on the volatility σ. But testing the above rule of thumb (4.50)/(4.51) on European options shows that it works reasonably well. Here we compare the finite-difference results to the analytic solution formulas (A.15)/(A.17), the numerical errors of which are comparatively negligible. The procedure works similar well for American options, although then the function $V(S,t)$ is not C^2-smooth at $S_f(t)$. (The effect of the lack in smoothness is similar as in Fig. 4.12.) In practical applications of Crank–Nicolson's method one can observe quite well that doubling of m and $\nu_{\max}$ decreases the absolute error approximately by a factor of four. To obtain a minimum of information on the error, the core Algorithm 4.14 should be applied at least for two grids following the lines outlined above. The information on the error can be used to match the grid size Δ to the desired accuracy.

Let us illustrate the above considerations with an example, compare Figs. 4.13 and 4.14, and Table 4.1. For an American put and $x_{\max} = -x_{\min} = 5$ we calculate several approximations, and test Eq. (4.50) in the form $\eta(\Delta) = \eta^* + \gamma\Delta^2$. We illustrate the approximations as points in the (Δ^2, η)-plane. The better the assumption (4.50) is satisfied, the closer the calculated points lie on a straight line. Figure 4.14 suggests that this error-control model can be expected to work well.

In order to check the error quality of a computer program on standard American options, one may check the put-call symmetry relation (A.23). For example, for the parameters of Fig. 4.13/Table 4.1, the corresponding call with $S = K$ and switched parameters $r = 0.2$, $\delta = 0.25$ is calculated, and the results match very well: For the finest discretization in Table 4.1, about 8 digits match with the value of the

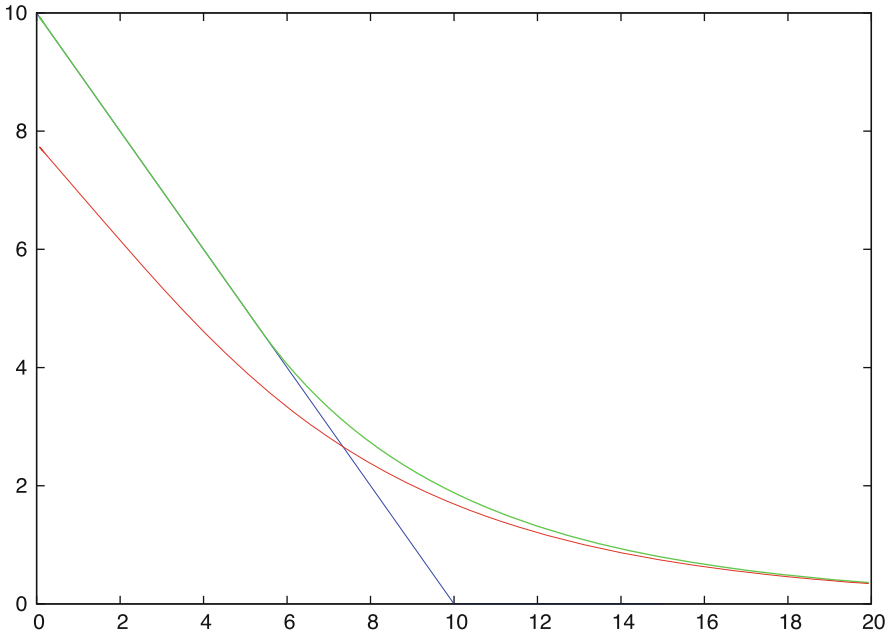

Fig. 4.13 Value $V(S,0)$ of an American put (in *green*) with $K = 10$, $r = 0.25$, $\sigma = 0.6$, $T = 1$ and dividend flow $\delta = 0.2$. For special values see Table 4.1. The corresponding curve of a European option in *red*, the payoff in *blue*

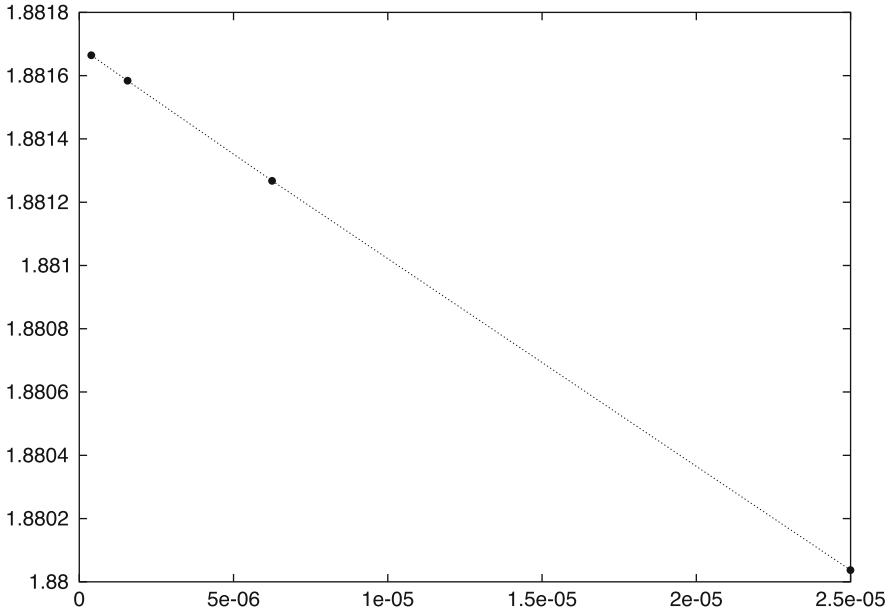

Fig. 4.14 Approximations V depending on Δ^2, with $\Delta = (x_{\max} - x_{\min})/m = 1/\nu_{\max}$; results of Fig. 4.13 and Table 4.1

Table 4.1 Results reported in Fig. 4.13

$m = \nu_{\max}$	$V(10,0)$
50	1.8562637
100	1.8752110
200	1.8800368
400	1.8812676
800	1.8815842
1600	1.8816652

corresponding call. But this is only a necessary criterion for accuracy; the number of matching digits of (A.23) does not relate to the number of correct digits of $V(S,0)$.

4.7.2 Extrapolation

The obviously reasonable error model sketched above suggests applying (4.50) to obtain an improved approximation η at practically zero cost. Such a procedure is called *extrapolation* ($\longrightarrow$ Exercise 1.11). In a graphical illustration η over Δ^2 as in Fig. 4.14, extrapolation amounts to construct a straight line through two of the calculated points. The value of the straight line for $\Delta^2 = 0$ gives the extrapolated value from

$$\eta^* \approx \frac{4\eta_2 - \eta_1}{3}\,. \tag{4.52}$$

In our example, this procedure allows to estimate the correct value to be close to 1.8817. Combining, for example, two approximations of rather low quality, namely, $m = 50$ with $m = 100$, gives already an extrapolated approximation of 1.8815. And based on the two best approximations of Table 4.1, the extrapolated approximation is 1.881690.[13]

Typically, the extrapolation formula provided by (4.52) is significantly more accurate than η_2. But we have no further information on the accuracy of η_2 from the calculated η_1, η_2. Calculating a third approximation η_3 reveals more information. For example, a higher-order extrapolation can be constructed ($\longrightarrow$ Exercise 4.15). Figure 4.15 reports on the accuracies.

The convergence rate in Theorem 4.4 was derived under the assumptions of a structured equidistant grid and a $\mathcal{C}^4$-smooth solution. Practical experiments with nonuniform grids and nonsmooth data suggest that the convergence rate may still behave reasonably. But the finite-difference discretization error is not the whole story. The more flexible finite-element approaches in Chap. 5 will shed light on convergence under more general conditions.

[13] With $m = 20000$, our best result was 1.8816935.

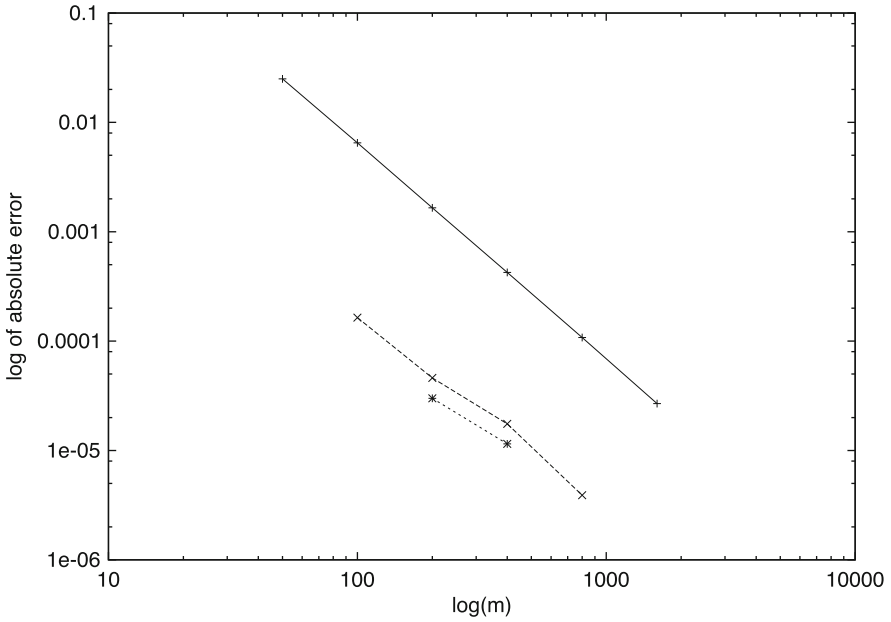

Fig. 4.15 Finite-difference methods, log of absolute error in $V(K, 0)$ over $\log(m)$, where $m = \nu_{\max}$, and the basis of the logarithm is 10. *Solid line*: plain algorithm, results in Table 4.1; *dashed line*: extrapolation (4.52) based on two approximations; *dotted line*: higher-order extrapolation of Exercise 4.15. Note that the axes in Fig. 4.15 are completely different from those of Fig. 4.14

4.8 Analytic Methods

Numerical methods typically are designed such that they achieve convergence. So, in principle, every accuracy can be reached, only limited by the available computer time and by hardware restrictions. In several cases this high potential of numerical methods is not needed. Rather, some analytic formula may be sufficient that delivers medium accuracy at low cost. Such "analytic methods" have been developed. Often their accuracy is reasonable as compared to the underlying modeling error. The limited accuracy goes along with a nice feature that is characteristic for analytic methods: their costs are clear, and known in advance.

In reality there is hardly a clear-cut between numerical and analytic methods. On the one hand, numerical methods require analysis for their derivation. And on the other hand, analytic methods involve numerical algorithms. These may be elementary evaluations of functions like the logarithm or the square root as in the Black–Scholes formula, or may consist of a sub-algorithm like Newton's iteration for zero finding.[14] There is hardly a purely analytic method.

[14]The latter situation might cause some uncertainty on the costs.

The finite-difference approach, which approximates the surface $V(S,t)$, requires intermediate values for $0 < t < T$ for the purpose of approximating $V(S,0)$. In the financial practice one is basically interested in values for $t = 0$, intermediate values are rarely asked for. So the only temporal input parameter is the time to maturity $T - t$ (or T in case the current time is set to zero, $t = 0$). Recall that also in the Black–Scholes formula, time only enters in the form $T - t$ (⟶ Appendix A.4). So it makes sense to write the formula in terms of the time to maturity τ,

$$\tau := T - t\,.$$

Setting $\tilde{V}(S,\tau) := V(S,T-\tau) = V(S,t)$ leads to a PDE for $\tilde{V}$. We drop the tilde (throughout Sect. 4.8), and arrive at a compact version of the Black–Scholes formulas (A.15) or (A.17),

$$\begin{aligned}
d_1(S,\tau;K,r,\sigma) &:= \tfrac{1}{\sigma\sqrt{\tau}}\left\{\log\tfrac{S}{K} + \left(r + \tfrac{\sigma^2}{2}\right)\tau\right\}\,,\\
d_2(S,\tau;K,r,\sigma) &:= \tfrac{1}{\sigma\sqrt{\tau}}\left\{\log\tfrac{S}{K} + \left(r - \tfrac{\sigma^2}{2}\right)\tau\right\} = d_1 - \sigma\sqrt{\tau}\,,\\
V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K,r,\sigma) &= -SF(-d_1) + K\mathrm{e}^{-r\tau}F(-d_2)\,,\\
V_{\mathrm{C}}^{\mathrm{Eur}}(S,\tau;K,r,\sigma) &= SF(d_1) - K\mathrm{e}^{-r\tau}F(d_2)\,.
\end{aligned} \tag{4.53}$$

(dividend-free case). F denotes the cumulative standard normal distribution function. For dividend-free vanilla options we only need an approximation formula for the American put $V_{\mathrm{P}}^{\mathrm{Am}}$; the other cases are covered by the Black–Scholes formula.

This Section introduces four analytic methods. The first two (Sects. 4.8.1 and 4.8.2) are described in detail such that the implementation of the algorithms is an easy matter. Of the method of lines (in Sect. 4.8.3) only basic ideas are set forth. More detail is presented on the integral representation (Sect. 4.8.4). We assume $r > 0$.

4.8.1 *Approximation Based on Interpolation*

If a lower bound V^{low} and an upper bound V^{up} on the American put are available,

$$V^{\mathrm{low}} \le V_{\mathrm{P}}^{\mathrm{Am}} \le V^{\mathrm{up}}\,,$$

then the idea is to construct an α aiming at

$$V_{\mathrm{P}}^{\mathrm{Am}} = \alpha V^{\mathrm{up}} + (1-\alpha)V^{\mathrm{low}}\,.$$

This is the approach of [204]. The parameter α, $0 \le \alpha \le 1$, defines an interpolation between V^{low} and V^{up}. Since $V_{\mathrm{P}}^{\mathrm{Am}}$ depends on the market data S, τ, K, r, σ, the single parameter α and the above interpolation can not be expected to provide an exact

value of $V_{\mathrm{P}}^{\mathrm{Am}}$. (An exact value would mean that an exact formula for $V_{\mathrm{P}}^{\mathrm{Am}}$ would exist.) Rather a formula for α is developed as a function of S, τ, K, r, σ such that the interpolation formula

$$\alpha V^{\mathrm{up}} + (1-\alpha)V^{\mathrm{low}} \tag{4.54}$$

provides a good approximation for a wide range of market data. The smaller the gap between V^{low} and V^{up}, the better is the approximation.

An immediate candidate for the lower bound V^{low} is the value $V_{\mathrm{P}}^{\mathrm{Eur}}$ provided by the Black–Scholes formula,

$$V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K) \le V_{\mathrm{P}}^{\mathrm{Am}}(S,\tau;K)\,.$$

From (4.27) the left-hand boundary condition of a European put with strike $\tilde{K}$ is $\tilde{K}\mathrm{e}^{-r\tau}$ for all τ and all $\tilde{K}$. Clearly, for $\tilde{K} = K\mathrm{e}^{r\tau}$ and $S = 0$,

$$V_{\mathrm{P}}^{\mathrm{Am}}(0,\tau;K) = V_{\mathrm{P}}^{\mathrm{Eur}}(0,\tau;K\mathrm{e}^{r\tau})\,,$$

since both sides equal the payoff value K. From the properties of the American put we know $\frac{\partial V}{\partial S} \ge -1$ and $\frac{\partial^2 V}{\partial S^2} \ge 0$. Hence we conclude that

$$V_{\mathrm{P}}^{\mathrm{Am}}(S,\tau;K) \le V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K\mathrm{e}^{r\tau})$$

at least for small S. In fact, this inequality holds for all $S > 0$, which can be shown with Jensen's inequality, see Appendix B.1. In summary, the upper bound is

$$V^{\mathrm{up}} := V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K\mathrm{e}^{r\tau})\,,$$

see Fig. 4.16. The resulting approximation formula is

$$\overline{V} := \alpha V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K\mathrm{e}^{r\tau}) + (1-\alpha)V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau;K)\,. \tag{4.55}$$

The parameter α depends on S, τ, K, r, σ, so does $\overline{V}$. Actually, the Black–Scholes formula (4.53) suggests that α and $\overline{V}$ only depend on the three dimensionless parameters

$$S/K \text{ (“moneyness”)}\,,\ r\tau\,,\ \text{and } \sigma^2\tau\,.$$

The approximation must be constructed such that the lower bound $(K-S)^+$ of the payoff is obeyed. As we will see, all depends on the free boundary S_{f}, which must be approximated as well.

Johnson [204] sets up a model for α with two free parameters $a_0,\ a_1$, which were determined by carrying out a regression analysis based on computed values of $V_{\mathrm{P}}^{\mathrm{Am}}$.

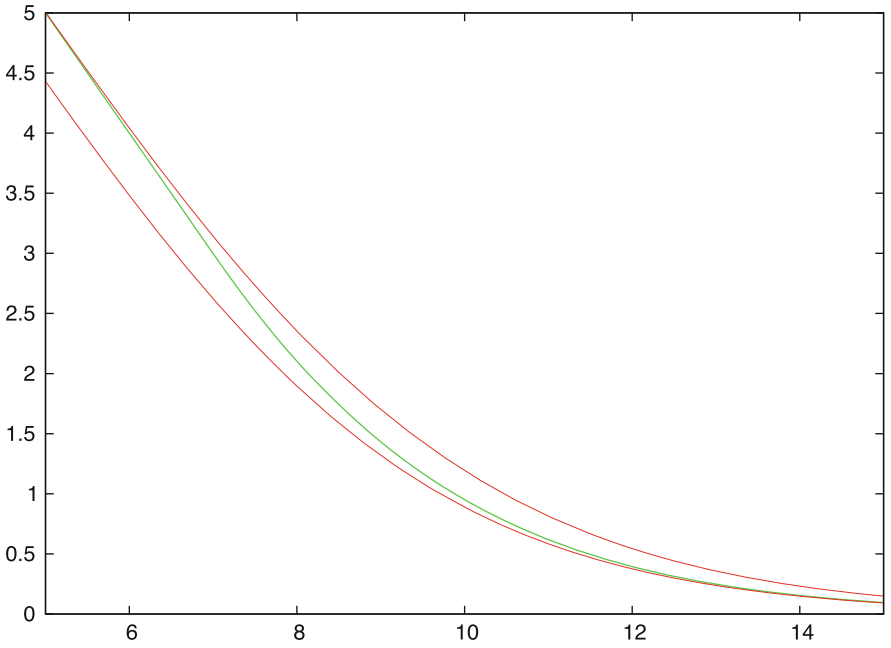

Fig. 4.16 Bounds on an American put $V(S,t;K)$ for $t = 0$ as function of S, with $K = 10$, $r = 0.06$, $\sigma = 0.3$, $\tau = 1$. Medium curve (in *green*): American put; lower curve (*red*): European put $V^{\text{Eur}}(S,0;K)$; upper curve (*red*): European put $V^{\text{Eur}}(S,0;\tilde{K})$, with $\tilde{K} = K\mathrm{e}^{r\tau}$

The result is

$$\alpha := \left(\frac{r\tau}{a_0 r\tau + a_1}\right)^{\beta}, \text{ where } \beta := \frac{\ln(S/S_{\mathrm{f}})}{\ln(K/S_{\mathrm{f}})}, \qquad (4.56)$$
$$a_0 = 3.9649\,, \quad a_1 = 0.032325\,.$$

The ansatz for α is designed such that for $S = K$ (and hence $\beta = 1$) upper and lower bound behavior and calculated option values can be matched with reasonable accuracy with only two parameters a_0, a_1. The S-dependent β is introduced to improve the approximation for $S < K$ and $S > K$. Obviously, $S = S_{\mathrm{f}} \Rightarrow \beta = 0 \Rightarrow \alpha = 1$, which captures the upper bound. And for the lower bound, $\alpha = 0$ is reached for $S \to \infty$, and for $r\tau = 0$. (The reader may discuss (4.56) to check the assertions.)

The model for α of Eq. (4.56) involves the unknown free-boundary curve S_{f}. To approximate S_{f}, observe the extreme cases

$$S_{\mathrm{f}} = K \quad \text{for} \quad \tau = 0\,,$$
$$S_{\mathrm{f}} = K\frac{2r}{\sigma^2 + 2r} \quad \text{for} \quad T \to \infty\,.$$

(For the latter case consult Exercise 4.7 and Appendix A.5.) This motivates to set the approximation $\overline{S}_f$ for S_f as

$$\overline{S}_f := K\left(\frac{2r}{\sigma^2 + 2r}\right)^{\gamma}, \tag{4.57}$$

for a suitably modeled exponent γ. To match the extreme cases, γ should vanish for $\tau = 0$, and $\gamma \approx 1$ for large values of τ. [204] suggests

$$\begin{aligned} \gamma &:= \frac{\sigma^2\tau}{b_0\sigma^2\tau + b_1}, \\ b_0 &= 1.04083, \quad b_1 = 0.00963. \end{aligned} \tag{4.58}$$

The constants b_0 and b_1 were again obtained by a regression analysis.

The analytic expressions of (4.57), (4.58) provide an approximation $\overline{S}_f$ of S_f, and then by (4.56), (4.55) an approximation $\overline{V}$ of V_P^{Am} for $S > S_f$, based on the Black–Scholes formulas (4.53) for V_P^{Eur}.

Algorithm 4.17 (Interpolation)

For given S, τ, K, r, σ evaluate $\gamma, \overline{S}_f, \beta$ based on $\overline{S}_f$ and α.

Evaluate the Black–Scholes formula for V_P^{Eur}

for the arguments in (4.55).

Then $\overline{V}$ from (4.55) is an approximation to V_P^{Am} for $S > \overline{S}_f$.

This purely analytic method is fast and simple. Numerical experiments show that the approximation quality of $\overline{S}_f$ is poor. But for S not too close to $\overline{S}_f$ the approximation quality of $\overline{V}$ is quite good. The error is small for $r\tau \leq 0.125$, which is satisfied for average values of the risk-free rate r and time to maturity τ. For larger values of $r\tau$, when the gap between lower and upper bound widens, the approximation works less well. An extension to options on dividend-paying assets is given in [42].

4.8.2 Quadratic Approximation

Next we describe an analytic method due to [252]. Recall that in the continuation region both V_P^{Am} and V_P^{Eur} obey the Black–Scholes equation. Since this equation is linear, also the difference

$$p(S, \tau) := V_P^{Am}(S, \tau) - V_P^{Eur}(S, \tau) \tag{4.59}$$

satisfies the Black–Scholes equation. The relation $V^{\mathrm{Am}} \geq V^{\mathrm{Eur}}$ suggests to interpret the difference p as *early-exercise premium*. Since both $V_{\mathrm{P}}^{\mathrm{Am}}$ and $V_{\mathrm{P}}^{\mathrm{Eur}}$ have the same payoff, the terminal condition for $\tau = 0$ is zero, $p(S,0) = 0$. The closeness of $p(S,\tau)$ to zero should scale roughly by

$$H(\tau) := 1 - \mathrm{e}^{-r\tau} \,. \tag{4.60}$$

This motivates introducing a scaled version f of p,

$$p(S,\tau) =: H(\tau) f(S, H(\tau)) \tag{4.61}$$

For the analysis we repeat the Black–Scholes equation, here for $p(S,\tau)$, where subscripts denote partial differentiation, and $q := \frac{2r}{\sigma^2}$:

$$-\frac{q}{r} p_\tau + S^2 p_{SS} + qSp_S - qp = 0 \tag{4.62}$$

Substituting (4.61) and

$$p_S = Hf_S \,,\; p_{SS} = Hf_{SS} \,,\; p_\tau = H_\tau f + Hf_H H_\tau$$

and using

$$\frac{1}{r} H_\tau = 1 - H$$

yields after a short calculation (the reader may check) the modified version of the Black–Scholes equation

$$S^2 f_{SS} + qSf_S - \frac{q}{H} f\big[1 + H(1-H)\frac{f_H}{f}\big] = 0 \,. \tag{4.63}$$

H and q are nonzero for $r > 0$. Note that (4.63) is the "full" equation, nothing is simplified yet. No partial derivative with respect to t shows up, but instead the partial derivative f_H .

At this point, following [252], we introduce a simplifying approximation. The factor $H(H-1)$ for the H varying in the range $0 \leq H < 1$ is a quadratic term with maximum value of $1/4$, and close to zero for $\tau \approx 0$ and for large values of τ, compare (4.60). This suggests that the term

$$H(1-H)\frac{f_H}{f} \tag{4.64}$$

may be small compared to 1, and to neglect it in (4.63). (This motivates the name "quadratic approximation.") The resulting equation

$$S^2 f_{SS} + qSf_S - \frac{q}{H} f = 0 \tag{4.65}$$

is an ordinary differential equation with analytic solution, parameterized by H. An analysis similar as in Exercise 4.7 leads to the solution

$$f(S) = \alpha S^\lambda \ , \ \text{where} \ \ \lambda := -\frac{1}{2}\left\{(q-1) + \sqrt{(q-1)^2 + \frac{4q}{H}}\right\} \ , \tag{4.66}$$

for a parameter α. Combining (4.59), (4.61) and (4.66) we deduce for $S > S_\mathrm{f}$ the approximation $\overline{V}$

$$V_\mathrm{P}^\mathrm{Am}(S,\tau) \approx \overline{V}(S,\tau) := V_\mathrm{P}^\mathrm{Eur}(S,\tau) + \alpha H(\tau) S^\lambda \, . \tag{4.67}$$

The parameter α must be such that $\overline{V}$ reaches the payoff at S_f,

$$V_\mathrm{P}^\mathrm{Eur}(S_\mathrm{f},\tau) + \alpha H S_\mathrm{f}^\lambda = K - S_\mathrm{f} \, . \tag{4.68}$$

Here S_f is parameterized by H via (4.60), and therefore depends on τ. To fix the two unknowns S_f and α let us warm up the high-contact condition. This requires the partial derivative of $\overline{V}$ with respect to S. The main part is

$$\frac{\partial V_\mathrm{P}^\mathrm{Eur}(S,\tau)}{\partial S} = F(d_1) - 1$$

where F is the cumulative normal distribution function, and d_1 (and below d_2) are the expressions defined by (4.53). d_1 and d_2 depend on all relevant market parameters; we emphasize the dependence on S by writing $d_1(S)$. This gives the high-contact condition

$$\frac{\partial \overline{V}(S_\mathrm{f},\tau)}{\partial S} = F(d_1(S_\mathrm{f})) - 1 + \alpha\lambda H S_\mathrm{f}^{\lambda-1} = -1 \, ,$$

and immediately α in terms of S_f:

$$\alpha = -\frac{F(d_1(S_\mathrm{f}))}{\lambda H S_\mathrm{f}^{\lambda-1}} \, . \tag{4.69}$$

Substituting into (4.68) yields one equation for the remaining unknown S_f,

$$V_\mathrm{P}^\mathrm{Eur}(S_\mathrm{f},\tau) - F(d_1(S_\mathrm{f}))\frac{1}{\lambda} S_\mathrm{f} = K - S_\mathrm{f} \, ,$$

which in view of the put-call parity (A.16) and $F(-d) = 1 - F(d)$ reads

$$S_{\mathrm{f}}F(d_1) - K\mathrm{e}^{-r\tau}F(d_2) - S_{\mathrm{f}} + K\mathrm{e}^{-r\tau} - F(d_1)\frac{S_{\mathrm{f}}}{\lambda} - K + S_{\mathrm{f}} = 0\,.$$

This can be summarized to

$$S_{\mathrm{f}}\,F(d_1(S_{\mathrm{f}}))\left[1 - \frac{1}{\lambda}\right] + K\mathrm{e}^{-r\tau}\left[1 - F(d_2(S_{\mathrm{f}}))\right] - K = 0\,. \tag{4.70}$$

Since d_1 and d_2 vary with S_{f}, (4.70) is an implicit equation for S_{f} and must be solved iteratively. In this way a sequence of approximations $S_1, S_2, \ldots$ to S_{f} is constructed. We summarize

Algorithm 4.18 (Quadratic Approximation)

For given S, τ, K, r, σ evaluate $q = \dfrac{2r}{\sigma^2}$, $H = 1 - \mathrm{e}^{-r\tau}$ and λ from (4.66).

Solve (4.70) iteratively for S_{f}.

(This involves a sub-algorithm, from which $F(d_1(S_{\mathrm{f}}))$ should be saved.)

Evaluate $V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau)$ using the Black–Scholes formula (4.53).

$$\overline{V} := V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau) - \frac{1}{\lambda}S_{\mathrm{f}}F(d_1(S_{\mathrm{f}}))\left(\frac{S}{S_{\mathrm{f}}}\right)^{\lambda} \tag{4.71}$$

is the approximation for $S > S_{\mathrm{f}}$,

and $\overline{V} = K - S$ for $S \le S_{\mathrm{f}}$.

Note that $\lambda < 0$, and λ depends on τ via $H(\tau)$. The time-consuming part of the quadratic-approximation method consists of the numerical root finding procedure. But here a moderate accuracy suffices, since a very small error in S_{f} does not affect the error in $\bar{V}$. (⟶ Exercises 4.17 and 4.18)

4.8.3 Analytic Method of Lines

In solving PDEs numerically, the *method of lines* is a well-known approach. It is based on a semidiscretization, where the domain (here the (S,τ) half strip) is replaced by a set of lines parallel to the S-axis, each defined by a constant value of τ. To this end, the interval $0 \le \tau \le T$ is discretized into $\nu_{\max}$ sub-intervals by $\tau_\nu := \nu\Delta\tau$, $\Delta\tau := T/\nu_{\max}$, $\nu = 1, \ldots, \nu_{\max} - 1$. To deserve the attribute "analytic," we assume $\nu_{\max}$ to be small, say, work with three lines. We write the Black–Scholes

equation as in Sect. 4.5.3,

$$-\frac{\partial V(S,\tau)}{\partial \tau} + \mathcal{L}_{\mathrm{BS}}(V(S,\tau)) = 0\,, \tag{4.72}$$

where the negative sign compensates for the transition from t to τ, and replace the partial derivative $\partial V/\partial\tau$ by the difference quotient

$$\frac{V(S,\tau) - V(S,\tau - \Delta\tau)}{\Delta\tau}\,.$$

This gives a semidiscretized version of (4.72), namely, the ordinary differential equation

$$w(S,\tau - \Delta\tau) - w(S,\tau) + \Delta\tau\, \mathcal{L}_{\mathrm{BS}}(w(S,\tau)) = 0\,,$$

which holds for $S > S_{\mathrm{f}}$. Here we use the notation w rather than V to indicate that a discretization error is involved. This semidiscretized version is applied for each of the parallel lines, $\tau = \tau_\nu$, $\nu = 1,\ldots,\nu_{\max} - 1$. Figure 4.17 may motivate the procedure. For each line $\tau = \tau_\nu$, the function $w(S,\tau_{\nu-1})$ is known from the previous line, starting from the known payoff for $\tau = 0$. The equation to be solved for each line τ_ν is

$$\frac{1}{2}\Delta\tau\,\sigma^2 S^2 \frac{\partial^2 w}{\partial S^2} + \Delta\tau\, rS\frac{\partial w}{\partial S} - (1 + \Delta\tau\, r)w = -w(\cdot,\tau_{\nu-1})\,. \tag{4.73}$$

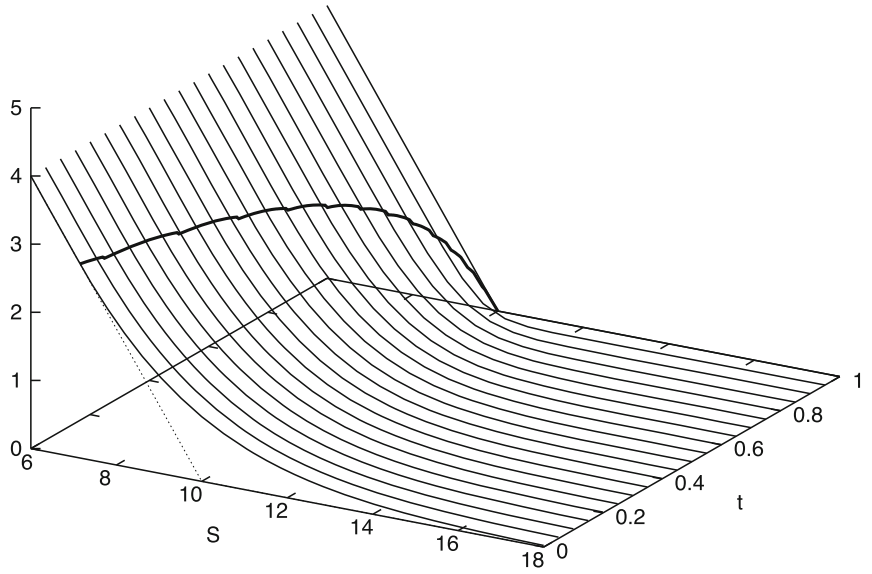

Fig. 4.17 Method of lines, situation as in Fig. 1.5. The early-exercise curve is indicated

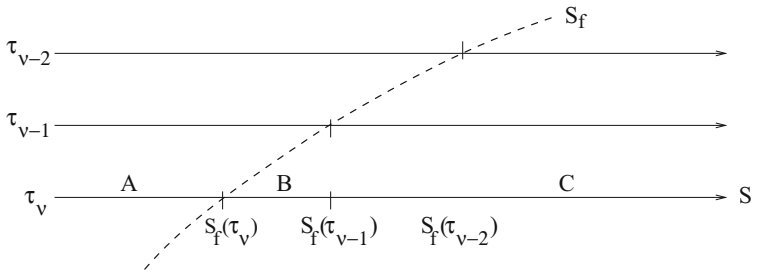

Fig. 4.18 Method of lines, situation along line τ_ν: *A*: solution is given by payoff; *B*: inhomogeneous term of differential equation given by payoff; *C*: inhomogeneous term given by $-w(., \tau_{\nu-1})$

This is a second-order ordinary differential equation for $w(S, \tau_\nu)$, with boundary conditions for $S_f(\tau_\nu)$ and $S \to \infty$. The solution is obtained analytically, similar as in Exercise 4.7. Hence there is no discretization error in S-direction. The right-hand function $-w(S, \tau_{\nu-1})$ is known, and is an inhomogeneous term of the ODE.

The resulting *analytic method of lines* is carried out in [66]. The above describes the basic idea. A complication arises from the early-exercise curve, which separates each of the parallel lines into two parts. Since for the previous line $\tau_{\nu-1}$ the separation point lies more "on the right" (recall that for a put the curve $S_f(\tau)$ is monotonically decreasing for growing τ), the inhomogeneous term $w(\cdot, \tau_{\nu-1})$ consists of two parts as well, but separated differently (see Fig. 4.18). Accordingly, neglecting for the moment the input of previous lines $\tau_{\nu-2}$, $\tau_{\nu-3}, \ldots$, the analytic solution of (4.73) for the line τ_ν consists of three parts, defined on the three intervals

$$\begin{aligned} \text{A:} \quad & 0 < S < S_f(\tau_\nu) \,, \\ \text{B:} \quad & S_f(\tau_\nu) \le S < S_f(\tau_{\nu-1}) \,, \\ \text{C:} \quad & S_f(\tau_{\nu-1}) \le S \,. \end{aligned}$$

On the left-hand interval A, w equals the payoff; nothing needs to be calculated. For the middle interval B the inhomogeneous term $-w(., \tau_{\nu-1})$ is given by the payoff. Since the analytic solution involves two integration constants, and since the inhomogeneous terms differ on the intervals B and C, we encounter together with the unknown $S_f(\tau_\nu)$ five unknown parameters. One of the integration constants is zero because of the boundary condition for $S \to \infty$, similar as in Exercise 4.7. The unknown separation point $S_f(\tau_\nu)$ is again fixed by the high-contact conditions (4.34). Two remaining conditions are given by the requirement that both w and $\frac{\mathrm{d}w}{\mathrm{d}S}$ are continuous at the matching point $S_f(\tau_{\nu-1})$. This fixes all variables for the line τ_ν.

Over all lines, $\nu_{\max}$ type-B intervals are involved, and the only remaining type-C interval is that for $S \ge S_f(\tau_0) = K$. The resulting formulas are somewhat complex, for details see [66]. The method is used along with extrapolation. To this end, carry out the method three times, with $\nu_{\max} = 1, 2, 3$, and denote the results $\overline{V}_1, \overline{V}_2, \overline{V}_3$.

Then the three-point extrapolation formula

$$\overline{V} := \frac{1}{2}\,(\,9\overline{V}_3 - 8\overline{V}_2 + \overline{V}_1\,) \tag{4.74}$$

gives rather accurate results.

The method of lines can be carried out numerically [273]. For lines parallel to the t-axis, see Exercise 4.3 and Fig. 4.21.

4.8.4 *Integral-Equation Method*

Recall for European put options the integral representation (1.66)

$$V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau) = \mathrm{e}^{-r\tau}\int_0^\infty (K-S_T)^+ f_{\mathrm{GBM}}(S_T,\tau;\,S,r-\delta,\sigma)\,\mathrm{d}S_T\,,$$

where $\tau := T - t$ denotes the remaining time to expiration, and f_{GBM} is the density function from (1.64). Solving this integral one arrives at the Black–Scholes formula. We repeat from (4.53) the two functions (here with constant dividend yield rate $\delta \geq 0$),

$$d_1(S,\tau;K) := \frac{\log\frac{S}{K} + \left(r - \delta + \frac{\sigma^2}{2}\right)\tau}{\sigma\sqrt{\tau}}\,, \quad d_2(S,\tau;K) := d_1 - \sigma\sqrt{\tau}\,, \tag{4.75}$$

for $\tau > 0$. With d_1, d_2 evaluated at S, τ, K, recall

$$V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau) = -S\mathrm{e}^{-\delta\tau}F(-d_1) + K\mathrm{e}^{-r\tau}F(-d_2)\,,$$

where F denotes the standard normal cumulative distribution. (See also Appendix A.4.) Further recall from (4.59) the early-exercise premium p, with

$$V_{\mathrm{P}}^{\mathrm{Am}}(S,\tau) = V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau) + p(S,\tau)\,.$$

As suggested by [222] and others, the premium function p can be represented as an integral over functions depending on the free boundary S_{f}. The result is

$$\begin{aligned} V_{\mathrm{P}}^{\mathrm{Am}}(S,\tau) = \quad & V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau) + \\ & + \int_0^\tau [\,rK\mathrm{e}^{-r\xi}F(-d_2(S,\xi;S_{\mathrm{f}}(\tau-\xi))) \\ & \qquad -\delta S\mathrm{e}^{-\delta\xi}F(-d_1(S,\xi;S_{\mathrm{f}}(\tau-\xi)))\,]\,\mathrm{d}\xi\,. \end{aligned} \tag{4.76}$$

The integral is identical to

$$\int_0^\tau [\, rK\mathrm{e}^{-r(\tau-\xi)}F(-d_2(S,\tau-\xi;S_\mathrm{f}(\xi))) \\ -\delta S\mathrm{e}^{-\delta(\tau-\xi)}F(-d_1(S,\tau-\xi;S_\mathrm{f}(\xi)))\,]\,\mathrm{d}\xi\,. \tag{4.77}$$

4.8.4.1 Integral Equation for S_f

Substitute $V(S_\mathrm{f}(\tau),\tau) = K - S_\mathrm{f}(\tau)$ into (4.76) and obtain

$$\begin{aligned} K - S_\mathrm{f}(\tau) = &- S_\mathrm{f}(\tau)\,\mathrm{e}^{-\delta\tau}\,F(-d_1(S_\mathrm{f}(\tau),\tau;K)) \\ &+ K\mathrm{e}^{-r\tau}\,F(-d_2(S_\mathrm{f}(\tau),\tau;K)) \\ &+ \int_0^\tau [\, rK\mathrm{e}^{-r\xi}\,F(-d_2(S_\mathrm{f}(\tau),\xi;S_\mathrm{f}(\tau-\xi))) \\ &\qquad -\delta S_\mathrm{f}(\tau)\,\mathrm{e}^{-\delta\xi}\,F(-d_1(S_\mathrm{f}(\tau),\xi;S_\mathrm{f}(\tau-\xi)))\,]\,\mathrm{d}\xi\,. \end{aligned} \tag{4.78}$$

This constitutes an integral equation for the free-boundary function (early-exercise curve) $S_\mathrm{f}(\tau)$ of an American put.

4.8.4.2 Numerical Solution of the Integral Equation

We denote the integrand in (4.78) by $g(S_\mathrm{f}(\tau), S_\mathrm{f}(\tau-\xi), \xi)$ ($\longrightarrow$ Exercise 4.19). So the integral equation reads

$$K - S_\mathrm{f}(\tau) = V_\mathrm{P}^\mathrm{Eur}(S_\mathrm{f}(\tau),\tau) + \int_0^\tau g(S_\mathrm{f}(\tau), S_\mathrm{f}(\tau-\xi), \xi)\,\mathrm{d}\xi\,.$$

Let the τ-interval be subdivided by discrete τ_ν into M subintervals, with $\tau_0 = 0$, $\tau_M = \tau$, and with equidistant steps $\Delta\tau = \tau/M$, and $\tau_\nu = \nu\Delta\tau$. The numerical treatment resembles that for ODE initial-value problems. Basically the integral is approximated by a composite trapezoidal sum (C.2). Note from Appendix A.5 that $S_\mathrm{f}(\tau)$ for $\tau \to 0^+$ is known,

$$S_{\mathrm{f}0} := \lim_{\tau\to 0^+} S_\mathrm{f}(\tau) = \min\{K, \frac{r}{\delta}K\}\,.$$

We use the notation $S_{\mathrm{f}\nu} := S_\mathrm{f}(\tau_\nu)$. Specifically for τ_1, the integral and (4.78) can be approximated by the trapezoidal rule

$$K - S_{\mathrm{f}1} = V_\mathrm{P}^\mathrm{Eur}(S_{\mathrm{f}1},\tau_1) + \frac{\Delta\tau}{2}[g(S_{\mathrm{f}1},S_{\mathrm{f}1},\tau_0) + g(S_{\mathrm{f}1},S_{\mathrm{f}0},\tau_1)]\,, \tag{4.79}$$

which is solved iteratively for its only unknown $S_{\mathrm{f}1}$ by any root-finding procedure. After $S_{\mathrm{f}1}$ is calculated to sufficient accuracy, the next equation is

$$K - S_{\mathrm{f}2} = V_{\mathrm{P}}^{\mathrm{Eur}}(S_{\mathrm{f}2}, \tau_2) + \frac{\Delta\tau}{2}[g(S_{\mathrm{f}2}, S_{\mathrm{f}2}, \tau_0) + 2g(S_{\mathrm{f}2}, S_{\mathrm{f}1}, \tau_1) + g(S_{\mathrm{f}2}, S_{\mathrm{f}0}, \tau_2)]\,,$$

which is solved for $S_{\mathrm{f}2}$. In this way, the composite trapezoidal sum builds up until we reach the final iteration for $S_{\mathrm{f}M}$. So, recursively for $k = 2, \ldots, M$ solve

$$K - S_{\mathrm{f}k} = V_{\mathrm{P}}^{\mathrm{Eur}}(S_{\mathrm{f}k}, \tau_k) + \\ + \frac{\Delta\tau}{2}\left[g(S_{\mathrm{f}k}, S_{\mathrm{f}k}, \tau_0) + 2\sum_{\nu=1}^{k-1} g(S_{\mathrm{f}k}, S_{\mathrm{f}(k-\nu)}, \tau_\nu) + g(S_{\mathrm{f}k}, S_{\mathrm{f}0}, \tau_k)\right] \tag{4.80}$$

for $S_{\mathrm{f}k}$. This recursion is run for $\tau = T$ to obtain values for $t = 0$.

The iterative solution of the above nonlinear equations [as (4.79), (4.80)] can be done, for example, by the secant method (C.5). The error control of the integral-equation method represented by (4.80) involves the discretization error of the trapezoidal sum as well as the error remaining when the secant iteration is stopped. Recall that the secant method requires *two* reasonable initial guesses. Alternatively, we recommend the highly robust bisection method. There is ample opportunity to test various strategies (⟶ Exercise 4.20).

4.8.4.3 Evaluation of the Premium

Now, the free boundary S_{f} is approximated by the chain of points

$$(\tau_0, S_{\mathrm{f}0}), (\tau_1, S_{\mathrm{f}1}), \ldots, (\tau_M, S_{\mathrm{f}M})\,.$$

Based on this approximation, the evaluation of (4.76) is a simple task. Apply the analogous trapezoidal sum with the same discretization to approximate $V(S, \tau)$ for $\tau = \tau_M$:

$$V(S, \tau) \approx V_{\mathrm{P}}^{\mathrm{Eur}}(S, \tau) + \\ + \frac{\Delta\tau}{2}[g(S, S_{\mathrm{f}M}, 0) + 2\sum_{\nu=1}^{M-1} g(S, S_{\mathrm{f}(M-\nu)}, \tau_\nu) + g(S, S_{\mathrm{f}0}, \tau)]\,. \tag{4.81}$$

The evaluation of (4.81) does not need any further iteration and is much cheaper than the preceding recursion (4.80).

4.8.4.4 Calculation of the Greeks

The same holds true for evaluating greeks. After calculating the partial derivatives of (4.76), one obtains corresponding formulas for the greeks. For example, delta is given by the formula

$$\Delta_{\mathrm{P}}^{\mathrm{Am}} = -\mathrm{e}^{-\delta\tau}\, F(-d_1) - \int_0^\tau g_{\mathrm{P}}^{\Delta}\, \mathrm{d}\xi$$

for a function g_{P}^{Δ} defined below. The calculation works as simply as in (4.81); the free boundary S_{f} is not calculated again. And similarly, other greeks are obtained, both for put and call. The resulting formulas are given in [190]. With the version of (4.77), and d_1 evaluated at the arguments $(S, \tau - \xi, S_{\mathrm{f}}(\xi))$,

$$g_{\mathrm{P}}^{\Delta} = \delta \mathrm{e}^{-\delta(\tau-\xi)}\, F(-d_1(S, \tau-\xi, S_{\mathrm{f}}(\xi))) + \frac{\mathrm{e}^{-d_1^2/2}}{\sqrt{2\pi}}\, \mathrm{e}^{-\delta(\tau-\xi)}\, \frac{rK - \delta S_{\mathrm{f}}(\xi)}{\sigma S_{\mathrm{f}}(\xi)\sqrt{\tau-\xi}}\,.$$

For these arguments and $\xi \to \tau$, $|d_1|$ is getting infinite, and

$$g_{\mathrm{P}}^{\Delta} = \begin{cases} 0 & \text{for } S > S_{\mathrm{f}}\,, \\ \delta & \text{for } S < S_{\mathrm{f}}\,. \end{cases}$$

4.8.5 Other Methods

The early-exercise curve $S_{\mathrm{f}}(\tau)$ can be approximated by pieces of exponential functions

$$B\exp(b\tau) \quad \text{for } \tau_1 \le \tau \le \tau_2\,,$$

for parameters B, b and suitable intervals for τ. Substituting this expression for $S_{\mathrm{f}}(\tau)$ into d_1 and d_2 in (4.76) leads to the observation that the integrals can be evaluated analytically in terms of the distribution function F. The parameters B, b are determined such that the high-contact boundary-condition condition is satisfied. Depending on the number of pieces of exponential functions, a good approximation of (4.76) is obtained. This is the method of [208]. The accuracy of the highly efficient three-piece approximation corresponds to that of the integral-equation method with about $M = 100$ subintervals.

[56] establishes LUBA, an analytic method for American calls. The derivation is beyond the scope of this textbook, but is worth at least a brief sketch because of its striking computational power. The method starts from a *capped call*, which is basically a vanilla European call, with the exception that for $t < T$ the option is exercised at the first time t such that S_t reaches the cap. The price

of the capped call can be replicated with two barrier options. Their analytical formulas constitute a lower bound LB on the option. This in turn, via the integral representation (4.76) lends to an upper bound UB. Then LB and UB are interpolated with a regression ansatz comparable to the interpolation of Sect. 4.8.1. The resulting specific approximation of [56] is called LUBA, which stands for lower upper bound approximation.

4.9 Criteria for Comparisons

In this chapter, we have learned about the basic structure of finite-difference methods, and we have studied several analytic approaches. How do these methods compare? As we shall see, this question is difficult to answer. There are several criteria to judge the performance of a computational method. The criteria include reliability, range of applicability, amount of information provided by the method, and speed, and error. Speed and error are relatively easy to compare, and we shall concentrate on these two criteria.

For the computational arena, we need to define a set of test examples, based on which we have to calculate a *benchmark* in high accuracy. Results of any chosen method will be compared to the benchmark. To measure the deviation, a suitable error must be defined. This Sect. 4.9 roughly sketches the steps of a comparison.

4.9.1 Set of Test Examples

We concentrate on the valuation of plain-vanilla options. This restriction to vanillas has the advantage that all kind of numerical methods are applicable and can be compared. And we confine ourselves to the valuation of American put options. The parameters $K, S, T, \sigma, r, \delta$ are chosen

- $K = 100$
- $S \in \{90,\ 100,\ 110,\ 150\}$
- $T \in \{0.5,\ 1,\ 2\}$
- $\sigma \in \{0.1,\ 0.3,\ 0.5\}$
- $r \in \{0.05,\ 0.1\}$ for $\delta = 0$; $\quad r \in \{0.15,\ 0.2\}$ for $\delta = 0.1$

Altogether these are 72 combinations with dividend rate $\delta = 0$ and as many for $\delta = 0.1$. But for $\sigma = 0.1$, in 12 of these cases, either

$$V(S,0) \approx 0 \quad \text{or} \quad V(S,0) = \text{payoff}$$

occurs. In these cases, a relative error is meaningless, or nothing is to be calculated. Hence we remove theses 12 cases ($\sigma = 0.1, S = 90, S = 150$). The remaining 60 parameter combinations were organized into two files.[15]

For each set of parameters we calculated $V(S, 0)$ with rather high accuracy (7–8 decimal digits). To this end, we applied as reference method an extrapolation based on finite-difference approximations, as suggested in Sect. 4.7.2. The obtained values complete the benchmark files. Any method can be compared to the benchmark as long as its relative error is not smaller than 10^{-6}.

4.9.2 Measure of the Error

To measure performances, we calculate the *root mean square relative error*

$$\mathrm{RMS} := \sqrt{\frac{1}{60}\sum_{i=1}^{60}\left(\frac{\overline{V}_i - V_i}{V_i}\right)^2}. \tag{4.82}$$

Here V_i denotes the "accurate" benchmark value of the ith parameter combination, and $\overline{V}_i$ denotes the value calculated with the method whose performance is to be measured.

4.9.3 Arena of Competing Methods

We have chosen the following prototypical methods:

B-M: binomial method with M time steps, Algorithm 1.4,
$M = 12, 25, 50, \ldots, 1600$;
FD-BS-M: finite differences Brennan–Schwartz, Algorithm 4.15,
with $M := m = \nu_{\max}$, $M = 200, 400, \ldots, 6400$;
J: Johnson's interpolation, Algorithm 4.17;
Q: quadratic approximation, Algorithm 4.18;
I-M: integral-equation method with M subintervals, Sect. 4.8.4,
$M = 50, 100, \ldots, 3200$;
FD-BS-ex: version of FD-BS with two solutions with M and $M/2$
and extrapolation.

Keep in mind that the above methods provide different amount of information; in some sense we compare apples with oranges. The integer M represents a fineness of

[15]The files `BENCHMARK00` for $\delta = 0$ and `BENCHMARK01` for $\delta = 0.1$ can be found on www.compfin.de.

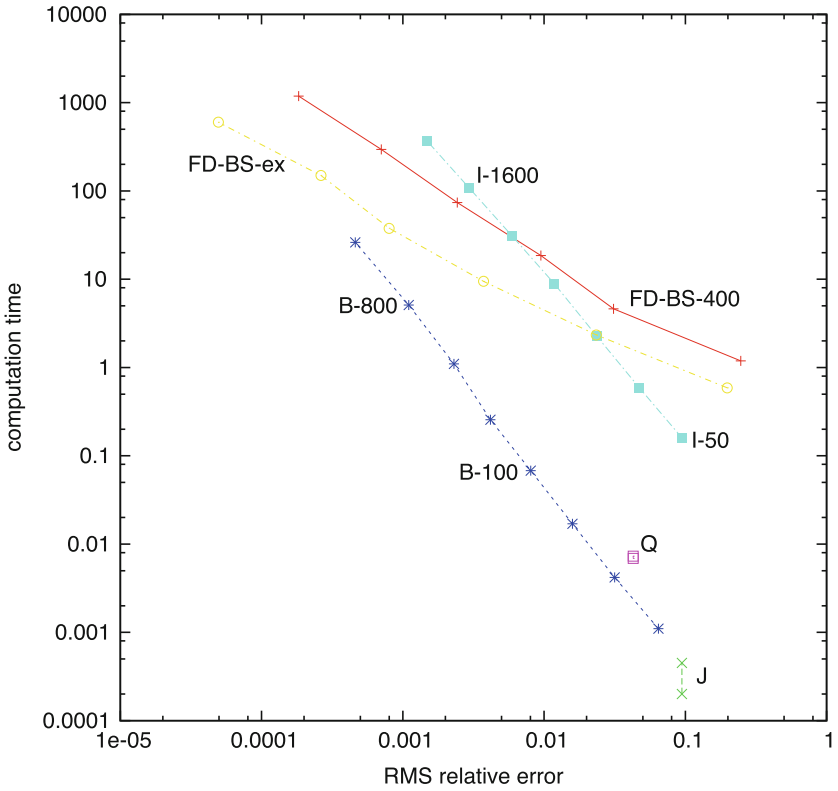

Fig. 4.19 Computing times and RMS errors of several methods, see the text. *Points mark* calculated RMS errors; corresponding *points* are connected by *lines*

discretization, which is consecutively doubled for clarity of exposition. Computing times in Fig. 4.19 report the time in seconds needed to valuate all of the 60 options for $\delta = 0$; overhead is subtracted.[16] The log scaling in Fig. 4.19 is most practical ($\longrightarrow$ Exercise 4.21). For the versions with shortest computing time (J), the time is hardly measurable, which is indicated by a bar of likely computing times.

In Fig. 4.19, the accuracy orders of the various methods can not be seen directly. The convergence rate would become apparent in case the absolute error is depicted over the grid size. Such a figure reveals the first-order convergence of the binomial method and the integral-equation method, and essentially a second-order convergence of the finite-difference method.

[16] All of the above methods were implemented in FORTRAN (F90 compiler) and run on a DS20 processor.

4.9.4 Preliminary Results

In the sense of Pareto optimization, smaller values in Fig. 4.19 are preferred to larger ones. Entries in the lower left part of the figure refer to methods with higher efficiency. The Pareto frontier in this figure is largely dominated by the binomial method (B). This holds at least for medium demands for accuracy. Both the analytic methods (J) and (Q) do not need the evaluation of the the Black–Scholes formula and hence $\sqrt{}$, log, exp in full accuracy. So their evaluation can be accelerated. Hence, for low accuracy, Johnson's interpolation method (J) and the quadratic approximation (Q) are competitive. This is not clear from the figure, where unnecessary accuracy of the underlying Black–Scholes formula falsely suggests that the quadratic approximation (Q) is dominated by the binomial method. For high demands for accuracy, the finite-difference method is competitive. The basic version of the binomial method dominates the basic version of the integral-equation method (I). The aspect of convergence applies to FD, B, I, but not to the fixed accuracy of Q, J. This may be seen as distinction between a numerical method and an analytic method.

4.9.5 Outlook

The above observations should not be considered as definite recommendations. It is important to realize that the conclusions refer to speed and RMS error only. Several aspects are neglected and lacking. For example, the finite-difference method calculates the surface of the value function $V(S, t)$, and provides more information than the binomial method. Or, the integral-equation method allows to calculate the greeks more effectively, and approximates the early-exercise curve very well (B does not). The above has selected one representative method of important classes of methods. These basic versions are implemented and compared. There are more efficient methods not shown in Fig. 4.19. For example, LUBA has shown to dominate the methods with comparable accuracy. Neither the highly efficient front-fixing methods are shown, nor the improvement [175] of the integral method, nor the fast approximation by exponential pieces. Improvements differ in the degree of speedup. Further, storage requirements are not taken into account. Implementation details do matter! And applied to a specific type of exotic option, the prototype methods chosen for Fig. 4.19 may behave and compare differently. Monte Carlo methods are not included at all, because their merits are beyond vanilla options. So the conclusions of this section aim at basic principles. They are tentative, and not comprehensive. We do not answer the question, what might be the "best" method for a particular application. For early comparisons, see [4, 56, 57, 211]. More recent developments have not been compared.

4.10 Notes and Comments

On Sect. 4.1

General references on numerical PDEs include [80, 281, 341, 356, 369]. A special solution of (4.2) is

$$y(x,\tau) = \frac{1}{2\sqrt{\pi\tau}} \exp\left(-\frac{x^2}{4\tau}\right).$$

For small values of τ, the transformation (4.3) may take bad values in the argument of the exponential function because q_δ can be too large. The result will be an overflow. In such a situation, the transformation

$$\begin{aligned} \tau &:= \tfrac{1}{2}\sigma^2(T-t) \\ x &:= \log\left(\tfrac{S}{K}\right) + \left(r - \delta - \tfrac{\sigma^2}{2}\right)(T-t) \\ y(x,\tau) &:= \mathrm{e}^{-rt} V(S,t) \end{aligned}$$

can be used as alternative [28]. Again (4.2) results, but initial conditions and boundary conditions must be adapted appropriately (see also Appendix A.6). The equations also hold for options on foreign currencies. Then δ represents the foreign interest rate. As will be seen in Sect. 6.4, the quantities q and q_δ are basically the Péclet number. It turns out that large values of the Péclet number are a general source of difficulties. For other transformations see [381]. *Well-posed* means the existence of a unique solution that depends continuously on the data.

For the valuation of American options in case of **discrete dividend** payments there is a big difference between call and put. A call is exercised immediately prior to the dividend date, provided some analytically known criteria are satisfied [234]. In contrast, a put must be calculated numerically. By arbitrage reasons, the stock price jumps at the ex-dividend date t_D,

$$S_{t_D^+} = S_{t_D^-} - D\,,$$

where D is the net amount paid at t_D. The price V_t of the put does not jump along the path S_t because the option's holder has no benefit from the payment. This continuity of $V(S_t,t)$ can be written

$$V(S,\, t_D^-) = V(S-D,\, t_D^+)\,,$$

which amounts to a jump in the value function $V(S,t)$ at t_D.[17] For a numerical implementation, place a node t_ν at t_D, interrupt the integration of the PDE at t_D, and

[17] For tree methods, dividends are discussed in Appendix D.2.

apply interpolation to evaluate V at $S_i - D$ in case this is not a node. Then the PDE is applied again. For a method-of-lines approach see [273]. Exercise 4.1b provides some insight into the early-exercise structure. For $t_D < t < T$ the early-exercise curve is that of a non-dividend paying stock [27, 292].

On Sect. 4.2

We follow the notation $w_{i,\nu}$ for the approximation at the node (x_i, τ_ν), to stress the surface character of the solution y over a two-dimensional domain. In the literature a frequent notation is w_i^ν, which emphasizes the different character of the space variable (here x) and the time variable (here τ). Our vectors $w^{(\nu)}$ with components $w_i^{(\nu)}$ come close to this convention.

Finite differences work for nonuniform meshes as well. Then formally the discretization errors are of first order only. But under mild assumptions on a slowly varying mesh, second-order accuracy can be obtained [257].

Summarizing the Black–Scholes equation to

$$\frac{\partial V}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V) = 0 \tag{4.83}$$

where $\mathcal{L}_{\mathrm{BS}}$ represents the other terms of the equation, see Sect. 4.5.3, motivates an interpretation of the finite-difference schemes in the light of numerical ODEs. There the forward approach is known as *explicit Euler method* and the backward approach as *implicit Euler method*. The explicit scheme corresponds to the trinomial-tree method mentioned in Sect. 1.4 [191].

On Sect. 4.3

Crank and Nicolson suggested their approach in 1947 [91]. Theorem 4.4 discusses three main principles of numerical analysis, namely, order of convergence, stability, and efficiency. A Crank–Nicolson variant has been developed that is consistent with the volatility smile, which reflects the dependence of the volatility on the strike [10].

In view of the representation (4.20) the Crank–Nicolson approach corresponds to the ODE *trapezoidal rule*. Following these lines suggests to apply other ODE approaches, some of which lead to methods that relate more than two time levels. In particular, the backward difference formula BDF (4.11) is of interest, which evaluates $\mathcal{L}$ at one time level only. Using formula (4.11) for the time discretization, a three-term recursion involving $w^{(\nu+1)}, w^{(\nu)}, w^{(\nu-1)}$ replaces the two-term recursion (4.24) ($\longrightarrow$ Exercise 4.3). But multistep methods such as BDF may suffer from the lack of smoothness at the exercise boundary. This effect is mollified when the inequality is tackled by a penalty term. But even then it is interesting

to consider other alternatives with better stability properties than Crank–Nicolson. Crank–Nicolson is A-stable, several other methods are L-stable, which better damp out high-frequency oscillation, see [71, 194, 221]. For numerical ODEs we refer to [165, 236]. From the ODE analysis circumstances are known where the implicit Euler method behaves superior to the trapezoidal rule. The latter method may show a *slowly damped* oscillating error. Accordingly, in several PDE situations the fully implicit method of Sect. 4.2.5 behaves better than the Crank–Nicolson method [310, 386].

On Sect. 4.4

The boundary condition $V_C(0,t) = 0$ in (4.26) can be shown independently of any underlying model [269]. If European options are evaluated via the analytic formulas (A.15)–(A.17), the boundary conditions in (4.28) are of no practical interest. When boundary conditions are not clear, it sometimes helps to set $V_{SS} = 0$ (or $y_{xx} = 0$), which amounts to assume linear behavior. See [353] for a discussion, and for the effect of boundary conditions on accuracy and stability. For bounds on the error caused by truncating the infinite x- or S-interval, see [214]. Boundary conditions for a term structure equation are discussed in [117].

On Sect. 4.5

For a proof of the Black–Scholes inequality, see [237, p. 111]. The obstacle problem in this chapter is described following [376]. Also the smooth pasting argument of Exercise 4.6 is based on that work. For other arguments concerning smooth pasting see [277], and [234], where you find a discussion of $S_f(t)$, and of the behavior of this curve for $t \to T$. There are several different possibilities to implement boundary conditions at $x_{\min}$, $x_{\max}$, see [353, p. 122]. The accuracy can be improved with artificial boundary conditions [169]. For direct methods, see also [99, 194]. Front-fixing goes back to Landau 1950, see [90]. For front-fixing applications to finance, consult, for example, [188, 288, 381], and the comments on Sect. 4.7.

The general definition of a linear complementarity problem is

$$\mathsf{AB} = 0\,, \quad \mathsf{A} \geq 0\,, \quad \mathsf{B} \geq 0\,,$$

where A and B are abbreviations of more complex expressions. This can be also written

$$\min(\mathsf{A}, \mathsf{B}) = 0\,.$$

A general reference on free boundaries and on linear complementarity is [119].

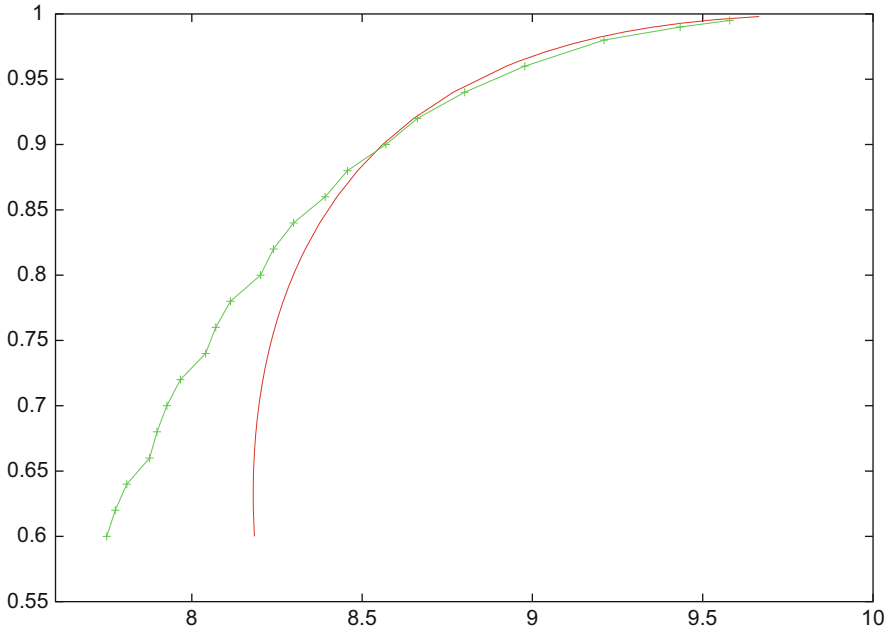

Fig. 4.20 (S, t)-plane. Approximations of an early-exercise curve of an American put ($T = 1$, $\sigma = 0.3$, $K = 10$); *green*: raw data out of an finite-difference approximation, *red*: asymptotic behavior for $t \approx T$. The asymptotic curve is valid only close to the strike K, much smaller than shown here

Figure 4.20 shows a detail of approximations to an early-exercise curve. The finite-difference calculated points are connected by straight lines. The figure also shows a local approximation valid close to maturity: For $t < T$ and $t \to T$, the asymptotic behavior of S_f can be approximated by, for example,

$$S_\mathrm{f}(t) \sim K\left(1 - \sigma\sqrt{(t-T)\log(T-t)}\right)$$

for an American put without dividends [22, 282]. For other asymptotic formulas, see [74, 75, 158]. Recall from the notes on Sect. 4.1 that discrete dividend payments change the early-exercise curve [273]; see also Appendix D.2.

For a proof of the high-contact condition or smooth-pasting principle see [277], p.114. For a discussion of the smoothness of the free boundary S_f see [282] and the references therein.

On Sect. 4.6

By choosing the θ in (4.41) one fixes at which position along the time axis the second-order spatial derivatives are focused. With

$$\theta = \frac{1}{2} - \frac{1}{12}\frac{\Delta x^2}{\Delta \tau}$$

a scheme results that is fourth-order accurate in x-direction. The application on American options requires careful compensation of the discontinuities [265]. One possibility of a variable $\Delta\tau$-time stepping is to set the nodes

$$\tau_\nu := \tau_{\max}\,\frac{\nu^2}{\nu_{\max}^2}\,,$$

suggested by [188].

Based on the experience of this author, an optimal choice of the relaxation parameter ω_{R} for the iterative variant in Algorithm 4.14 can not be given. The simple strategy $\omega_{\mathrm{R}} = 1$ appears recommendable. The method of Brennan and Schwartz has been analyzed in [201]. The formulation of Problem 4.12 reminds of the dynamic programming principle of (1.69).

On Sect. 4.7

Since the accuracy of the results is not easily guaranteed, it does seem advisable to hesitate before exposing wealth to a chance of loss or damage. After having implemented a finite-difference algorithm it may be recommendable to compare the results with those obtained by means of other algorithms.[18] The lacking smoothness of solutions near $(S,t) \approx (K,T)$ due to the nonsmooth payoff can be largely improved by solving for the difference function $V_{\mathrm{P}}^{\mathrm{Am}}(S,\tau) - V_{\mathrm{P}}^{\mathrm{Eur}}(S,\tau)$, see also Sect. 4.8.2. The lacking smoothness along the early-exercise curve can be diminished by using a front-fixing approach, which can be applied to the above difference. But one mast pay a price. Note that the nonlinearity has entered the front-fixing equation (4.86) ($\longrightarrow$ Exercise 4.8). The success of the front-fixing approach depends on whether the corresponding root-finding iteration finds a solution. Further, in our experience the lack of smoothness is only hidden and might lead to instabilities, such as oscillations in the early-exercise curve. A transformation such as $\log(S/S_{\mathrm{f}})$ does not lead to constant coefficients because one of the factors depends on the early-exercise curve. The alternative front-fixing approach of [188]

[18] As already mentioned in Sect. 4.7, the risk of having chosen an inappropriate model is mostly larger than the risk of inaccurate digits.

first applies the transformation $S = Ke^x$, $\tau = T - t$. Then the infinite (x, τ)-strip is truncated to a finite domain by the function $a(\tau) := x_f(\tau) - L$ for large enough $|L|$ ($L > 0$ for a put, $L < 0$ for a call), where $x_f(\tau) := \log(S_f(T - \tau)/K)$ denotes the transformed early-exercise curve. The final boundary-value problem localized on a rectangle is obtained by transforming the independent variable x to $z := x - a(\tau)$ (for a put). Front-fixing approaches have shown to be highly efficient.

The question how accurate different methods are has become a major concern in recent research; see for instance [83]. Clearly one compares a finite-difference European option with the analytic formulas (A.15)/(A.17). The latter are to be preferred, except the surface $V(S, t)$ is the ultimate object. The correctness of codes can be checked by testing the validity of symmetry relations (A.23).

Greeks such as delta$= \frac{\partial V}{\partial S}$ can be calculated accurately by solving specific PDEs that are derived from the Black–Scholes equation by differentiating. But delta can be approximated easily based on the a calculated approximation of V. To this end, calculate an interpolating Lagrange polynomial $L(S)$ on the line $t = 0$ based on three to five neighboring nodes (Appendix C.1), and take the derivative $L'(S)$.

We have introduced finite differences mainly in view of calculating standard American options. For exotic options PDEs occur, the solutions of which depend on three or more independent variables [21, 353, 376]; see also Chap. 6.

On Sect. 4.8

There are many analytic methods. For example, a binomial tree with a fixed number of nodes can be considered as analytic method. Classic approaches include [63, 150]. Seydel [338] suggests to analyze the attainable accuracy beforehand, depending on the parameters of options, for example, for the interpolation method. The quadratic approximation method has been extended to the more general situation of commodity options, where the cost of carry is involved [26], and a more ambitious initial guess is constructed. Integral representations are based on an inhomogeneous differential equation as that in Sect. 4.5.3. Kim's integral representation (4.76) can be derived via Mellin's transformation [294], or via Duhamel's principle [234], see also [202]. A condition number is derived by [174]. For implementations and improvements, see [175, 211]. The exponential function has been used for approximating the early-exercise curve already in [292]. There are other approaches with integral equations. From the Black–Scholes equation and the high-contact condition we recommend to derive

$$\frac{\partial V_P(S_f(t), t)}{\partial t} = 0\,.$$

This equation enables an effective construction of the the early-exercise curve [74, 75].

On Other Methods

Here we give a few hints on methods neither belonging to this chapter on finite differences, nor to Chaps. 5 or 6. General hints can be found in [321], in particular with the references of [57]. Closely related to linear complementarity problems are minimization methods. An efficient realization by means of methods of linear optimization is suggested in [98]. The uniform grid can only be the first step toward more flexible approaches, such as the finite elements to be introduced in Chap. 5. For grid stretching and coordinate transformations see [197, 240]. Spectral methods have shown to be highly efficient, consult [381]. For penalty methods we refer to [133, 288], and to Sect. 6.7. Another possibility to enhance the power of finite differences is the *multigrid* approach; for general expositions see [161, 364]; for application to finance see [81, 293]. An irregular grid based on Sobol points is suggested in [36].

4.11 Exercises

4.1 (Discrete Dividend Payment)
Assume that a stock pays one dividend D at ex-dividend date t_D, with $0 < t_D < T$.

(a) Calculate a corresponding continuous dividend rate δ under the assumptions

$$\dot{S} = -\delta S\,, \quad S(T) = S(0) - D > 0\,.$$

(b) Define for an American put with strike K

$$\tilde{t} := t_D - \frac{1}{r}\log\left(\frac{D}{K} + 1\right)\,.$$

Assume $r > 0$, $D > 0$, and a time instant t in $\tilde{t} < t < t_D$. Argue that instead of exercising early it is reasonable to wait for the dividend.
Note: For $\tilde{t} > 0$, depending on S, early exercise may be reasonable for $0 \le t < \tilde{t}$.

4.2 (Stability of the Fully Implicit Method)
The backward-difference method is defined via the solution of the Eq. (4.18)/(4.19). Prove the stability.
Hint: Use the results of Sect. 4.2.4 and $w^{(\nu)} = A^{-1}w^{(\nu-1)}$.

4.3 (Semidiscretization, Method of Lines)
For a semidiscretization of the Black–Scholes equation (1.5) consider the semidiscretized domain

$$0 \le t \le T\,, \quad S = S_i := i\Delta S\,, \quad \Delta S := \frac{S_{\max}}{m}\,, \quad i = 0, 1, \ldots, m$$

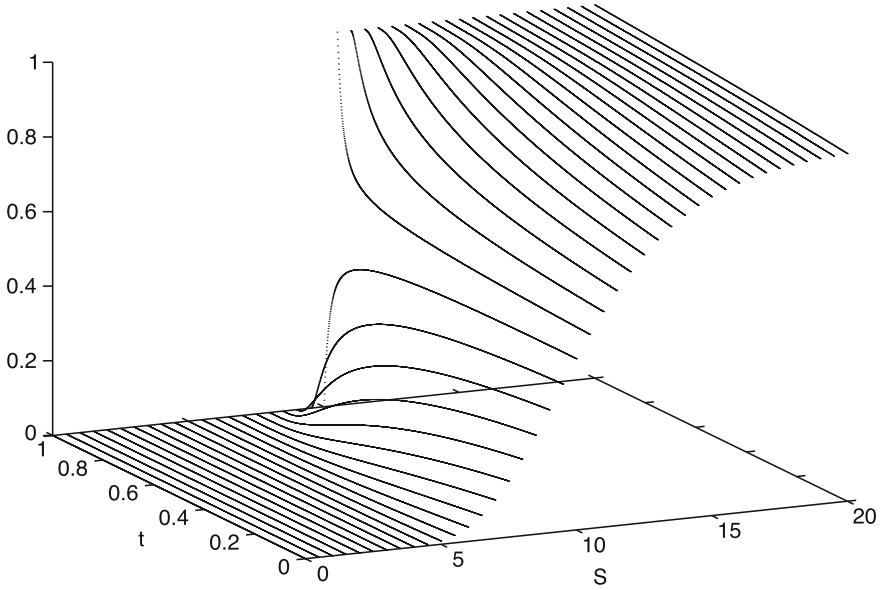

Fig. 4.21 V over (S, t); method of lines for a binary call option, compare Exercise 4.3 ($K = 10$, $T = 1$, $r = 0.06$, $\delta = 0$, $\sigma = 0.3$). With kind permission of Miriam Weingarten

for suitable values of $S_{\max} > K$ and m. On this set of lines parallel to the t-axis define for $\tau := T - t$ and $1 \le i \le m - 1$ functions $w_i(\tau)$ as approximation to $V(S_i, \tau)$.

(a) Using the standard second-order difference schemes of Sect. 4.2.1, derive the ODE system $\dot{w} = Bw$ that up to boundary conditions approximates (1.5). Here w is the vector $(w_1, \ldots, w_{m-1})^{tr}$ and $\dot{w}$ denotes differentiation w.r.t. τ . Show that B is a tridiagonal matrix, and calculate its coefficients.

(b) For a European option assume Dirichlet boundary conditions for $w_0(\tau)$ and $w_m(\tau)$ and set up a vector c such that

$$\dot{w} = Bw + c \tag{4.84}$$

realizes the ODE system with correct boundary conditions, and with initial conditions taken from the payoff.

(c) Use the BDF formula (4.11) of Sect. 4.2.1, and implement this scheme for the initial-value problem with (4.84) and a European call option. (See Fig. 4.21 for an illustration.)

4.4 (Crank–Nicolson Order)
Let the function $y(x, \tau)$ solve the equation

$$y_\tau = y_{xx}$$

and be sufficiently smooth. With the difference quotient

$$\delta_{xx} w_{i,\nu} := \frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2}$$

the local discretization error ϵ of the Crank–Nicolson method is defined

$$\epsilon := \frac{y_{i,\nu+1} - y_{i,\nu}}{\Delta\tau} - \frac{1}{2}\left(\delta_{xx} y_{i,\nu} + \delta_{xx} y_{i,\nu+1}\right) .$$

Show

$$\epsilon = O(\Delta\tau^2) + O(\Delta x^2) .$$

4.5 (Boundary Conditions of a European Call)
Show that under the transformation (4.3)

$$\begin{aligned} &S\mathrm{e}^{-\delta(T-t)} - K\mathrm{e}^{-r(T-t)} = \\ &\exp\left\{\frac{x}{2}(q_\delta + 1) + \frac{\tau}{4}(q_\delta + 1)^2\right\} - \exp\left\{\frac{x}{2}(q_\delta - 1) + \frac{\tau}{4}(q_\delta - 1)^2\right\} \end{aligned}$$

holds, and prove (4.28).
Hints: Either transform the Black–Scholes equation (4.1) with

$$S := \bar{S}\exp(\delta(T-t))$$

into a dividend-free version to obtain the dividend version of (4.27), or apply the dividend version (A.16) of the put-call parity.

4.6 (Smooth Pasting of the American Put)
Suppose a portfolio consists of an American put and the corresponding underlying. Hence the value of the portfolio is $\Pi := V_{\mathrm{P}}^{\mathrm{Am}} + S$, where S satisfies the SDE (1.47). S_{f} is the value for which we have high contact, compare (4.31).

(a) Show that

$$\mathrm{d}\Pi = \begin{cases} 0 & \text{for } S < S_{\mathrm{f}} \\ \left(\dfrac{\partial V_{\mathrm{P}}^{\mathrm{Am}}}{\partial S} + 1\right)\sigma S\,\mathrm{d}W + O(\mathrm{d}t) & \text{for } S > S_{\mathrm{f}} . \end{cases}$$

(b) Use this to argue

$$\frac{\partial V_{\mathrm{P}}^{\mathrm{Am}}}{\partial S}(S_{\mathrm{f}}(t), t) = -1 .$$

Hint: Use $\mathrm{d}S > 0 \Rightarrow \mathrm{d}W > 0$ for small $\mathrm{d}t$. Assume $\frac{\partial V}{\partial S} > -1$ and construct an arbitrage strategy for $\mathrm{d}S > 0$.

4.7 (Perpetual Put Option)
For $T \to \infty$ it is sufficient to analyze the ODE

$$\frac{\sigma^2}{2} S^2 \frac{\mathrm{d}^2 V}{\mathrm{d}S^2} + (r - \delta) S \frac{\mathrm{d}V}{\mathrm{d}S} - rV = 0 \,.$$

Consider an American put contacting the payoff $(K - S)^+$ at $S = \alpha < K$. Show:

(a) Upon substituting the boundary condition for $S \to \infty$ one obtains

$$V(S) = c \left(\frac{S}{K} \right)^{\lambda_2} \,, \tag{4.85}$$

where $\lambda_2 = \frac{1}{2} \left(1 - q_\delta - \sqrt{(q_\delta - 1)^2 + 4q} \right)$, $q = \frac{2r}{\sigma^2}$, $q_\delta = \frac{2(r-\delta)}{\sigma^2}$ and c is a positive constant. Fix c by using the left-hand boundary $V(\alpha) = K - \alpha$.
Hint: Apply the transformation $S = K\mathrm{e}^x$. (The other root λ_1 drops out.)

(b) V is decreasing and convex.

For $S < \alpha$ the option is exercised; then its intrinsic value is $K - S$. For $S > \alpha$ the option is not exercised and has a value $V(S) > K - S$. The holder of the option decides when to exercise. This means, the holder makes a decision on the contact $S = \alpha$ such that the value of the option becomes maximal [269].

(c) Show: $V'(\alpha_0) = -1$, if α_0 maximizes the value of the option.

4.8 (Front-Fixing for American Options)
Apply the transformation

$$\zeta := \frac{S}{S_\mathrm{f}(t)} \quad , \quad y(\zeta, t) := V(S, t)$$

to the Black–Scholes equation (4.1).

(a) Show

$$\frac{\partial y}{\partial t} + \frac{\sigma^2}{2} \zeta^2 \frac{\partial^2 y}{\partial \zeta^2} + \left[(r - \delta) - \frac{1}{S_\mathrm{f}} \frac{\mathrm{d}S_\mathrm{f}}{\mathrm{d}t} \right] \zeta \frac{\partial y}{\partial \zeta} - ry = 0 \,. \tag{4.86}$$

(b) Set up the domain for (ζ, t) and formulate the boundary conditions for an American call. (Assume $\delta > 0$.)

(c) (Project) Set up a finite-difference scheme to solve boundary-value problem derived above. The curve $S_\mathrm{f}(t)$ is implicitly defined by the PDE (4.86), with final value $S_\mathrm{f}(T) = \max(K, \frac{r}{\delta} K)$.

4.9 (Boundary Conditions of American Options)
Show that the boundary conditions of American options satisfy

$$\lim_{x\to\pm\infty} y(x,\tau) = \lim_{x\to\pm\infty} g(x,\tau)\,,$$

where g is defined in Problem 4.7.

4.10 (Gauss–Seidel Method as Special Case of SOR)
Let the $n \times n$ matrix $A = (a_{ij})$ be partitioned additively into $A = D - L - U$, with D diagonal matrix, L strict lower triangular matrix, U strict upper triangular matrix, $x \in \mathbb{R}^n$, $b \in \mathbb{R}^n$. The *Gauss–Seidel method* is defined by

$$(D-L)x^{(k)} = Ux^{(k-1)} + b$$

for $k = 1, 2, \ldots$. Show that with

$$r_i^{(k)} := b_i - \sum_{j=1}^{i-1} a_{ij}x_j^{(k)} - \sum_{j=i}^{n} a_{ij}x_j^{(k-1)}$$

and for $\omega_{\mathrm{R}} = 1$ the relation

$$x_i^{(k)} = x_i^{(k-1)} + \omega_{\mathrm{R}}\frac{r_i^{(k)}}{a_{ii}}$$

holds. For general $1 < \omega_{\mathrm{R}} < 2$ this defines the SOR (successive overrelaxation) method.

4.11 (Brennan–Schwartz Algorithm)
Let A be a tridiagonal matrix as in (C.6), and b and g vectors. The system of equations $Aw = b$ is to be solved such that the side condition $w \geq g$ is obeyed componentwise. Assume for the case of a put $w_i = g_i$ for $1 \leq i \leq i_{\mathrm{f}}$ and $w_i > g_i$ for $i_{\mathrm{f}} < i \leq n$, where i_{f} is unknown.

(a) Formulate an algorithm similar as Algorithm C.3 that solves $Aw = b$ in the backward/forward approach. In the final forward loop, for each i the calculated candidate $\tilde{w}_i$ is tested for $w_i \geq g_i$: Set $w_i := \max\{\tilde{w}_i,\ g_i\}$.
(b) Apply the algorithm to the case of a put with A, b, g from Sect. 4.6.1. For the case of a call adapt the forward/backward Algorithm C.3. Incorporate this approach into Algorithm 4.14.

4.12 (American Call)
Formulate the analogue of Algorithm 4.13 for the case of a call.

4.13
Implement Algorithms 4.14 and 4.15.
Test example: Example 1.6 and others.

4.14 (Approximating the Free Boundary)
Assume that after a finite-difference calculation of an American put three approximate values $V(S_i, t)$ are available, for a value of t and $i = k, k+1, k+2$. Assume further an index k such that these three (S, V)-pairs are close to the free boundary $S_f(t)$, and inside the continuation region.

(a) Derive an approximation $\bar{S}_f$ to $S_f(t)$ based on the available data.
(b) Discuss the error $O(\bar{S}_f - S_f)$.

Hints: The derivative $\frac{\partial V}{\partial S}$ at S_f is -1. For (b) assume an equidistant spacing of the S_i.

4.15 (Extrapolation of Higher Order)
Similar as in Sect. 4.7 assume an error model

$$\eta^* = \eta(\Delta) - \gamma_1 \Delta^2 - \gamma_2 \Delta^3$$

and three calculated values

$$\eta_1 := \eta(\Delta)\,, \quad \eta_2 := \eta\left(\frac{\Delta}{2}\right), \quad \eta_3 := \eta\left(\frac{\Delta}{4}\right).$$

Show that

$$\eta^* = \frac{1}{21}(\eta_1 - 12\eta_2 + 32\eta_3)\,.$$

4.16 (PDE for the Greek Delta)
Derive a PDE-boundary-value problem for the greek delta $\Delta := \frac{\partial V}{\partial S}$ in case of a plain-vanilla put.
Hint: Differentiate the Black-Scholes equation, its terminal condition, and its boundary conditions with respect to S.

4.17

(a) Derive (4.63).
(b) Derive (4.70).

4.18 (Analytic Method for the American Put)
(Project) Implement both the Algorithm 4.17 and Algorithm 4.18. For Algorithm 4.18 choose as initial guess the average of the strike and the lower bound (A.21). A secant method (C.5) is a good choice for the iteration. Think of how to combine Algorithms 4.17 and 4.18 into a hybrid algorithm.

4.19 Consider the functions d_1 and d_2 of (4.75). For the three cases $S < S_f(\tau)$, $S = S_f(\tau)$, $S > S_f(\tau)$, calculate the limit for $\xi \to 0^+$ of

$$rK\mathrm{e}^{-r\xi}\, F(-d_2(S, \xi; S_f(\tau - \xi))) - \delta S_f(\tau)\, \mathrm{e}^{-\delta\xi}\, F(-d_1(S, \xi; S_f(\tau - \xi)))\,.$$

4.20
Implement Kim's integral-equation method (Sect. 4.8.4).

4.21 (Complexity)
With n underlyings and time t an option problem comprises $n + 1$ independent variables. Assume that we discretize each of the $n + 1$ axes with M grid points, then M^{n+1} nodes are involved. Hence the *complexity* C of the n-factor model is

$$C := O(M^{n+1}) ,$$

which amounts to an exponential growth with the dimension, nicknamed *curse of dimension*. Depending on the chosen method, the error E is of the order $M^{-\ell}$,

$$E := O\left(\frac{1}{M^{\ell}}\right) .$$

Argue

$$\log C = -\frac{n+1}{\ell} \log E + \gamma$$

for a method-dependent constant γ.

Chapter 5
Finite-Element Methods

The finite-difference approach with equidistant grids is easy to understand and straightforward to implement. Resulting uniform rectangular grids are comfortable, but in many applications not flexible enough. Steep gradients of the solution require a finer grid locally such that the difference quotients provide good approximations of the differentials. On the other hand, a flat gradient may be well modeled on a coarse grid. Arranging such a flexibility of the grid with finite-difference methods is possible but cumbersome.

An alternative type of methods for solving PDEs that does provide high flexibility is the class of finite-element methods (FEM). A "finite element" designates a mathematical topic such as an interval and thereupon defined a piece of function. There are alternative names such as *variational methods*, or *weighted residuals*, or *Ritz–Galerkin methods*. These names hint at underlying principles that serve to derive suitable equations. As these different names suggest, there are several different approaches leading to finite elements. The methods are closely related.

The flexibility of finite-element methods is not only favorable to approximate functions, but also to approximate domains of computation that are not rectangular. This is important for multifactor options. For the one-dimensional situation of standard options, the possible improvement of a finite-element method over the standard methods of the previous chapter is not significant. With the focus on standard options, Chap. 5 may be skipped on first reading. But options with several underlyings may lead to domains of computation that are more "fancy."

For example, a two-asset basket with portfolio value $\alpha_1 S_1 + \alpha_2 S_2$ in the case of a call option leads to a payoff of type $\Psi(S_1, S_2) = (\alpha_1 S_1 + \alpha_2 S_2 - K)^+$. If such an option is endowed with barriers, then it is reasonable to set up barriers such that the payoff takes a constant value. For the two-asset basket, this amounts to barrier lines $\alpha_1 S_1 + \alpha_2 S_2 =$constant. This naturally leads to trapezoidal shapes of domains. For a special case with two knock-out barriers the payoff and the domain are illustrated by Fig. 5.1. This example will be considered in Sect. 5.4, see the domain in Fig. 5.5. In more complicated examples, the domain may be elliptic ($\longrightarrow$ Exercise 5.1). In

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_5

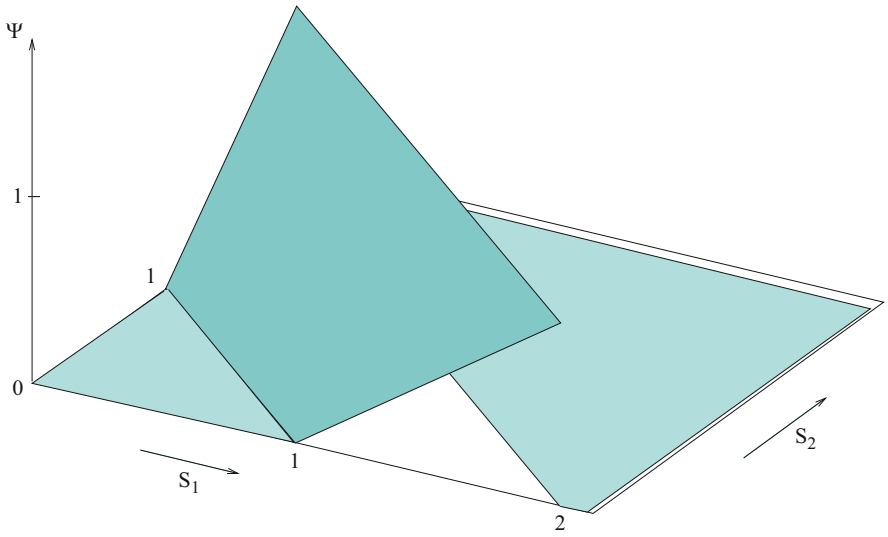

Fig. 5.1 Payoff $\Psi(S_1, S_2)$ of a call on a two-asset basket, with knock-out barrier (Example 5.6)

such situations of non-rectangular domains, finite elements are ideally applicable and highly recommendable.

Faced with the huge field of finite-element methods, in this chapter we confine ourselves to a step-by-step exposition towards the solution of two-asset options. We start with an overview on basic approaches and ideas (in Sect. 5.1). Then, in Sect. 5.2, we describe the approximation with the simplest finite elements, namely, piecewise straight-line segments, and apply this to a stationary model problem. These approaches will be applied to the time-dependent situation of pricing standard options, in Sect. 5.3. This sets the stage to the main application of FEM in financial engineering, options on two or more assets. Section 5.4 will present an application to an exotic option with two underlyings. Here we derive a weak form of the PDE, and discuss boundary conditions. Finally, in Sect. 5.5, we will introduce to error estimates. Methods more subtle than just the Taylor expansion of the discretization error are required to show that quadratic convergence is possible with unstructured grids and nonsmooth solutions. To keep the exposition of an error analysis short, we concentrate on the one-dimensional situation. But the ideas extend to multidimensional scenarios.

5.1 Weighted Residuals

Many of the principles on which finite-element methods are based, can be interpreted as weighted residuals. What does this mean? This heading points at ways in which a discretization can be set up, and how an approximation can be defined.

Fig. 5.2 Discretization of a continuum

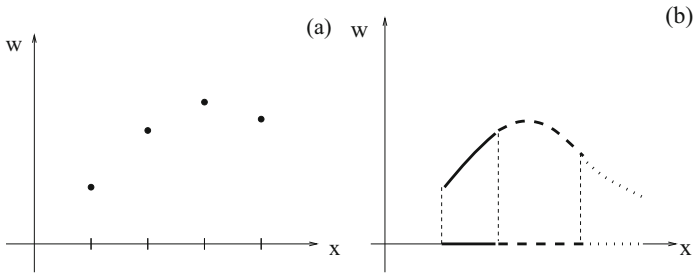

Fig. 5.3 Two kinds of approximations (one-dimensional situation)

There lies a duality in a discretization. This is illustrated by means of Fig. 5.2, which shows a partition of an x-axis. This discretization is either represented by

(a) discrete grid points x_i, or by
(b) a set of subintervals.

The two ways to see a discretization lead to different approaches of constructing an approximation w. Let us illustrate this with the one-dimensional situation of Fig. 5.3. An approximation w based on finite differences is built on the grid points and primarily consists of discrete points (Fig. 5.3a). In contrast, finite elements are founded on subdomains (intervals in Fig. 5.3b) with piecewise functions, which are defined by suitable criteria and constitute a global approximation w. In a narrower sense, a finite element is a pair consisting of one piece of subdomain and the corresponding function defined thereupon, mostly a polynomial. Figure 5.3 reflects the respective basic approaches; in a second step the isolated points of a finite-difference calculation can well be extended to continuous piecewise functions by means of interpolation (⟶ Appendix C.1).

A two-dimensional domain can be partitioned into triangles, for example, where w is again represented by piecewise polynomials. Figure 5.4 depicts the simplest such situation, namely, a triangle in an (x, y)-plane, and a piece of a linear function defined thereupon. Figure 5.5 below will provide an example how triangles easily fill a seemingly "irregular" domain.

As will be shown next, the approaches of finite-element methods use integrals. If done properly, integrals require less smoothness. This often matches applications better and adds to the flexibility of finite-element methods. The integrals can be derived in a natural way from minimum principles, or are constructed artificially. Finite elements based on polynomials make the calculation of the integrals easy.

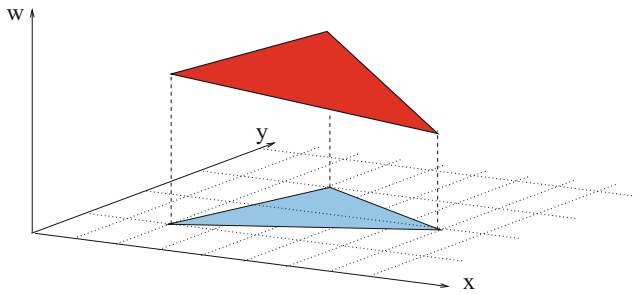

Fig. 5.4 A simple finite element in two dimensions, based on a triangle

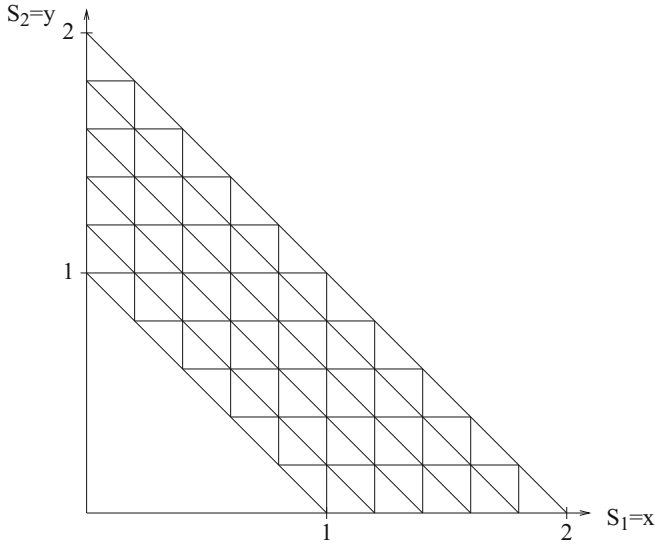

Fig. 5.5 A simple regular finite-element discretization of a domain $\mathcal{D}$ into triangles $\mathcal{D}_k$ (see Example 5.6)

5.1.1 The Principle of Weighted Residuals

To explain the principle of weighted residuals we discuss the formally simple case of the differential equation

$$Lu = f\,. \tag{5.1}$$

Here L symbolizes a linear differential operator. Important examples are

$$Lu := -u'' \text{ for } u(x), \text{ or} \tag{5.2}$$

$$Lu := -u_{xx} - u_{yy} \text{ for } u(x,y)\,. \tag{5.3}$$

The right-hand side f is a problem-dependent function. Solutions u of the differential equation (5.1) are studied on a domain $\mathcal{D} \subseteq \mathbb{R}^n$, with $n = 1$ in (5.2) and $n = 2$ in (5.3). The piecewise approach starts with a partition of the domain into a finite number m of subdomains $\mathcal{D}_k$,

$$\mathcal{D} = \bigcup_{k=1}^{m} \mathcal{D}_k \,. \tag{5.4}$$

All boundaries of $\mathcal{D}$ should be included, and approximations to u are calculated on the closure of $\mathcal{D}$. The partition is assumed disjoint up to the boundaries of $\mathcal{D}_k$, so $\mathcal{D}_j^\circ \cap \mathcal{D}_k^\circ = \emptyset$ for $j \neq k$. In the one-dimensional case ($n = 1$), for example, the $\mathcal{D}_k$ are subintervals of a whole interval $\mathcal{D}$. In the two-dimensional case, (5.4) may describe a partition into triangles, as illustrated in Fig. 5.5.

The ansatz for approximations w to a solution u is a basis representation with N basis functions φ_i,

$$w := \sum_{i=1}^{N} c_i \, \varphi_i \,. \tag{5.5}$$

The functions φ_i are also called *trial functions*. In the case of one independent variable x the $c_i \in \mathbb{R}$ are constant coefficients, and the φ_i are functions of x. Typically, N is chosen and $\varphi_1, \ldots, \varphi_N$ are prescribed. Depending on this choice, the free parameters $c_1, \ldots, c_N$ are to be determined such that $w \approx u$. The ansatz (5.5) was suggested by Ritz in 1908.

We have m subdomains and N basis functions. In the one-dimensional situation ($n = 1$), nodes and subintervals interlace, and m and N essentially can be identified. For $n = 1$ the two numbers m and N differ by at most one, depending on whether the solution is known or unknown at the end points of the interval $\mathcal{D}$. In the latter case it is convenient to have the summation index in (5.5) run as $i = 0, \ldots, m$. For dimensions $n > 1$ the number m of subdomains (e.g. triangles in case $n = 2$) in general is different from the number N of basis functions (nodes[1]). For example, in Fig. 5.5 we have 75 triangles and 51 nodes; 26 of the nodes are interior nodes and 25 are placed along the boundary. That is, $1 \leq k \leq 75$. The number N refers to the number of nodes for which a value of u is to be approximated.

One strategy to determine the coefficients c_i is based on the residual function

$$R(w) := Lw - f \,. \tag{5.6}$$

We look for a w such that the residual R becomes "small." Since the φ_i are considered prescribed, in view of (5.5) N conditions or equations must be established to define

[1]Basis functions can be constructed such that there is one for each node. Then N represents also the number of nodes.

and calculate the unknown $c_1, \ldots, c_N$. To this end we weight the residual R by introducing N weighting functions *(test functions)* $\psi_1, \ldots, \psi_N$ and require

$$\int_{\mathcal{D}} R(w)\, \psi_j \, \mathrm{d}\mathcal{D} = 0 \quad \text{for } j = 1, \ldots, N\,. \tag{5.7}$$

This amounts to the requirement that the residual be orthogonal to the set of weighting functions ψ_j. The "d$\mathcal{D}$" in (5.7) symbolizes the integration that matches $\mathcal{D} \subseteq \mathbb{R}^n$, as d$x$ for $n = 1$. For ease of notation, we frequently drop dx as well as the $\mathcal{D}$ at the n-dimensional integral. For the model problem (5.1) the system of Eqs. (5.7) consists of the N equations

$$\int_{\mathcal{D}} Lw\, \psi_j = \int_{\mathcal{D}} f\, \psi_j \quad (j = 1, \ldots, N) \tag{5.8}$$

for the N unknowns $c_1, \ldots, c_N$, which define w. Often the equations in (5.8) are written using a formulation with inner products,

$$(Lw, \psi_j) = (f, \psi_j)\,,$$

defined as the corresponding integrals in (5.8). For linear L the ansatz (5.5) implies

$$\int Lw\, \psi_j = \int \left(\sum_i c_i L\varphi_i \right) \psi_j = \sum_i c_i \underbrace{\int L\varphi_i \psi_j}_{=:a_{ij}}\,.$$

The integrals a_{ij} constitute a matrix A. The $r_j := \int f\psi_j$ set up the elements of a vector r and the coefficients c_j a vector $c = (c_1, \ldots, c_N)^{\mathrm{tr}}$. In vector notation the system of equations is rewritten as

$$Ac = r\,. \tag{5.9}$$

This outlines the general principle, but leaves open the questions how to handle boundary conditions and how to select basis functions φ_i and weighting functions ψ_j. The freedom to choose trial functions φ_i and test functions ψ_j allows to construct several different methods. For the time being suppose that these functions have sufficient potential to be differentiated or integrated. We will enter a discussion of relevant function spaces in Sect. 5.5.

5.1.2 Examples of Weighting Functions

We postpone the choice of basis functions φ_i and begin with listing important examples of how to select weighting functions ψ:

1.) **Galerkin's choice:**
Choose $\psi_j := \varphi_j$ for all j. Then $a_{ij} = \int L\varphi_i \varphi_j$.

2.) **Collocation:**
Choose $\psi_j := \delta(x - x_j)$. Here δ denotes Dirac's delta function, which in $\mathbb{R}^1$ satisfies $\int f\delta(x - x_j)\,\mathrm{d}x = f(x_j)$. As a consequence,

$$\int Lw\,\psi_j = Lw(x_j)\,, \quad \int f\psi_j = f(x_j)\,.$$

That is, a system of equations $Lw(x_j) = f(x_j)$ results, which amounts to evaluating the differential equation at selected points x_j.

3.) **Least squares:**
Choose

$$\psi_j := \frac{\partial R}{\partial c_j}\,.$$

This choice of test functions deserves its name *least-squares*, because to minimize $\int (R(c_1, \ldots, c_N))^2$ the necessary criterion is the vanishing of the gradient, so

$$\int_{\mathcal{D}} R\frac{\partial R}{\partial c_j} = 0 \quad \text{for all } j\,.$$

5.1.3 Examples of Basis Functions

The construction of suitable basis functions φ_i observes the underlying partition into subdomains $\mathcal{D}_k$. Our concern will be to meet two aims: resulting methods must be accurate, and their implementation should become efficient.

The efficiency can be focused on the sparsity of matrices. In particular, if the matrix A of the linear equations is sparse, then the system can be solved efficiently even when it is large. In order to achieve sparsity we require that $\varphi_i \equiv 0$ on most of the subdomains $\mathcal{D}_k$. Figure 5.6 illustrates an example for the one-dimensional

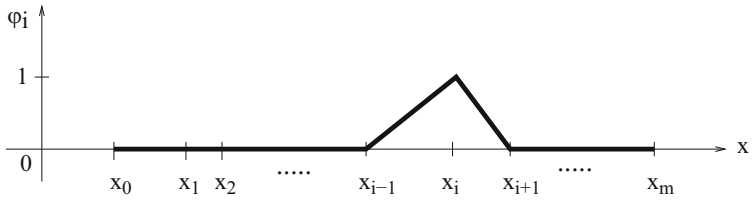

Fig. 5.6 "Hat function": simple choice of finite elements

case $n = 1$. This *hat function* of Fig. 5.6 is the simplest example related to finite elements. It is piecewise linear, and each function φ_i has a support consisting of only two subintervals, $\varphi_i(x) \neq 0$ for $x \in$ support. A consequence is

$$\int_{\mathcal{D}} \varphi_i \varphi_j = 0 \quad \text{for } |i-j| > 1\,, \tag{5.10}$$

as well as an analogous relation for $\int \varphi_i' \varphi_j'$. We will discuss hat functions in the following Sect. 5.2. Basis functions more advanced than the canonical hat functions are constructed using piecewise polynomials of higher degree. In this way, basis functions can be obtained with $\mathcal{C}^1$- or $\mathcal{C}^2$-smoothness (⟶ Exercise 5.2). Recall from interpolation (⟶ Appendix C.1) that polynomials of degree three can lead to $\mathcal{C}^2$-smooth splines.

5.1.4 Smoothness

We have left open how close an approximation w of (5.5)/(5.9) is to the solution u of (5.1). Clearly, $R(u) = 0$ and u satisfies (5.7). But w in general does not solve (5.1). The differential equation (5.1) is a stronger requirement than the integral relations (5.7).

The accuracy depends on the smoothness of the basis functions. Depending on the chosen method, different kinds of smoothness are relevant. Let us illustrate this matter on the model problem (5.2),

$$Lu = -u'', \quad \text{with } u, \varphi, \psi \in \{\, u \mid u(0) = u(1) = 0 \,\}\,.$$

Integration by parts formally implies

$$\int_0^1 \varphi'' \psi = -\int_0^1 \varphi' \psi' = \int_0^1 \varphi \psi''\,,$$

because the boundary conditions $u(0) = u(1) = 0$ let the nonintegral terms vanish. These three versions of the integral can be distinguished by the smoothness requirements on φ and ψ, and by the question whether the integrals exist. One will choose the integral version that corresponds to the underlying method, and to the smoothness of the solution. For example, for Galerkin's approach the elements a_{ij} of A consist of the integrals

$$-\int_0^1 \varphi_i' \varphi_j'\,.$$

We will return to the topics of accuracy, convergence, and function spaces in Sect. 5.5 (with Appendix C.3).

5.2 Ritz–Galerkin Method with One-Dimensional Hat Functions

As mentioned before, any required flexibility is provided by finite-element methods. This holds to a larger extent in higher-dimensional spaces. In this section, for simplicity, we stick to the one-dimensional situation, $x \in \mathbb{R}$. The dependence on the time variable t will be postponed to Sect. 5.3.

Assume a partition of the x-domain by a set of increasing mesh points $x_0, \ldots, x_m$. A nonuniform spacing is advisable in several instances in order to improve the accuracy. For example, close to the strike, a denser grid is appropriate to mollify the lack of smoothness of a payoff. In contrast, to model infinity, one rarefies the nodes for larger x and shifts the final node x_m to a large value. One strategy is to select a spacing such that locally (up to additional scaling and shifts) $\sinh(x_i) = \eta_i$, where η_i are chosen equidistantly. A dense spacing is also advisable for barrier options close to the barrier, where the gradient of option prices is high.

5.2.1 Hat Functions

The prototype of a finite-element method makes use of the hat functions, which we define formally (compare Figs. 5.6 and 5.7).

Definition 5.1 (Hat Functions) For $1 \le i \le m-1$ set $\varphi_i(x) := 0$ on all subintervals except two:

$$\varphi_i(x) := \frac{x - x_{i-1}}{x_i - x_{i-1}} \quad \text{for } x_{i-1} \le x < x_i\,,$$

$$\varphi_i(x) := \frac{x_{i+1} - x}{x_{i+1} - x_i} \quad \text{for } x_i \le x < x_{i+1}\,,$$

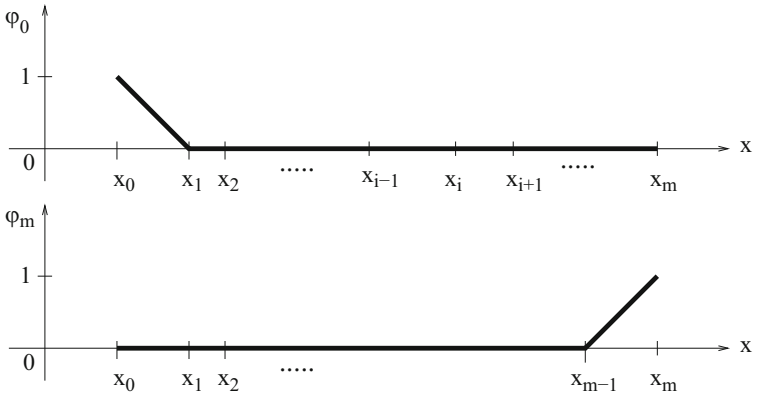

Fig. 5.7 Special "hat functions" φ_0 and φ_m

and boundary functions φ_0, φ_m nonzero on just one subinterval:

$$\varphi_0(x) := \frac{x_1 - x}{x_1 - x_0} \quad \text{for } x_0 \le x < x_1 \,,$$
$$\varphi_m(x) := \frac{x - x_{m-1}}{x_m - x_{m-1}} \quad \text{for } x_{m-1} \le x \le x_m \,.$$

For each node x_i there is one hat function. These $m + 1$ hat functions satisfy the following properties.

Properties 5.2 (Hat Functions) *The following properties (a)–(e) hold:*

(a) The $\varphi_0, \ldots, \varphi_m$ form a basis of the space of polygons

$$\{\, g \in \mathcal{C}^0[x_0, x_m] \;\mid\; g \text{ straight line on } \mathcal{D}_k := [x_k, x_{k+1}] \,, \text{ for all } k = 0, \ldots, m-1 \,\} \,.$$

That is to say, for each polygon v on the union of $\mathcal{D}_0, \ldots, \mathcal{D}_{m-1}$ there are unique coefficients $c_0, \ldots, c_m$ such that

$$v = \sum_{i=0}^{m} c_i \varphi_i \,.$$

(b) On any $\mathcal{D}_k$ only φ_k and $\varphi_{k+1} \neq 0$ are nonzero. Hence

$$\varphi_i \varphi_j = 0 \;\text{ for }\; |i - j| > 1 \,,$$

which explains (5.10).

(c) A simple approximation of the integral $\int_{x_0}^{x_m} f \varphi_j \, \mathrm{d}x$ can be calculated as follows: Substitute f by the interpolating polygon

$$f_{\mathrm{p}} := \sum_{i=0}^{m} f_i \varphi_i \quad \text{, where} \quad f_i := f(x_i) \,,$$

and obtain for each j the approximating integral

$$I_j := \int_{x_0}^{x_m} f_{\mathrm{p}} \varphi_j \, \mathrm{d}x = \int_{x_0}^{x_m} \sum_{i=0}^{m} f_i \varphi_i \varphi_j \, \mathrm{d}x = \sum_{i=0}^{m} f_i \underbrace{\int_{x_0}^{x_m} \varphi_i \varphi_j \, \mathrm{d}x}_{=: b_{ij}} \,.$$

The b_{ij} constitute a symmetric matrix B and the f_i a vector $\bar{f}$. If we arrange all integrals $I_j\,(0 \le j \le m)$ into a vector, then all integrals can be written in a compact way in vector notation as

$$B\bar{f}\,.$$

This will approximate the vector r in (5.9).

(d) The "large" $(m+1)^2$–matrix $B := (b_{ij})$ can be set up $\mathcal{D}_k$-elementwise by (2×2)-matrices (discussed below in Sect. 5.2.2). The (2×2)-matrices are those integrals that integrate only over a single subdomain $\mathcal{D}_k$. For each $\mathcal{D}_k$ in our one-dimensional setting exactly the four integrals $\int \varphi_i \varphi_j \mathrm{d}x$ for $i,j \in \{k, k+1\}$ are nonzero. They can be arranged into a (2×2)-matrix

$$\int_{x_k}^{x_{k+1}} \begin{pmatrix} \varphi_k^2 & \varphi_k\varphi_{k+1} \\ \varphi_{k+1}\varphi_k & \varphi_{k+1}^2 \end{pmatrix} \mathrm{d}x\,.$$

(The integral over a matrix is understood elementwise.) These are the integrals on $\mathcal{D}_k$, where the integrand is a product of the factors

$$\frac{x_{k+1}-x}{x_{k+1}-x_k} \quad \text{and} \quad \frac{x-x_k}{x_{k+1}-x_k}\,.$$

The four numbers

$$\frac{1}{(x_{k+1}-x_k)^2} \int_{x_k}^{x_{k+1}} \begin{pmatrix} (x_{k+1}-x)^2 & (x_{k+1}-x)(x-x_k) \\ (x-x_k)(x_{k+1}-x) & (x-x_k)^2 \end{pmatrix} \mathrm{d}x$$

result. With $h_k := x_{k+1} - x_k$ integration yields the *element-mass matrix* ($\longrightarrow$ Exercise 5.3)

$$\frac{1}{6} h_k \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix}\,.$$

(e) Analogously, integrating $\varphi_i' \varphi_j'$ yields

$$\begin{aligned} &\int_{x_k}^{x_{k+1}} \begin{pmatrix} \varphi_k'^2 & \varphi_k'\varphi_{k+1}' \\ \varphi_{k+1}'\varphi_k' & \varphi_{k+1}'^2 \end{pmatrix} \mathrm{d}x \\ &= \frac{1}{h_k^2} \int_{x_k}^{x_{k+1}} \begin{pmatrix} (-1)^2 & (-1)1 \\ 1(-1) & 1^2 \end{pmatrix} \mathrm{d}x = \frac{1}{h_k} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}\,. \end{aligned}$$

These matrices are called *element-stiffness matrices*. They are used to set up the matrix A.

5.2.2 Assembling

The next step is to assemble the matrices A and B. It might be tempting to organize this task as follows: run a double loop on all basis indices i, j (N node indices) and check for each (i, j) on which $\mathcal{D}_k$ the integral

$$\int_{\mathcal{D}_k} \varphi_i \varphi_j$$

is nonzero. Such a procedure of performing a double loop has the complexity of $O(N^2 m)$. This is cumbersome as compared to the alternative of running a single loop on the subdomain index k and benefit from all relevant integrals on $\mathcal{D}_k$, which are precalculated above (Fig. 5.8).

To this end, split the integrals

$$\int_{x_0}^{x_m} = \sum_{k=0}^{m-1} \int_{\mathcal{D}_k}$$

to construct the $(m+1) \times (m+1)$-matrices $A = (a_{ij})$ and $B = (b_{ij})$ *additively* out of the small element matrices. For the case of the one-dimensional hat functions with

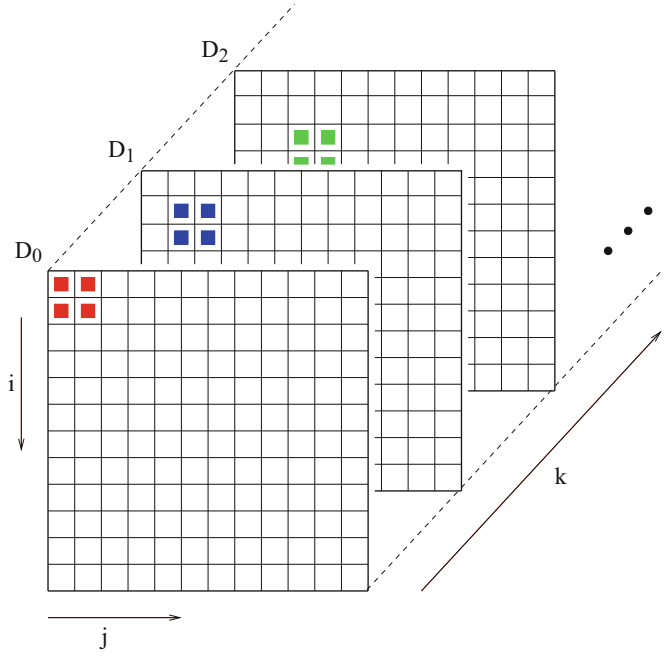

Fig. 5.8 Assembling in the one-dimensional setting

subintervals

$$\mathcal{D}_k = \{\, x \mid x_k \le x \le x_{k+1} \,\}$$

the element matrices are (2×2), see above. In this case only those integrals of $\varphi_i' \varphi_j'$ and $\varphi_i \varphi_j$ are nonzero, for which $i, j \in \mathcal{I}_k$, where

$$i, j \in \mathcal{I}_k := \{k, k+1\}\,. \tag{5.11}$$

$\mathcal{I}_k$ is the set of indices of those products of basis functions that are nonzero on $\mathcal{D}_k$. The *assembling algorithm* performs a loop over the subdomain index $k = 0, 1, \ldots, m-1$ and distributes the (2×2)-element matrices additively to the positions $i, j \in \mathcal{I}_k$. Before the assembling is started, the matrices A and B must be initialized with zeros. For $k = 0, \ldots, m-1$ one obtains for A the $(m+1)^2$-matrix

$$A = \begin{pmatrix} \frac{1}{h_0} & -\frac{1}{h_0} & & & \\ -\frac{1}{h_0} & \frac{1}{h_0} + \frac{1}{h_1} & -\frac{1}{h_1} & & \\ & -\frac{1}{h_1} & \frac{1}{h_1} + \frac{1}{h_2} & -\frac{1}{h_2} & \\ & & -\frac{1}{h_2} & \ddots & \ddots \\ & & & \ddots & \end{pmatrix}. \tag{5.12}$$

The matrix B is assembled in an analogous way. In the one-dimensional situation the matrices are tridiagonal. For an equidistant grid with $h = h_k$ the matrix A specializes to

$$A = \frac{1}{h} \begin{pmatrix} 1 & -1 & & & 0 \\ -1 & 2 & -1 & & \\ & -1 & 2 & \ddots & \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & 2 & -1 \\ 0 & & & & -1 & 1 \end{pmatrix} \tag{5.13}$$

and B to

$$B = \frac{h}{6} \begin{pmatrix} 2 & 1 & & & 0 \\ 1 & 4 & 1 & & \\ & 1 & 4 & \ddots & \\ & & \ddots & \ddots & \ddots \\ & & & \ddots & 4 & 1 \\ 0 & & & & 1 & 2 \end{pmatrix}. \tag{5.14}$$

5.2.3 A Simple Application

In order to demonstrate the procedure, let us consider the simple time-independent ("stationary") model boundary-value problem

$$Lu := -u'' = f \quad \text{with} \quad u(x_0) = u(x_m) = 0\,. \tag{5.15}$$

Substituting $w := \sum_{i=0}^{m} c_i\varphi_i$ into the differential equation, in view of (5.8), leads to

$$\sum_{i=0}^{m} c_i \int_{x_0}^{x_m} L\varphi_i\,\varphi_j\,\mathrm{d}x = \int_{x_0}^{x_m} f\varphi_j\,\mathrm{d}x\,.$$

This is the result of the Ritz–Galerkin approach. Next we apply integration by parts on the left-hand side, and invoke Property 5.2(c) on the right-hand side. The resulting system of equations is

$$\sum_{i=0}^{m} c_i \underbrace{\int_{x_0}^{x_m} \varphi_i'\varphi_j'\,\mathrm{d}x}_{a_{ij}} = \sum_{i=0}^{m} f_i \underbrace{\int_{x_0}^{x_m} \varphi_i\varphi_j\,\mathrm{d}x}_{b_{ij}}, \quad j = 0, 1, \ldots, m\,. \tag{5.16}$$

This system is preliminary because the homogeneous boundary conditions $u(x_0) = u(x_m) = 0$ are not yet taken into account.

At this state, the preliminary system of Eqs. (5.16) can be written as

$$Ac = B\bar{f}\,. \tag{5.17}$$

It is easy to see that the matrix A from (5.13) is singular, because

$$A(1, 1, \ldots, 1)^{tr} = 0\,.$$

The singularity reflects the fact that the system (5.17) does not have a unique solution. This is consistent with the differential equation $-u'' = f(x)$: If $u(x)$ is solution, then also $u(x) + \alpha$ for arbitrary α. Unique solvability is attained by satisfying the boundary conditions; a solution u of $-u'' = f$ must be fixed by at least one essential boundary condition. For our example (5.15) we know in view of $u(x_0) = u(x_m) = 0$ the coefficients $c_0 = c_m = 0$. This information can be inserted into the system of equations in such a way that the matrix A changes to a nonsingular matrix without losing symmetry. To this end, cancel the first and the last of the $n + 1$ equations in (5.17), and make use of $c_0 = c_m = 0$. Now the inner part of size $(m - 1) \times (m - 1)$ of A remains. The matrix B is $(m - 1) \times (m + 1)$.

Finally, for the special case of an equidistant grid, the system of equations is

$$\begin{pmatrix} 2 & -1 & & & 0 \\ -1 & 2 & \ddots & & \\ & \ddots & \ddots & \ddots & \\ & & \ddots & 2 & -1 \\ 0 & & & -1 & 2 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_{m-2} \\ c_{m-1} \end{pmatrix} = \frac{h^2}{6} \begin{pmatrix} 1 & 4 & 1 & & & 0 \\ & 1 & 4 & 1 & & \\ & & \ddots & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ 0 & & & & 1 & 4 & 1 \end{pmatrix} \begin{pmatrix} \bar{f}_0 \\ \bar{f}_1 \\ \vdots \\ \bar{f}_{m-1} \\ \bar{f}_m \end{pmatrix}. \tag{5.18}$$

In (5.18) we have used an equidistant grid for sake of a lucid exposition. Our main focus is the nonequidistant version, which is also implemented easily. In case nonhomogeneous boundary conditions are prescribed, appropriate values of c_0 or c_m are predefined. The importance of finite-element methods in structural engineering has lead to call the global matrix A the stiffness matrix, and B is called the mass matrix.

5.3 Application to Standard Options

Finite elements are especially advantageous in higher-dimensional spaces (several underlyings). But it also works for the one-dimensional case of standard options. This is the theme of this section. In contrast to the previous section, time must be included.

5.3.1 European Options

We know that the valuation of single-asset European options with vanilla payoff makes use of the Black–Scholes formula. But for the sake of exposition, and for non-vanilla payoff, let us briefly sketch a finite-element approach. Here we apply the FEM approach to the transformed version $y_\tau = y_{xx}$ of the Black–Scholes equation with constant parameters. In view of the general basis representation in (5.5) one may think of starting from $w = \sum w_i \varphi_i(x, \tau)$ with constant coefficients w_i. This would require two-dimensional basis functions. (We shall come back to such functions in Sect. 5.4.) To make use of one-dimensional hat functions, apply a separation ansatz in the form $\sum w_i(\tau)\varphi_i(x)$ with functions $w_i(\tau)$. As a consequence

of this simple approach, the same x-grid is applied for all τ, which results in a rectangular grid in the (x, τ)-plane. Dirichlet boundary conditions

$$y(x_{\min}, \tau) = \alpha(\tau),\ y(x_{\max}, \tau) = \beta(\tau)$$

mean that in view of the shape of φ_0, φ_m (Definition 5.1, Fig. 5.7) the values $w_0 = \alpha$ or $w_m = \beta$ would be known. It is practical to separate known terms and restrict the sum to the terms with unknown weights w_i. This can be managed by introducing a special function φ_b that compensates for Dirichlet boundary conditions on y. The function $\varphi_\mathrm{b}(x, \tau)$ is no basis function, and is constructed in advance. For example,

$$\varphi_\mathrm{b}(x, \tau) := (\beta(\tau) - \alpha(\tau)) \frac{x - x_{\min}}{x_{\max} - x_{\min}} + \alpha(\tau)$$

does the job for the above boundary conditions. So φ_b can be considered to be known, and the sum $\sum w_i \varphi_i$ does not reflect any nonzero Dirichlet boundary conditions on y. Then the final ansatz is

$$\sum_i w_i(\tau)\varphi_i(x) + \varphi_\mathrm{b}(x, \tau)\,, \tag{5.19}$$

and the index i counts those nodes x_i for which no boundary conditions of the above type are prescribed, $1 \le i \le m - 1$ in case two Dirichlet boundary conditions are given. The basis functions $\varphi_1, \ldots, \varphi_N$ are chosen to be the hat functions, which incorporate the discretization of the x-axis. Hence, $N = m-1$, and x_0 corresponds to $x_{\min}$, and x_m to $x_{\max}$. The functions $w_1, \ldots, w_{m-1}$ are unknown, and $w_0 = w_m = 0$.

Calculating derivatives of (5.19) and substituting into $y_\tau = y_{xx}$ leads to the Ritz–Galerkin approach

$$\int_{x_0}^{x_m} \left[\sum_{i=1}^{m-1} \dot{w}_i \varphi_i + \dot{\varphi}_\mathrm{b} \right] \varphi_j \,\mathrm{d}x = \int_{x_0}^{x_m} \left[\sum_{i=1}^{m-1} w_i \varphi_i'' + \varphi_\mathrm{b}'' \right] \varphi_j \,\mathrm{d}x$$

for $j = 1, \ldots, m - 1$. The overdot represents differentiation with respect to τ, and the prime with respect to x. Arranging the terms that involve derivatives of φ_b into vectors $a(\tau)$, $b(\tau)$,

$$a(\tau) := \begin{pmatrix} \int \varphi_\mathrm{b}''(x, \tau)\, \varphi_1(x)\,\mathrm{d}x \\ \vdots \\ \int \varphi_\mathrm{b}''(x, \tau)\, \varphi_{m-1}(x)\,\mathrm{d}x \end{pmatrix}, \quad b(\tau) := \begin{pmatrix} \int \dot{\varphi}_\mathrm{b}(x, \tau)\, \varphi_1(x)\,\mathrm{d}x \\ \vdots \\ \int \dot{\varphi}_\mathrm{b}(x, \tau)\, \varphi_{m-1}(x)\,\mathrm{d}x \end{pmatrix},$$

and using the matrices A, B as in (5.13)/(5.14), we arrive after integration by parts at

$$B\dot{w} + b = -Aw - a\,. \tag{5.20}$$

Note that for the specific φ_b from above $\varphi_b'' = 0$ and $a = 0$. For vanilla options, α and β can be drawn from (4.28), and b can be set up analytically; a and b can be considered as known. This completes the semidiscretization. Time τ is still continuous, and (5.20) defines the unknown vector function $w(\tau) := (w_1(\tau), \ldots, w_{m-1}(\tau))^{tr}$ as solution of a system of ordinary differential equations. This is a method of lines approach. The lines are defined by $x = x_i$ for $1 \le i \le m-1$, and the approximations along the lines are given by $w_i(\tau)$.

Initial conditions for $\tau = 0$ are derived from (5.19). Assume the initial condition from the payoff as $y(x, 0) = \gamma(x)$, then

$$\sum_{i=1}^{N} w_i(0)\varphi_i(x) + \varphi_b(x, 0) = \gamma(x) .$$

For vanilla payoff, γ is given by (4.5)/(4.6). Specifically for $x = x_j$ the sum reduces to $w_j(0) \cdot 1$, leading to

$$w_j(0) = \gamma(x_j) - \varphi_b(x_j, 0) .$$

To complete the discretization, time τ must be discretized. Standard software for ODEs can be applied to (5.20), in particular, codes for stiff systems. For discretizing with difference quotients consult Sect. 4.2.1. For example, apply the ODE trapezoidal rule as in (4.20) for the discretization of $\dot{w}$ in (5.20). We leave the derivation of the resulting Crank–Nicolson type discretization as an exercise to the reader. With the usual notation of the vector $w^{(\nu)}$ approximating $w(\tau_\nu)$, the result can be written

$$\begin{aligned} (B + \tfrac{\Delta\tau}{2}A)\, w^{(\nu+1)} &= (B - \tfrac{\Delta\tau}{2}A)\, w^{(\nu)} \\ &\quad - \tfrac{\Delta\tau}{2}\left(a^{(\nu)} + a^{(\nu+1)} + b^{(\nu)} + b^{(\nu+1)}\right) . \end{aligned} \tag{5.21}$$

The structure of (5.21) strongly resembles the finite-difference approach (4.24). This similarity suggests that the order is the same, because for the finite-element A's and B's we have (compare (5.13)/(5.14))

$$A = O\left(\frac{1}{\Delta x}\right) , \quad B = O(\Delta x) .$$

The separation of the variables x and τ in (5.19) allows to investigate the orders of the discretizations separately. In $\Delta\tau$, the order $O(\Delta\tau^2)$ of the Crank–Nicolson type approach (5.21) is clear from the ODE trapezoidal rule. It remains to derive the order of convergence with respect to the discretization in x. Because of the separation of variables it is sufficient to derive the convergence for a one-dimensional model problem. This will be done in Sect. 5.5.

5.3.2 Variational Form of the Obstacle Problem

To warm up for the discussion of the American option case, let us return to the simple obstacle problem of Sect. 4.5.5 with the obstacle function $g(x)$, or $g(x,\tau)$. This problem can be formulated as a variational inequality. The function u solving the obstacle problem can be characterized by comparing it to functions v out of a set $\mathcal{K}$ of *competing functions*

$$\mathcal{K} := \{\ v \in \mathcal{C}^0[-1,1] \mid\ v(-1) = v(1) = 0\,,$$
$$v(x) \geq g(x)\ \text{ for }\ -1 \leq x \leq 1,\ v \text{ piecewise } \in \mathcal{C}^1\}\,.$$

The requirements on u imply $u \in \mathcal{K}$. For $v \in \mathcal{K}$ we have $v - g \geq 0$ and in view of $-u'' \geq 0$ also $-u''(v-g) \geq 0$. Hence for all $v \in \mathcal{K}$ the inequality

$$\int_{-1}^{1} -u''(v-g)\,\mathrm{d}x \geq 0$$

must hold. By the LCP formulation (4.39) the integral

$$\int_{-1}^{1} -u''(u-g)\,\mathrm{d}x = 0$$

vanishes. Subtracting yields

$$\int_{-1}^{1} -u''(v-u)\,\mathrm{d}x \geq 0\ \text{ for any } v \in \mathcal{K}\,.$$

The obstacle function g does not occur explicitly in this formulation; the obstacle is implicitly defined in $\mathcal{K}$. Integration by parts leads to

$$\underbrace{[-u'(v-u)]_{-1}^{1}}_{=0} + \int_{-1}^{1} u'(v-u)'\,\mathrm{d}x \geq 0\,.$$

The integral-free term vanishes because of $u(-1) = v(-1)$, $u(1) = v(1)$. In summary, we have derived the statement:

$$\text{If } u \text{ solves the obstacle problem (4.39), then}\quad \int_{-1}^{1} u'(v-u)'\,\mathrm{d}x \geq 0 \quad \text{for all } v \in \mathcal{K}\,. \tag{5.22}$$

Since v varies in the set $\mathcal{K}$ of competing functions, an inequality such as in (5.22) is called *variational inequality*. The characterization of u by (5.22) can be used to construct an approximation w: Instead of u, find a $w \in \mathcal{K}$ such that the

inequality (5.22) is satisfied for all $v \in \mathcal{K}$,

$$\int_{-1}^{1} w'(v-w)' \, \mathrm{d}x \geq 0 \quad \text{for all } v \in \mathcal{K} \, .$$

The characterization (5.22) is related to a minimum problem, because the integral vanishes for $v = u$.

5.3.3 *Variational Form of an American Option*

Analogously as the simple obstacle problem also the problem of calculating American options can be formulated as variational problem, compare Problem 4.7. The class of competing functions must be redefined as

$$\begin{aligned} \mathcal{K} := \{ \; & v \in \mathcal{C}^0[x_{\min}, x_{\max}] \;\mid\; \tfrac{\partial v}{\partial x} \text{ piecewise } \mathcal{C}^0 \, , \\ & v(x,\tau) \geq g(x,\tau) \;\text{ for all } x, \tau \; , \; v(x,0) = g(x,0) \, , \\ & v(x_{\max},\tau) = g(x_{\max},\tau), \; v(x_{\min},\tau) = g(x_{\min},\tau) \, \} \, . \end{aligned} \tag{5.23}$$

For the following, $v \in \mathcal{K}$ for the $\mathcal{K}$ from (5.23). Let y denote the exact solution of Problem 4.7. As solution of the partial differential inequality, y is $\mathcal{C}^2$-smooth on the continuation region, and $y \in \mathcal{K}$. From

$$v \geq g, \quad \frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \geq 0$$

we deduce

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \right) (v-g) \, \mathrm{d}x \geq 0 \, .$$

Invoking the complementarity

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \right) (y-g) \, \mathrm{d}x = 0$$

and subtraction gives

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y}{\partial \tau} - \frac{\partial^2 y}{\partial x^2} \right) (v-y) \, \mathrm{d}x \geq 0 \, .$$

Integration by parts leads to the inequality

$$\int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y}{\partial \tau}(v-y) + \frac{\partial y}{\partial x}\left(\frac{\partial v}{\partial x} - \frac{\partial y}{\partial x} \right) \right) \mathrm{d}x - \frac{\partial y}{\partial x}(v-y) \Bigg|_{x_{\min}}^{x_{\max}} \geq 0 \,.$$

The nonintegral term vanishes, because at the boundary for $x_{\min}$, $x_{\max}$, in view of $v = g$, $y = g$, the equality $v = y$ holds. The final result is

$$I(y;v) := \int_{x_{\min}}^{x_{\max}} \left(\frac{\partial y}{\partial \tau} \cdot (v-y) + \frac{\partial y}{\partial x}\left(\frac{\partial v}{\partial x} - \frac{\partial y}{\partial x} \right) \right) \mathrm{d}x \geq 0 \quad \text{for all } v \in \mathcal{K} \,. \tag{5.24}$$

The exact y is characterized by the fact that the inequality (5.24) holds for all comparison functions $v \in \mathcal{K}$. For the special choice $v = y$ the integral takes its minimal value,

$$\min_{v \in \mathcal{K}} I(y;v) = I(y;y) = 0 \,.$$

A more general question is, whether the inequality (5.24) holds for a $\widehat{y} \in \mathcal{K}$ that is not $\mathcal{C}^2$-smooth on the continuation region.[2] The aim is:

Problem 5.3 (Weak Version) *Construct a $\widehat{y} \in \mathcal{K}$ such that $I(\widehat{y}; v) \geq 0$ for all $v \in \mathcal{K}$.*

This formulation of our problem is called *weak version*, because it does *not* use $\widehat{y} \in \mathcal{C}^2$. Solutions $\widehat{y}$ of Problem 5.3, which are globally continuous but only piecewise $\in \mathcal{C}^1$, are called *weak solutions*. The original partial differential equation requires $y \in \mathcal{C}^2$ and hence more smoothness. Such $\mathcal{C}^2$-solutions are called *strong solutions* or *classical solutions* (⟶ Sect. 5.5).

5.3.4 Implementation of Finite Elements

A discretized version of the weak problem is obtained by replacing the space $\mathcal{K}$ by a finite-dimensional subspace $\widehat{\mathcal{K}}$, which is spanned by a finite number of basis functions. That is, we search for a $\widehat{y} \in \widehat{\mathcal{K}}$ such that

$$I(\widehat{y}; \widehat{v}) \geq 0 \quad \text{for all } \widehat{v} \in \widehat{\mathcal{K}} \,,$$

where $I(y;v)$ is defined in (5.24). This sets the arena for finite element methods.

[2] For the Black–Scholes $y(x,\tau)$ or $V(S,t)$ the weaker $y \in \mathcal{C}^{2,1}$ suffices. Recall that the American option is widely $\mathcal{C}^2$-smooth, except across the early-exercise curve.

As a first step to approximately solve the minimum problem, assume as in Sect. 5.3.1 separation approximations for $\widehat{y}$ and $\widehat{v}$ in the similar forms

$$\begin{aligned} \widehat{y} &= \sum_i w_i(\tau)\varphi_i(x)\,, \\ \widehat{v} &= \sum_i v_i(\tau)\varphi_i(x)\,. \end{aligned} \tag{5.25}$$

Summation is over a finite number of terms, which represents $\widehat{y},\ \widehat{v} \in \widehat{\mathcal{K}}$. The reduced smoothness of these expressions match the requirements of $\mathcal{K}$ from (5.23); time dependence is incorporated in the coefficient functions w_i and v_i. Since the basis functions φ_i represent the x_i-grid, we again perform a semidiscretization. Plugging the ansatz (5.25) into $I(\widehat{y};\widehat{v})$ from (5.24) gives

$$\begin{aligned} &\int \left\{ \left(\sum_i \frac{\mathrm{d}w_i}{\mathrm{d}\tau}\varphi_i \right) \left(\sum_j (v_j - w_j)\varphi_j \right) + \right. \\ &\qquad \left. \left(\sum_i w_i \varphi_i' \right) \left(\sum_j (v_j - w_j)\varphi_j' \right) \right\} \mathrm{d}x \\ &= \sum_i \sum_j \frac{\mathrm{d}w_i}{\mathrm{d}\tau}(v_j - w_j) \int \varphi_i \varphi_j \,\mathrm{d}x + \sum_i \sum_j w_i (v_j - w_j) \int \varphi_i' \varphi_j' \,\mathrm{d}x \geq 0\,. \end{aligned}$$

Translated into vector notation for the coefficient functions $w_i(\tau)$, $v_i(\tau)$, this is equivalent to

$$\left(\frac{\mathrm{d}w}{\mathrm{d}\tau} \right)^{\mathrm{tr}} B(v - w) + w^{\mathrm{tr}} A(v - w) \geq 0$$

or[3]

$$(v - w)^{\mathrm{tr}} \left(B \frac{\mathrm{d}w}{\mathrm{d}\tau} + Aw \right) \geq 0\,.$$

This is the (semi-)discretized weak version of $I(\widehat{y};\widehat{v}) \geq 0$. The matrices A and B are defined via the assembling described above; for equidistant steps the special versions in (5.13), (5.14) arise.

As a second step, the time τ is discretized as well. To this end let us define the vectors

$$w^{(v)} := w(\tau_v), \quad v^{(v)} := v(\tau_v)\,.$$

[3]Notation: Now v is the vector of the coefficient functions.

Upon substituting, and θ-averaging the Aw term as in Sect. 4.6.1, we arrive at the inequalities

$$\left(v^{(\nu+1)} - w^{(\nu+1)}\right)^{tr} \left(B\frac{1}{\Delta\tau}(w^{(\nu+1)} - w^{(\nu)}) + \theta A w^{(\nu+1)} + (1-\theta)Aw^{(\nu)} \right) \geq 0 \tag{5.26}$$

for all ν. For $\theta = 1/2$ this is a Crank–Nicolson-type method. Rearranging (5.26) leads to

$$\left(v^{(\nu+1)} - w^{(\nu+1)}\right)^{tr} \left((B + \Delta\tau\,\theta A)\, w^{(\nu+1)} + (\Delta\tau(1-\theta)A - B)\, w^{(\nu)} \right) \geq 0\,.$$

With the abbreviations

$$\begin{aligned} r &:= (B - \Delta\tau(1-\theta)A)\, w^{(\nu)}\,, \\ C &:= B + \Delta\tau\,\theta A\,, \end{aligned} \tag{5.27}$$

the inequality can be rewritten as

$$\left(v^{(\nu+1)} - w^{(\nu+1)}\right)^{tr} \left(Cw^{(\nu+1)} - r \right) \geq 0\,. \tag{5.28}$$

This is the fully discretized version of $I(\widehat{y}; v) \geq 0$.

5.3.4.1 Side Conditions

To match the requirements of $\mathcal{K}$, the inequalities $\widehat{y} \geq g$ and $\widehat{v} \geq g$ must hold. $\widehat{y}(x,\tau) \geq g(x,\tau)$ amounts to

$$\sum w_i(\tau)\varphi_i(x) \geq g(x,\tau)\,.$$

For hat functions φ_i (with $\varphi_i(x_i) = 1$ and $\varphi_i(x_j) = 0$ for $j \neq i$) and $x = x_j$ this implies $w_j(\tau) \geq g(x_j,\tau)$. With $\tau = \tau_\nu$ we have

$$w^{(\nu)} \geq g^{(\nu)}; \quad \text{analogously } v^{(\nu)} \geq g^{(\nu)}\,.$$

For each time level ν we must find a solution that satisfies both the inequality (5.26)–(5.28) and the side condition

$$w^{(\nu+1)} \geq g^{(\nu+1)} \text{ for all } v^{(\nu+1)} \geq g^{(\nu+1)}\,.$$

In summary, the algorithm is

Algorithm 5.4 (Finite Elements for American Standard Options)

Choose θ $(\theta = 1/2)$. *Calculate* $w^{(0)}$, *and* C *from (5.27).*

For $\nu = 1, \ldots, \nu_{\max}$:

Calculate $r = (B - \Delta\tau(1-\theta)A)w^{(\nu-1)}$ *and* $g = g^{(\nu)}$.

Construct a w *such that for all* $v \geq g$

$$(v-w)^{tr}(Cw - r) \geq 0, \quad w \geq g.$$

Set $w^{(\nu)} := w$.

This algorithm generates a discretized solution of the weak Problem 5.3: The vectors w define $\widehat{y} \in \widehat{\mathcal{K}}$ via (5.25); $\widehat{v}$ is not needed explicitly. Let us emphasize again the main step (FE), which is the kernel of this algorithm and the main labor: Construct w such that

$$\begin{aligned} &\textbf{(FE)} \quad \text{for all } v \geq g \\ &\qquad (v-w)^{tr}(Cw - r) \geq 0\,, \quad w \geq g\,. \end{aligned} \tag{5.29}$$

This task (FE) can be reformulated into a task we already solved in Sect. 4.6. To this end recall the finite-difference equation (4.44), replacing A by C, and b by r. There the following holds for w:

$$\begin{aligned} &\textbf{(FD)} \quad Cw - r \geq 0\,, \quad w \geq g\,, \\ &\qquad (Cw - r)^{tr}(w - g) = 0\,. \end{aligned} \tag{5.30}$$

Theorem 5.5 (Equivalence) *The solution of the problem (FE) is equivalent to the solution of problem (FD).*

Proof

a) (FD) $\Longrightarrow$ (FE):
Let w solve (FD), so $w \geq g$, and

$$(v-w)^{tr}(Cw - r) = (v-g)^{tr}\underbrace{(Cw - r)}_{\geq 0} - \underbrace{(w-g)^{tr}(Cw-r)}_{=0}$$

hence $(v-w)^{tr}(Cw - r) \geq 0$ for all $v \geq g$.

b) (FE) $\Longrightarrow$ (FD):
Let w solve (FE), so $w \geq g$, and

$$v^{tr}(Cw - r) \geq w^{tr}(Cw - r) \quad \text{for all } v \geq g\,.$$

Suppose the kth component of $Cw - r$ is negative, and make v_k arbitrarily large. Then the left-hand side becomes arbitrarily small, which is a contradiction. So $Cw - r \geq 0$. Now

$$w \geq g \implies (w-g)^{tr}(Cw-r) \geq 0\,.$$

Set in (FE) $v = g$, then $(w-g)^{tr}(Cw-r) \leq 0$. Therefore $(w-g)^{tr}(Cw-r) = 0$.

5.3.4.2 Implementation

As a consequence of this equivalence, the solution of the finite-element problem (FE) can be calculated with the methods we applied to solve problem (FD) in Sect. 4.6. Following the exposition in Sect. 4.6.2, the kernel of the finite-element Algorithm 5.4 can be written as follows

(FE′) Solve $Cw = r$ componentwise such that

the side condition $w \geq g$ is obeyed.

The vector v is not calculated. Boundary conditions on w are set up in the same way as discussed in Sect. 4.4 and summarized in Algorithm 4.14. Consequently, the finite-element algorithm parallels Algorithm 4.14 closely in the special case of an equidistant x-grid; there is no need to repeat this algorithm ($\longrightarrow$ Exercise 5.4). In the general nonequidistant case, the off-diagonal and the diagonal elements of the tridiagonal matrix C vary with i. Then the formulation of the SOR-loop gets more involved. The details of the implementation are technical and omitted. The Algorithm 4.15 is the same in the finite-element case.

The computational results match those of Chap. 4 and are not repeated. The costs of the presented simple version of a finite-element approach are slightly lower than that of the finite-difference approach, because we can take advantage of an optimal spacing of the mesh points x_i. For arguments discussing the closeness of $\widehat{y}$ to y, we refer to Sect. 5.5.

5.4 Two-Asset Options

In Sect. 3.5.5 we discussed an option based on two assets with prices S_1, S_2. There we applied Monte Carlo to simulate the GBM model, see Example 3.9. For the mathematical model we have chosen the Black–Scholes market. The corresponding PDE for the value function $V(S_1, S_2, t)$ is

$$\begin{aligned}\frac{\partial V}{\partial t} &+ \tfrac{1}{2}\sigma_1^2 S_1^2 \frac{\partial^2 V}{\partial S_1^2} + (r-\delta_1)S_1 \frac{\partial V}{\partial S_1} - rV \\ &+ \tfrac{1}{2}\sigma_2^2 S_2^2 \frac{\partial^2 V}{\partial S_2^2} + (r-\delta_2)S_2 \frac{\partial V}{\partial S_2} + \rho\sigma_1\sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2} = 0\,,\end{aligned} \tag{5.31}$$

with dividend rates δ_1, δ_2. (For the general case see Sect. 6.2.) Notice that for $S_2 = 0$ the familiar one-dimensional Black–Scholes equation results. The model is completed by a payoff function $\Psi(S_1, S_2)$ and the terminal condition $V(S_1, S_2, T) = \Psi(S_1, S_2)$. The computational domain $\mathcal{D}$ is two-dimensional, $\mathcal{D} \subset \mathbb{R}^2$ (disregarding time t).

Example 5.6 (European Call on a Basket with Double Barrier) We consider a call on a two-asset basket with two knock-out barriers. The payoff of this exotic European-style option is

$$\Psi(S_1, S_2) = (S_1 + S_2 - K)^+ ,$$

up to the barriers (see Fig. 5.1). In the underlying basket the two assets are of equal weight. The two knock-out barriers are given by B_1 and B_2, down-and-out at B_1, and up-and-out at B_2. That is, the option ceases to exist when $S_1 + S_2 \le B_1$, or when $S_1 + S_2 \ge B_2$; in both cases $V = 0$. In this example, the computational domain $\mathcal{D}$ is easy to define: The value function is zero outside the barriers. Hence the domain is bounded by the two lines $S_1 + S_2 = B_1$ and $S_1 + S_2 = B_2$. This shape of $\mathcal{D}$ naturally suggests to tile the domain into a grid of triangular elements $\mathcal{D}_k$. One possible triangulation is shown in Fig. 5.5, where a structured regular subdivision is applied. For this example we choose the parameters

$$\begin{aligned} &K = 1\,,\ T = 1\,,\ \sigma_1 = \sigma_2 = 0.25\,,\ \rho = 0.7\,,\ r = 0.05\,, \\ &\delta_1 = \delta_2 = 0\,,\ \ B_1 = 1\,,\ B_2 = 2\,. \end{aligned}$$

The values V for $S_1 \to 0$ and $S_2 \to 0$ are known by the one-dimensional Black–Scholes equation; just set either $S_1 = 0$ or $S_2 = 0$ in (5.31). These values of single-asset double-barrier options for $B_1 \le S \le B_2$ can be evaluated by a closed-form formula, see [172]. We shall come back to this example below.

5.4.1 Analytical Preparations

It is convenient to solve the Black–Scholes equation in divergence form. To this end, use standard PDE variables $x := S_1$, $y := S_2$ for the independent variables, and $u(x, y, t)$ for the dependent variable, and derive the vector PDE for u

$$-\nabla \cdot (D(x, y)\nabla u) + b(x, y)^{tr} \nabla u + ru = u_t\,. \tag{5.32}$$

This makes use of the formal "nabla" vector $\nabla := (\frac{\partial}{\partial x}, \frac{\partial}{\partial y})^{tr}$, and

$$\begin{aligned} D(x, y) &:= \tfrac{1}{2} \begin{pmatrix} \sigma_1^2 x^2 & \rho\sigma_1\sigma_2 xy \\ \rho\sigma_1\sigma_2 xy & \sigma_2^2 y^2 \end{pmatrix}, \\ b(x, y) &:= -\begin{pmatrix} (r - \delta_1 - \sigma_1^2 - \rho\sigma_1\sigma_2/2)\, x \\ (r - \delta_2 - \sigma_2^2 - \rho\sigma_1\sigma_2/2)\, y \end{pmatrix}. \end{aligned} \tag{5.33}$$

∇u is the gradient of u, and the dot-product notation

$$\nabla \cdot U = \frac{\partial U_1}{\partial x} + \frac{\partial U_2}{\partial y}$$

for a vector function U denotes the divergence; the $\cdot$ corresponds to the scalar product, similar as ^{tr} for vectors. The reader is invited to check the equivalence with (5.31) ($\longrightarrow$ Exercise 5.5). The advantage of version (5.32) over (5.31) lies in a simple treatment of the second-order derivatives; they can be removed, and a weak version can be derived. This will become apparent below.

5.4.2 Weighted Residuals

The partial differential equation (5.32) can be represented by $R(u, x, y, t) = 0$, where

$$\begin{aligned} R(u, x, y, t) := & - \nabla \cdot (D(x, y) \nabla u(x, y, t)) + b(x, y)^{tr} \nabla u(x, y, t) \\ & + r u(x, y, t) - \frac{\partial u(x, y, t)}{\partial t} \end{aligned}$$

denotes the residual. As in Sect. 5.1, the residual is used to set up an integral equation. To this end, introduce weighting functions v, multiply the residual of the PDE with $v(x, y, t)$ and request

$$\int_{\mathcal{D}} R(u, x, y, t)\, v \, \mathrm{d}x \, \mathrm{d}y = 0 \,. \tag{5.34}$$

This integral over the computational domain $\mathcal{D} \subset \mathbb{R}^2$ is a double integral. It depends on t, and should vanish for all $0 \leq t \leq T$ and arbitrary v. We consider u to be a solution in case (5.34) holds for "all" v. This is a weak version of the PDE and requires less regularity of its "weak" solutions u. Aspects of accuracy are postponed to Sect. 5.5.

To exploit the potential of the integral version (5.34), we transform the second-order derivatives to first order, comparable to integration by parts. The leading integral over the second-order term is

$$\int_{\mathcal{D}} -\nabla \cdot (D \nabla u)\, v \, \mathrm{d}x \, \mathrm{d}y \,.$$

The reader may check for the vector $U := v D \nabla u$ the formula for the divergence $\nabla \cdot U$, namely,

$$\nabla \cdot (v D \nabla u) = (\nabla v)^{tr} D \nabla u + v \nabla \cdot D \nabla u \,,$$

and hence

$$-\int_{\mathcal{D}} v\,\nabla\cdot(D\nabla u)\,\mathrm{d}x\,\mathrm{d}y = \int_{\mathcal{D}} (\nabla v)^{tr} D\nabla u\,\mathrm{d}x\,\mathrm{d}y - \int_{\mathcal{D}} \nabla\cdot(vD\nabla u)\,\mathrm{d}x\,\mathrm{d}y\,.$$

Next we quote the divergence theorem, here for the two-dimensional situation:

$$\int_{\mathcal{D}} \nabla\cdot U\,\mathrm{d}x\,\mathrm{d}y = \int_{\partial\mathcal{D}} U^{tr} n\,\mathrm{d}s\,, \tag{5.35}$$

where $\partial\mathcal{D}$ denotes the boundary of $\mathcal{D}$, and n is the outward unit normal vector on $\partial\mathcal{D}$. (n is perpendicular to the curve $\partial\mathcal{D}$ and points away from $\mathcal{D}$.) The parameter s measures the arclength along the boundary $\partial\mathcal{D}$.[4] We apply the divergence theorem to the specific vector $U := vD\nabla u$, and arrive at the result for the second-order term

$$-\int_{\mathcal{D}} v\,\nabla\cdot(D\nabla u)\,\mathrm{d}x\,\mathrm{d}y = \int_{\mathcal{D}} (\nabla v)^{tr} D\nabla u\,\mathrm{d}x\,\mathrm{d}y - \int_{\partial\mathcal{D}} (vD\nabla u)^{tr} n\,\mathrm{d}s\,.$$

In (5.32)/(5.33) the matrix D is symmetric, $D = D^{tr}$. For symmetric D the integrand in the boundary integral is $v(\nabla u)^{tr} Dn$. After the above transformations of the leading integral, we rewrite (5.34) into

$$\int_{\mathcal{D}} \left[(\nabla v)^{tr} D\nabla u + vb^{tr}\nabla u + ruv - \frac{\partial u}{\partial t} v \right] \mathrm{d}x\,\mathrm{d}y - \int_{\partial\mathcal{D}} v(\nabla u)^{tr} Dn\,\mathrm{d}s = 0\,. \tag{5.36}$$

Recall that both u and v as well as ∇u and ∇v depend on x, y, t, and the integrals on t. This is the weak version of the PDE (5.32).

Next discretize the time $0 \le t \le T$ as in Chap. 4, say, with equidistant steps Δt. For the simplest implicit approach, the derivative with respect to time t is resolved by the first-order difference quotient,

$$\frac{\partial u(x,y,t)}{\partial t} \approx \frac{u(x,y,t+\Delta t) - u(x,y,t)}{\Delta t}\,.$$

For backward running time t,

$$u_{\mathrm{pre}} := u(x,y,t+\Delta t)$$

is known at time t from the calculation of the previous time level. The analogue of the fully implicit time-stepping method is then to solve (5.36) at time level t for $\frac{\partial u}{\partial t}$

[4] Recall from calculus the definition $\int_C f(x,y)\mathrm{d}s = \int_a^b f(g(\xi),h(\xi))\frac{\mathrm{d}s}{\mathrm{d}\xi}\mathrm{d}\xi$ where $(g(\xi),h(\xi))$ for $a \le \xi \le b$ is a parameterization of a planar curve C; ξ is the curve parameter. The value of this *line integral* is independent of the orientation of the curve C and independent of the particular parameterization.

replaced by

$$\frac{1}{\Delta t}(u_{\text{pre}} - u)\,,$$

starting at $t = T - \Delta t$ with the payoff, $u_{\text{pre}} = \Psi$. With this approximation, the function u in (5.36) approximates the value function V at time level t. Alternatively, a second-order time-discretization can be applied, similar as in Sect. 4.3. For the required regularity of the functions u and v, consult Sect. 5.5.

5.4.3 Boundary

Boundary conditions enter via the boundary integral around the boundary $\partial\mathcal{D}$. In practice, the computational domain $\mathcal{D}$ is defined by specifying $\partial\mathcal{D}$. To this end, express the curve $\partial\mathcal{D}$ as the union of a finite number of non-overlapping piecewise smooth boundary curves $\partial\mathcal{D}_1, \partial\mathcal{D}_2, \ldots$. Each of these curves must be parameterized as in

$$\partial\mathcal{D}_1 := \{\,(g_1(\xi), h_1(\xi)) \mid a_1 \le \xi \le b_1\,\}\,.$$

In this way, an orientation is given by starting the curve at the parameter value $\xi = a_1$ and ending at $\xi = b_1$. By specifying parameter intervals as $a_1 \le \xi \le b_1$ and parametric functions as g_1, h_1, the entire boundary is defined. The convention is that the orientation is done such that the domain $\mathcal{D}$ is *on the left-hand side*, as we run through the parameterizations for increasing parameter values ξ.

Now the curve $\partial\mathcal{D}$ is defined and we address the boundary integral along that curve. It is split into a sum of integrals according to the piecewise smooth curves $\partial\mathcal{D}_1, \partial\mathcal{D}_2, \ldots$. For example, the boundary of the domain in Fig. 5.5 consists of four such parts (⟶ Exercise 5.6).

The product-type integrand $f(x, y) := v(\nabla u)^{tr} Dn$ suggests to place emphasis on two specific kinds of boundary condition, namely,

- v is prescribed (Dirichlet boundary conditions),
- $(\nabla u)^{tr} Dn$ is prescribed (Neumann boundary conditions).

The boundary differential operator $(\nabla u)^{tr} Dn = n^{tr} D \nabla u$ can be considered as a generalized directional derivative since $\frac{\partial u}{\partial n} = n^{tr} \nabla u$. Mixed boundary conditions are possible as well. If we cast the components of the vector $n^{tr} D$ into a vector (α_1, α_2), then all type of boundary conditions can be written in the form

$$\alpha_1(x, y)\frac{\partial u}{\partial x} + \alpha_2(x, y)\frac{\partial u}{\partial y} = \alpha_0(x, y)\, u + \beta(x, y)$$

with proper functions α_0 and β. Then

$$v\,(\alpha_0(x,y)\,u+\beta(x,y))$$

is substituted into the boundary integral, which is approximated numerically using the edges of the triangulation of $\mathcal{D}$.

Fortunately, boundary conditions are frequently of simple form. In particular one encounters the two types

- $u=0$ (or $v=0$), which is of Dirichlet type with $\alpha_1=\alpha_2=\beta=0$ and $\alpha_0\neq 0$.
- $(\nabla u)^{tr}Dn=0$, which is of Neumann type with $\alpha_0=\beta=0$ and nonzero vector (α_1,α_2).

The boundary $\partial\mathcal{D}$ may consist, for example, of two parts $\partial\mathcal{D}_{\mathrm{D}}$ and $\partial\mathcal{D}_{\mathrm{N}}$ with $\partial\mathcal{D}=\partial\mathcal{D}_{\mathrm{D}}\cup\partial\mathcal{D}_{\mathrm{N}}$, $\partial\mathcal{D}_{\mathrm{D}}\cap\partial\mathcal{D}_{\mathrm{N}}=\emptyset$, and Dirichlet conditions on $\partial\mathcal{D}_{\mathrm{D}}$ and Neumann conditions on $\partial\mathcal{D}_{\mathrm{N}}$. Clearly, boundary integrals vanish for the special cases $v=0$ or $(\nabla u)^{tr}Dn=0$. Neumann conditions are advantageous in that they need not be specified for weak formulations. This entails an advantage of FEM over discretizing the PDEs by finite differences. In the latter case, *all* boundary conditions must be implemented. For FEM it suffices to implement Dirichlet conditions. Defining the right boundary conditions can be demanding. Aside to be financially meaningful, another aim is the problem to be well-posed—that is, it defines a unique solution. To some extent, defining proper boundary conditions is an art.

Example 5.7 (European Binary Put as in Example 3.9) In Chap. 3 the Example 3.9 of a binary put was simulated with Monte Carlo, and no boundary or boundary conditions were needed. Here we prepare the example to be solved by FEM. Again, $x:=S_1$, $y:=S_2$. As in Chap. 4, the domain $0<x<\infty$, $0<y<\infty$ must be truncated to finite size. A simple choice of a computational domain is a rectangle

$$\mathcal{D}=\{\,(x,y)\mid 0\le x\le x_{\max},\ 0\le y\le y_{\max}\,\}$$

with $x_{\max},y_{\max}$ large enough such that zero boundary conditions $u=0$ can be chosen as approximation for $x=x_{\max}$ or $y=y_{\max}$. The rectangle is bounded by four straight lines, which can be parameterized, for example, by

$$\begin{aligned}
\partial\mathcal{D}_1&:=\{\,x=\xi,\ y=0\ \mid\ \ 0\le\xi\le x_{\max}\,\},\\
\partial\mathcal{D}_2&:=\{\,x=x_{\max},\ y=\xi\ \mid\ \ 0\le\xi\le y_{\max}\,\},\\
\partial\mathcal{D}_3&:=\{\,x=x_{\max}-\xi,\ y=y_{\max}\ \mid\ \ 0\le\xi\le x_{\max}\,\},\\
\partial\mathcal{D}_4&:=\{\,x=0,\ y=y_{\max}-\xi\ \mid\ \ 0\le\xi\le y_{\max}\,\}.
\end{aligned}$$

Now $\partial\mathcal{D}=\partial\mathcal{D}_1\cup\partial\mathcal{D}_2\cup\partial\mathcal{D}_3\cup\partial\mathcal{D}_4$, and the parameterized curve has the domain on the left.

Dirichlet conditions are imposed for $\partial\mathcal{D}_2$ and $\partial\mathcal{D}_3$, where we have chosen to approximate boundary values by requesting $u = 0$. For $y = 0$ the boundary conditions can be chosen as the values of the one-dimensional European binary put. An analytic formula for the one-dimensional case of a European binary put is

$$V_{\text{binP}}^{\text{Eur}}(S,t) := c\,\mathrm{e}^{-r(T-t)}\,F\left(-\frac{\log(S/K)+(r-\sigma^2/2)(T-t)}{\sigma\sqrt{T-t}}\right),$$

for a face value c, with standard normal distribution F [172]. For $y = 0$ we set $S = x$. The same formula can be applied for the boundary with $x = 0$; then $S = y$. In this way, on $\partial\mathcal{D}_1$ and $\partial\mathcal{D}_4$ the boundary conditions are of Dirichlet type with $u = V_{\text{binP}}^{\text{Eur}}$. With this choice of boundary conditions, $\partial\mathcal{D}_{\text{D}} = \partial\mathcal{D}$ and $\partial\mathcal{D}_{\text{N}} = \emptyset$. But there is a simpler choice: As [300] points out, this Dirichlet condition is implicitly defined by the PDE, because the one-dimensional PDE is embedded in (5.31) for $S_1 = 0$ or $S_2 = 0$. So no boundary condition needs to be specified along $\partial\mathcal{D}_1$ and $\partial\mathcal{D}_4$. This amounts to zero Neumann conditions. Both the Dirichlet version and the Neumann version work. The latter has the advantage of avoiding the effort of evaluating $V_{\text{binP}}^{\text{Eur}}$.

The implementation of the weak form in (5.36) is straightforward when, for example, the package `FreeFem++` is applied. Thereby a figure similar as Fig. 3.7 is produced easily.

5.4.4 Involved Matrices

The accuracy of FEM depends on how the grid is chosen. Algorithms for mesh generation and mesh adaption are needed, but these are demanding topics. It is cumbersome to implement a two-dimensional FEM yourself. For first results, one may work with a fixed structured grid. But in general it is advisable and comfortable to apply a FEM package to solve (5.36). Here we merely focus on how the two-dimensional analogue of the hat functions enters.

For the Ritz–Galerkin approach we apply the basis representation

$$w(x,y,t) = \sum_i w_i(t)\,\varphi_i(x,y) \tag{5.37}$$

as approximation for u, and set $v = \varphi_j$. This ansatz separates time τ and "space" (x,y). The functions φ_i are defined on $\mathcal{D}$.

For basis functions, we choose the two-dimensional hat functions, which perfectly match triangular elements. The situation is shown schematically in Fig. 5.9. There the central node l is node of several adjacent triangles, which constitute the support (shaded) on which φ_l is built by planar pieces. This approach defines a tent-like hat function φ_l, which is zero "outside." By linear combination of such basis functions, piecewise planar surfaces above the computational domain are constructed. Locally, for one triangle, this may look like the element in Fig. 5.4.

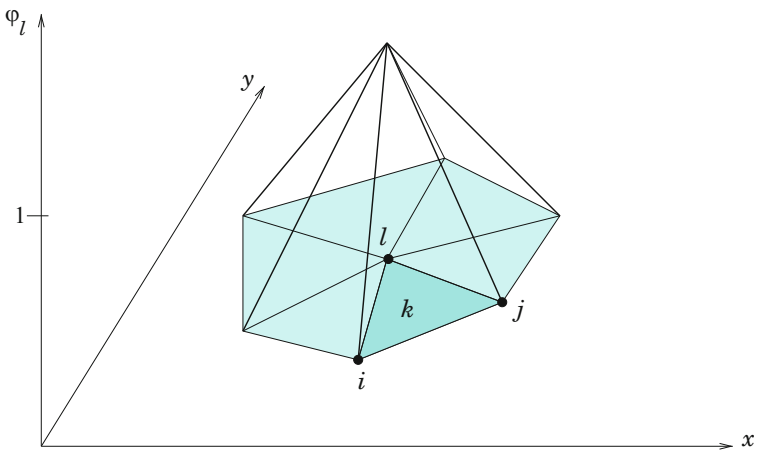

Fig. 5.9 Two-dimensional hat function $\varphi_l(x, y)$ (zero outside the *shaded area*)

Notice that $\nabla w = \sum w_i \nabla \varphi_i$. The weak form of (5.36) leads to

$$\int_{\mathcal{D}} (\nabla \varphi_j)^{tr} D \sum w_i \nabla \varphi_i + \varphi_j \left[b^{tr} (\sum w_i \nabla \varphi_i) + r \sum w_i \varphi_i - \sum \frac{\partial w_i}{\partial t} \varphi_i \right] \mathrm{d}x\,\mathrm{d}y - \int_{\partial \mathcal{D}} \varphi_j (\sum w_i \nabla \varphi_i)^{tr} D n \,\mathrm{d}s = 0\,,$$

for all j. This is a system of ODEs

$$\sum_i w_i \int_{\mathcal{D}} \left[(\nabla \varphi_j)^{tr} D \nabla \varphi_i + \varphi_j b^{tr} \nabla \varphi_i + \varphi_j r \varphi_i \right] \mathrm{d}x\,\mathrm{d}y - \sum_i \frac{\partial w_i}{\partial t} \int_{\mathcal{D}} \varphi_i \varphi_j \,\mathrm{d}x\,\mathrm{d}y - \sum_i w_i \int_{\partial \mathcal{D}} \varphi_j (\nabla \varphi_i)^{tr} D n \,\mathrm{d}s = 0\,. \tag{5.38}$$

As an exercise, the reader should rewrite this ODE system in matrix-vector notation. In summary, FEM needs the integrals over the domain $\mathcal{D}$

$$\int (\nabla \varphi_j)^{tr} D \nabla \varphi_i \quad (\text{“diffusion terms”})\,,$$

$$\int \varphi_j b^{tr} \nabla \varphi_i \quad (\text{“convection terms”})\,,$$

$$\int \gamma \varphi_j \varphi_i \quad (\text{“reaction terms”})\,,$$

where γ is chosen appropriately, and in addition boundary integrals along $\partial \mathcal{D}$.

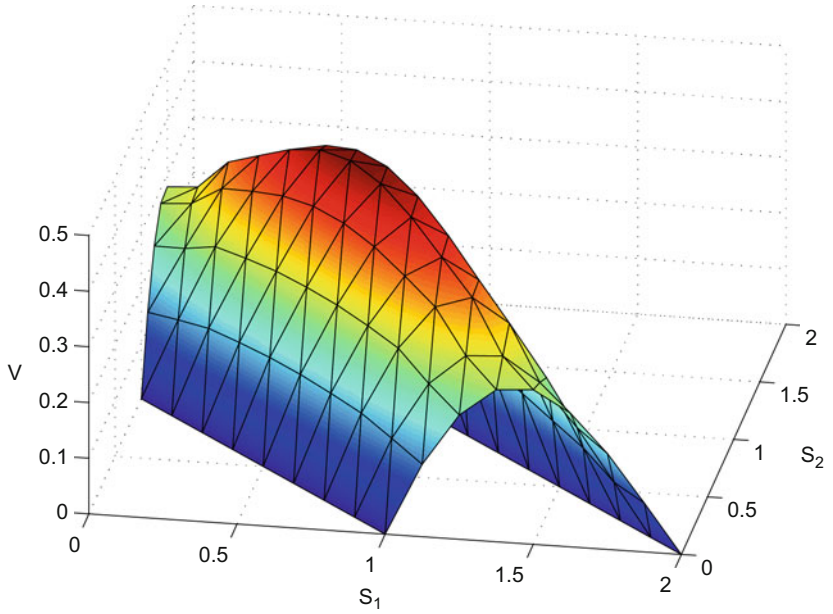

Fig. 5.10 Rough approximation of the value function $V(S_1, S_2, 0)$ of a basket double-barrier call option, Example 5.6. With kind permission of Anna Kvetnaia

For each number k of a triangle, there are three vertices of the triangle, with node numbers i, j, l in Fig. 5.9. Hence the table $\mathcal{I}$ of index sets that assigns nodes to triangles includes the entry

$$\mathcal{I}_k := \{i, j, l\}\,.$$

Only for the three node numbers $i, j, l \in \mathcal{I}_k$ the local integrals on $\mathcal{D}_k$ are nonzero. They can be arranged into 3×3 element matrices. For the derivation of the integrals, it makes sense to use a local numbering $1_k, 2_k, 3_k$ for the nodes of $\mathcal{D}_k$. For each global matrix, the assembling loop over k distributes up to 27 local integrals calculated on $\mathcal{D}_k$, nine integrals of each of the above three types.[5]

Back to Example 5.6, we solve (5.36) with FEM. Figure 5.10 shows a FEM solution with 192 triangles. Figure 5.11 illustrates a mesh structure for higher resolution obtained with `FreeFem++`. In the two-dimensional case, because of higher costs, we typically confine ourselves to an accuracy lower than in the one-dimensional situation. Based on our results we state

$$V(1.25,\, 0.25,\, 0) \approx 0.2949\,.$$

[5]Basic ingredients for the calculation of the local integrals on an arbitrary triangle $\mathcal{D}_k$ are the relations in Exercise 5.7. See also Exercises 5.8 and 5.9.

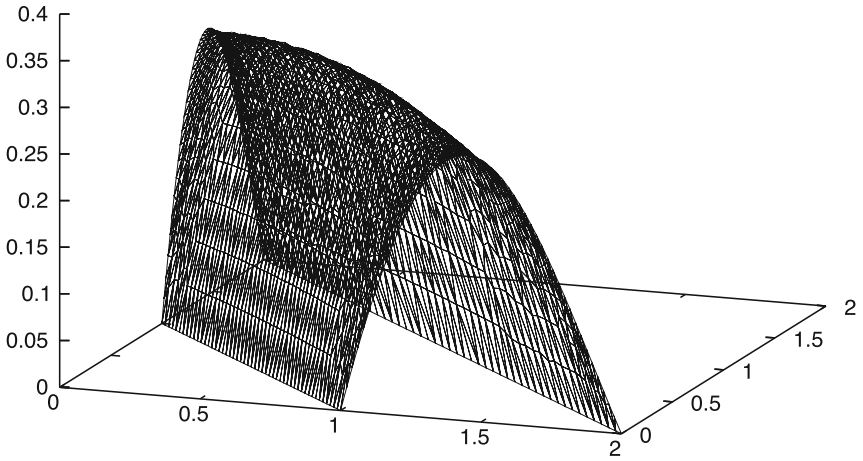

Fig. 5.11 Finer approximation of the value function $V(S_1, S_2, 0)$ of a basket double-barrier call option, Example 5.6

Example 5.8 (Heston's PDE) In Example 1.16 Heston's model was introduced, where v denotes a stochastic volatility. The corresponding PDE from [178] is

$$\begin{aligned} &\frac{\partial V}{\partial t} + \frac{1}{2} v S^2 \frac{\partial^2 V}{\partial S^2} + \frac{1}{2} \sigma_v^2 v \frac{\partial^2 V}{\partial v^2} + \rho \sigma_v v S \frac{\partial^2 V}{\partial S \partial v} \\ &\quad + rS \frac{\partial V}{\partial S} + [\kappa(\theta - v) - \lambda v] \frac{\partial V}{\partial v} - rV = 0\,, \end{aligned} \tag{5.39}$$

with parameters as in (1.59), and λ standing for the market price of volatility risk. Here we are interested in solutions $V(S, v, t)$ on part of a two-dimensional (S, v)-plane. The PDE (5.39) can be cast into version (5.32). As an exercise, the reader is encouraged to derive D and b, and with the payoff of a call and an own choice of parameters, to think about suitable boundary conditions, and to do experiments with (5.39). Note that for a call a reasonable requirement for maximum values of the volatility v is $V = S$. When in addition the interest rate r is replaced by a stochastic variable, the PDE is based on a three-dimensional domain [163].

5.5 Error Estimates

The similarity of the finite-element equation (5.21) with the finite-difference equation (4.24) suggests that the errors may be of the same order. In fact, numerical experiments confirm that the finite-element approach with the linear basis functions from Definition 5.1 produces errors decaying quadratically with the mesh size. Applying the finite-element Algorithm 5.4 and entering the calculated data into a

diagram as Fig. 4.14, confirms the quadratic order experimentally. The proof of this order of the error is more difficult for finite-element methods because weak solutions assume less smoothness. For standard options, the separation of variables in (5.19) also separates the discussion of the order, and an analysis of the one-dimensional situation suffices. This section explains some basic ideas of how to derive error estimates. We begin with reconsidering some of the related topics that have been introduced in previous sections.

5.5.1 Strong and Weak Solutions

Our exposition will be based on the model problem (5.15). That is, the simple second-order differential equation

$$-u'' = f(x) \quad \text{for } \alpha < x < \beta \tag{5.40}$$

with given f, and homogeneous Dirichlet-boundary conditions

$$u(\alpha) = u(\beta) = 0 \tag{5.41}$$

will serve as illustration. The differential equation is of the form $Lu = f$, compare (5.2). The domain $\mathcal{D} \subseteq \mathbb{R}^n$ on which functions u are defined specializes for $n = 1$ to the open and bounded interval $\mathcal{D} = \{x \in \mathbb{R}^1 \mid \alpha < x < \beta\}$. For continuous f, solutions of the differential equation (5.40) satisfy $u \in \mathcal{C}^2(\mathcal{D})$. In order to have operative boundary conditions, solutions u must be continuous on $\mathcal{D}$ including its boundary, which is denoted $\partial\mathcal{D}$. Therefore we require $u \in \mathcal{C}^0(\overline{\mathcal{D}})$ where $\overline{\mathcal{D}} := \mathcal{D} \cup \partial\mathcal{D}$. In summary, classical solutions of second-order differential equations require

$$u \in \mathcal{C}^2(\mathcal{D}) \cap \mathcal{C}^0(\overline{\mathcal{D}})\,. \tag{5.42}$$

The function space $\mathcal{C}^2(\mathcal{D}) \cap \mathcal{C}^0(\overline{\mathcal{D}})$ must be reduced further to comply with the boundary conditions.

For weak solutions the function space is larger ($\longrightarrow$ Appendix C.3). For functions u and v we define the inner product

$$(u, v) := \int_{\mathcal{D}} uv \,\mathrm{d}x\,. \tag{5.43}$$

Strong solutions u of $Lu = f$ satisfy also

$$(Lu, v) = (f, v) \quad \text{for all } v\,. \tag{5.44}$$

Specifically for the model problem (5.40)/(5.41) integration by parts leads to

$$(Lu, v) = -\int_\alpha^\beta u'' v \, \mathrm{d}x = -u'v\Big|_\alpha^\beta + \int_\alpha^\beta u'v' \, \mathrm{d}x \,.$$

The nonintegral term on the right-hand side of the equation vanishes in case also v satisfies the homogeneous boundary conditions (5.41). The remaining integral is a **bilinear form**, which we abbreviate

$$b(u, v) := \int_\alpha^\beta u'v' \, \mathrm{d}x \,. \tag{5.45}$$

Bilinear forms as $b(u, v)$ from (5.45) are linear in each of the two arguments u and v. For example, $b(u_1 + u_2, v) = b(u_1, v) + b(u_2, v)$ holds. The bilinear form (5.45) is symmetric, $b(u, v) = b(v, u)$. For several classes of more general differential equations analogous bilinear forms are obtained. Formally, (5.44) can be rewritten as

$$b(u, v) = (f, v) \,, \tag{5.46}$$

where we assume that v satisfies the homogeneous boundary conditions (5.41).

The Eq. (5.46) has been derived out of the differential equation, for the solutions of which we have assumed smoothness in the sense of (5.42). Many "solutions" of practical importance do not satisfy (5.42) and, accordingly, are not smooth. In several applications, u or derivatives of u have discontinuities. For instance consider the obstacle problem of Sect. 4.5.5: The second derivative u'' of the solution fails to be continuous at α and β. Therefore $u \notin \mathcal{C}^2(-1, 1)$ no matter how smooth the data function is, compare Fig. 4.10. As mentioned earlier, integral relations require less smoothness.

In the derivation of (5.46) the integral version has resulted as a consequence of the primary differential equation. This is contrary to wide areas of applied mathematics, where an integral relation is based on first principles, and the differential equation is derived in a second step. For example, in the calculus of variations a minimization problem may be described by an integral performance measure, and the differential equation is a necessary criterion [350]. This situation suggests considering the integral relation as an equation of its own right rather than as offspring of a differential equation. This leads to the question, *what is the maximal function space* such that (5.46) with (5.43), (5.45) is meaningful? That means to ask, for which functions u and v do the integrals exist? For a more detailed background we refer to Appendix C.3. For the introductory exposition of this section it may suffice to sketch the maximal function space briefly. The suitable function space is denoted $\mathcal{H}^1$, the version equipped with the boundary conditions is denoted $\mathcal{H}_0^1$. This *Sobolev space* consists of those functions that are continuous on $\mathcal{D}$ and that are *piecewise differentiable* and satisfy the boundary conditions (5.41). This function space corresponds to the class of functions $\mathcal{K}$ in (5.23). By means of the Sobolev space $\mathcal{H}_0^1$ a weak solution of $Lu = f$ is defined, where L is a second-order differential operator and b the corresponding bilinear form.

Definition 5.9 (Weak Solution) $u \in \mathcal{H}_0^1$ is called weak solution [of $Lu = f$], if $b(u, v) = (f, v)$ holds for all $v \in \mathcal{H}_0^1$.
This definition implicitly expresses the task: find a $u \in \mathcal{H}_0^1$ *such that* $b(u, v) = (f, v)$ *for all* $v \in \mathcal{H}_0^1$. This problem is called *variational problem*. The model problem (5.40)/(5.41) serves as example for $Lu = f$; the corresponding bilinear form $b(u, v)$ is defined in (5.45) and (f, v) in (5.43). For the integrals (5.43) to exist, we in addition require f to be square integrable ($f \in \mathcal{L}^2$, compare Appendix C.3). Then (f, v) exists because of the Schwarzian inequality (C.16). In a similar way, weak solutions are introduced for more general problems; the formulation of Definition 5.9 applies.

5.5.2 *Approximation on Finite-Dimensional Subspaces*

For a practical computation of a weak solution the infinite-dimensional space $\mathcal{H}_0^1$ is replaced by a finite-dimensional subspace. Such finite-dimensional subspaces are spanned by basis functions φ_i. Simple examples are the hat functions of Sect. 5.2. Reminding of the important role splines play as basis functions, the finite-dimensional subspaces are denoted $\mathcal{S}$, and are called *finite-element spaces*. As stated in Property 5.2(a), the hat functions $\varphi_0, \ldots, \varphi_m$ span the space of polygons. Recall that each such polygon v can be represented as linear combination

$$v = \sum_{i=0}^{m} c_i \varphi_i \,.$$

The coefficients c_i are uniquely determined by the values of v at the nodes, $c_i = v(x_i)$. We call hat functions "linear elements" because they consist of piecewise straight lines. Apart from linear elements, for example, also quadratic or cubic elements are used, which are piecewise polynomials of second or third degree [79, 335, 382]. The attainable accuracy is different for basis functions consisting of higher-degree polynomials.

Since by definition the functions of the Sobolev space $\mathcal{H}_0^1$ fulfill the homogeneous boundary conditions, each subspace does so as well. Again the subscript $_0$ indicates the realization of the homogeneous boundary conditions (5.41).[6] A finite-dimensional subspace of $\mathcal{H}_0^1$ is defined by

$$\mathcal{S}_0 := \left\{ v = \sum_{i=0}^{m} c_i \varphi_i \;\mid\; \varphi_i \in \mathcal{H}_0^1 \right\} \,. \tag{5.47}$$

[6] In this subsection the meaning of the index $_0$ is twofold: It is the index of the "first" hat function, and serves as symbol of the homogeneous boundary conditions (5.41).

Properties of $\mathcal{S}_0$ are determined by the basis functions φ_i. As mentioned earlier, basis functions with small supports give rise to sparse matrices. The partition (5.4) of $\mathcal{D}$ is implicitly included in the definition $\mathcal{S}_0$ because this information is contained in the definition of the φ_i. For our purposes the hat functions suffice. The larger m is, the better $\mathcal{S}_0$ approximates the space $\mathcal{H}_0^1$, since a finer discretization (smaller $\mathcal{D}_k$) allows to approximate the functions from $\mathcal{H}_0^1$ better by polygons. We denote the largest diameter of the $\mathcal{D}_k$ by h, and ask for convergence. That is, we study the behavior of the error for $h \to 0$ (basically $m \to \infty$).

In analogy to the variational problem expressed in connection with Definition 5.9, a *discrete* weak solution w is defined by replacing the space $\mathcal{H}_0^1$ by a finite-dimensional subspace $\mathcal{S}_0$:

Problem 5.10 (Discrete Weak Solution) *Find a $w \in \mathcal{S}_0$ such that $b(w, v) = (f, v)$ for all $v \in \mathcal{S}_0$.*

The quality of the approximation relies on the discretization fineness h of $\mathcal{S}_0$, which is occasionally emphasized by writing w_h.

5.5.3 *Quadratic Convergence*

Having defined a weak solution u and a discrete approximation w, we turn to the error $u - w$. To measure the distance between functions in $\mathcal{H}_0^1$ we use the norm $\| \ \|_1$ ($\longrightarrow$ Appendix C.3). That is, our first aim is to construct a bound on $\|u - w\|_1$. Let us suppose that the bilinear form is continuous and $\mathcal{H}^1$-*elliptic*:

Assumptions 5.11 (Continuous $\mathcal{H}^1$-Elliptic Bilinear Form)

(a) There is a $\gamma_1 > 0$ such that $|b(u, v)| \leq \gamma_1 \|u\|_1 \|v\|_1$ for all $u, v \in \mathcal{H}^1$.
(b) There is a $\gamma_2 > 0$ such that $b(v, v) \geq \gamma_2 \|v\|_1^2$ for all $v \in \mathcal{H}^1$.

The assumption (a) is the continuity, and the property in (b) is called $\mathcal{H}^1$-ellipticity. Under the Assumptions 5.11, the problem to find a weak solution following Definition 5.9, possesses exactly one solution $u \in \mathcal{H}_0^1$; the same holds true for Problem 5.10. This is guaranteed by the Theorem of Lax–Milgram [53, 79]. In view of $\mathcal{S}_0 \subseteq \mathcal{H}_0^1$,

$$b(u, v) = (f, v) \quad \text{for all } v \in \mathcal{S}_0 .$$

Subtracting $b(w, v) = (f, v)$ and invoking the bilinearity implies

$$b(w - u, v) = 0 \quad \text{for all } v \in \mathcal{S}_0 . \tag{5.48}$$

The property of (5.48) is called *error-projection property*. The Assumptions 5.11 and the error projection are the basic ingredients to obtain a bound on the error $\|u - w\|_1$:

Lemma 5.12 (Céa) *Suppose the Assumptions 5.11 are satisfied. Then*

$$\|u - w\|_1 \leq \frac{\gamma_1}{\gamma_2} \inf_{v \in \mathcal{S}_0} \|u - v\|_1 \,. \tag{5.49}$$

Proof $v \in \mathcal{S}_0$ implies $\tilde{v} := w - v \in \mathcal{S}_0$. Applying (5.48) for $\tilde{v}$ yields

$$b(w - u, w - v) = 0 \quad \text{for all } v \in \mathcal{S}_0 \,.$$

Therefore

$$\begin{aligned} b(w - u, w - u) &= b(w - u, w - u) - b(w - u, w - v) \\ &= b(w - u, v - u) \,. \end{aligned}$$

Applying the assumptions shows

$$\begin{aligned} \gamma_2 \|w - u\|_1^2 \leq |b(w - u, w - u)| &= |b(w - u, v - u)| \\ &\leq \gamma_1 \|w - u\|_1 \|v - u\|_1 \,, \end{aligned}$$

from which

$$\|w - u\|_1 \leq \frac{\gamma_1}{\gamma_2} \|v - u\|_1$$

follows. Since this holds for all $v \in \mathcal{S}_0$, the assertion of the lemma is proven.

Let us check whether the Assumptions 5.11 are fulfilled by the model problem (5.40)/(5.41). For (a) this follows from the Schwarzian inequality (C.16) with the norms

$$\|u\|_1 = \left(\int_\alpha^\beta (u^2 + u'^2) \, \mathrm{d}x \right)^{1/2} , \quad \|u\|_0 = \left(\int_\alpha^\beta u^2 \, \mathrm{d}x \right)^{1/2} ,$$

because

$$\left(\int_\alpha^\beta u' v' \, \mathrm{d}x \right)^2 \leq \left(\int_\alpha^\beta u'^2 \, \mathrm{d}x \right) \left(\int_\alpha^\beta v'^2 \, \mathrm{d}x \right) \leq \|u\|_1^2 \, \|v\|_1^2 \,.$$

The Assumption 5.11(b) can be derived from the inequality of the Poincaré-type

$$\int_\alpha^\beta v^2 \, \mathrm{d}x \leq (\beta - \alpha)^2 \int_\alpha^\beta v'^2 \, \mathrm{d}x \,,$$

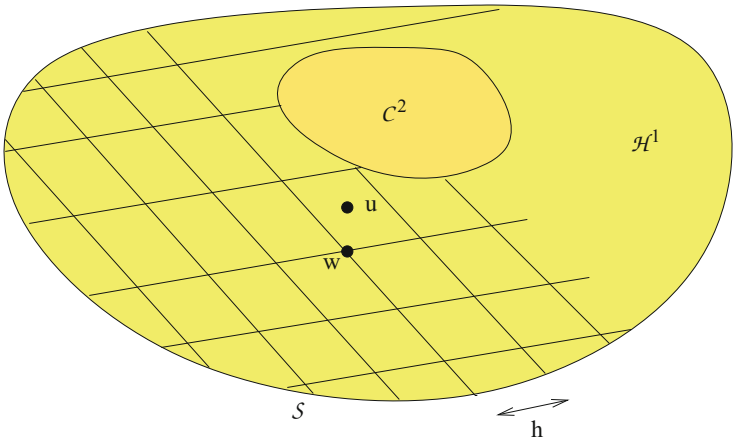

Fig. 5.12 Approximation spaces

which in turn is proven with the Schwarzian inequality (⟶ Exercise 5.10). Adding $\int v v'^2 \, dx$ on both sides leads to

$$\|v\|_1^2 \leq [(\beta - \alpha)^2 + 1] \, b(v, v) \, ,$$

from which the constant γ_2 of Assumption 5.11(b) results. Hence Céa's lemma applies to the model problem.

The next question is, how small the infimum in (5.49) may be. This is equivalent to the question, how close the subspace $\mathcal{S}_0$ can approximate the space $\mathcal{H}_0^1$ (⟶ Fig. 5.12). We will show that for hat functions and $\mathcal{S}_0$ from (5.47) the infimum is of the order $O(h)$. Again h denotes the maximum mesh size, and the notation w_h reminds us that the discrete solution depends on the grid with a spacing symbolized by h. To apply Céa's lemma, we need an upper bound for the infimum of $\|u - v\|_1$. Such a bound is found easily by a specific choice of v, which is taken as an arbitrary interpolating polygon u_I. Then by (5.49)

$$\|u - w_h\|_1 \leq \frac{\gamma_1}{\gamma_2} \inf_{v \in \mathcal{S}_0} \|u - v\|_1 \leq \frac{\gamma_1}{\gamma_2} \|u - u_\mathrm{I}\|_1 \, . \tag{5.50}$$

It remains to bound the error of interpolating polygons. This bound is provided by the following lemma, which is formulated for C^2-smooth functions u:

Lemma 5.13 (Error of an Interpolating Polygon) *For $u \in C^2$ let u_I be an arbitrary interpolating polygon and h the maximal distance between two consecutive nodes. Then*

(a) $\max\limits_x |u(x) - u_\mathrm{I}(x)| \leq \frac{h^2}{8} \max |u''(x)| \, ,$

(b) $\max\limits_x |u'(x) - u_\mathrm{I}'(x)| \leq h \max |u''(x)| \, .$

We leave the proof to the reader (⟶ Exercise 5.11). Lemma 5.13 asserts

$$\|u - u_1\|_1 = O(h)\,,$$

which together with (5.50) implies the claimed error statement

$$\|u - w_h\|_1 = O(h)\,. \tag{5.51}$$

Recall that this assertion is based on a continuous and $\mathcal{H}^1$-elliptic bilinear form and on hat functions φ_i. The $O(h)$-order in (5.51) is dominated by the unfavorable $O(h)$-order of the first-order derivative in Lemma 5.13(b). This low order is at variance with the actually observed $O(h^2)$-order attained by the approximation w_h itself (not its derivative). In fact, the square order holds. The final result is

$$\|u - w_h\|_0 \le Ch^2\|u\|_2 \tag{5.52}$$

for a constant C. This result is proven with the following lemma, which is based on a tricky idea due to Nitsche.

Lemma 5.14 (Nitsche) *Assume b is a symmetric bilinear form satisfying Assumptions 5.11, and u and w are defined as above. Then*

$$\|u - w\|_1 \le Kh^1\|f\|_0 \;\; \textit{implies} \;\; \|u - w\|_0 \le Ch^2\|f\|_0\,.$$

Proof Consider the auxiliary problem $Lz = \tilde{f} := u - w$, with weak version

$$b(z, \tilde{v}) = (\tilde{f}, \tilde{v})_0 \quad \text{for all } \tilde{v} \in \mathcal{H}_0^1\,,$$

which defines z. Choose specifically $\tilde{v} = u - w = \tilde{f}$. Then

$$b(z, u - w) = (u - w, u - w)_0 = \|u - w\|_0^2\,.$$

Invoking the error-projection property (5.48) we note

$$0 = b(u - w, v) = b(v, u - w) \quad \text{for all } v \in \mathcal{S}_0\,.$$

Subtracting this, yields

$$b(z - v, u - w) = \|u - w\|_0^2 \quad \text{for all } \; v \in \mathcal{S}_0\,.$$

We apply the continuity of b,

$$\|u - w\|_0^2 \le \gamma_1\|z - v\|_1\,\|u - w\|_1 \quad \text{for all } v \in \mathcal{S}_0\,,$$

and choose specifically v as the finite-element approximation of z. Then

$$\|u - w\|_0^2 \le \gamma_1 K_1 h^1 \|\tilde{f}\|_0 \cdot K_2 h^1 \|f\|_0 = Ch^2 \|u - w\|_0 \, \|f\|_0 \,,$$

from which the assertion follows.

This error of the order h^2 can be observed for the examples of Sect. 5.4, but not easily. The error is somewhat hidden among the other errors, namely, localization error, interpolation error, and the error of the time discretization.

The derivations of this section have been focused on the model problem (5.40)/(5.41) with a second-order differential equation and one independent variable $x\,(n = 1)$, and have been based on linear elements. Most of the assertions can be generalized to higher-order differential equations, to higher-dimensional domains $(n > 1)$, and to nonlinear elements. For example, in case the elements in $\mathcal{S}$ are polynomials of degree k, and the differential equation is of order $2l$, $\mathcal{S} \subseteq \mathcal{H}^l$, and the corresponding bilinear form on $\mathcal{H}^l$ satisfies the Assumptions 5.11 with norm $\| \; \|_l$, then the inequality

$$\|u - w_h\|_l \le Ch^{k+1-l} \|u\|_{k+1}$$

holds. This general statement includes for $k = 1$, $l = 1$ the special case of Eq. (5.52) discussed above. For the analysis of the general case, we refer to [79, 162]. This includes boundary conditions more general than the homogeneous Dirichlet conditions of (5.41).

5.6 Notes and Comments

On Sect. 5.1

As an alternative to piecewise defined finite elements one may use polynomials φ_j that are defined globally on $\mathcal{D}$, and that are pairwise orthogonal. Then the orthogonality is the reason for the vanishing of many integrals. Such type of methods are called spectral methods. Since the φ_i are globally smooth on $\mathcal{D}$, spectral methods can produce high accuracies. In other context, spectral methods were applied in [142]. For historical remarks on Ritz–Galerkin type methods, see [145].

Specifically designed basis functions can be generated by some low-dimensional approximation, comparable to PCA in finite dimensions ($\longrightarrow$ Exercise 2.16). Functions are suitable that represent preferred patterns of the solution. Then the number N of modes φ_i can be small. Such methods are described under the heading *principal orthogonal decomposition* (POD), or Karhunen–Loève expansion.

On Sect. 5.2

In the early stages of their development, finite-element methods have been applied intensively in structural engineering. In this field, stiffness matrix and mass matrix have a physical meaning leading to these names [382].

On Sect. 5.3

The approximation $\sum w_i(\tau)\varphi_i(x)$ for $\hat{y}$ is a one-dimensional finite-element approach. The geometry of the grid and the accuracy resemble the finite-difference approach. A two-dimensional approach as in

$$\sum w_i \varphi_i(x, \tau)$$

with two-dimensional hat functions and constant w_i is more involved and more flexible. Sections 5.3.2–5.3.4 widely follow [376].

On Sect. 5.4

For the calculation of the local integrals on an arbitrary triangle $\mathcal{D}_k$ consult the special FEM literature, such as [335]. In general an irregular triangulation better exploits the potential adaptivity of FEM. In particular, close to the barriers a fine mesh is required for high accuracy [304]. Since the gradient of u varies with time, a dynamic mesh refinement might be advisable, provided accuracy or stability do not deteriorate. For American options, boundary conditions $V = \Psi$ along the boundary are recommendable. For an illustration of assembling, see Topic 12 of the *Topics fCF*.

On Sect. 5.5

The assumption $u \in \mathcal{C}^2$ in Lemma 5.13 can be weakened to $u'' \in \mathcal{L}^2$ [351]. For domains $\mathcal{D} \in \mathbb{R}^2$ the claim of Lemma 5.13 holds analogously; then the second-order derivative u'' is replaced by the Hessian matrix of the second-order derivatives of u. This can be applied to mesh adaption, where one attempts to place nodes such that the Hessian is equilibrated across the mesh. The finite-dimensional function space $\mathcal{S}_0$ in (5.47) is assumed to be *sub*space of $\mathcal{H}_0^1$. Elements with this property are called *conforming elements*. A more accurate notation for $\mathcal{S}_0$ of (5.47) is $\mathcal{S}_0^1$. In the general case, conforming elements are characterized by $\mathcal{S}^l \subseteq \mathcal{H}^l$. In the representation of

v in Eq. (5.47) we avoid discussing the technical issue of how to organize different types of boundary conditions.

There are also smooth basis functions φ, for example, cubic Hermite polynomials. For sufficiently smooth solutions, such basis functions produce higher accuracy than hat functions do. For the accuracy of finite-element methods consult, for example, [2, 19, 53, 79, 162, 351].

On Other Methods

Finite-element methods are frequently used for approximating exotic options, in particular in multidimensional situations. For different types of options special methods have been developed. For applications, computational results and accuracies see also [2, 361, 362]. Front-fixing has been applied with finite elements in [188]. The accuracy aspect is also treated in [144]. Ritz–Galerkin methods are used with wavelet functions in [185, 263]; the latter paper is specifically devoted to stochastic volatility. A penalty approach with FEM is discussed in [230], where rectangular subdomains are furnished with basis functions as product of one-dimensional hat functions of the type $\varphi(x, y) = \varphi_i(x)\varphi_j(y)$.

5.7 Exercises

5.1 (Elliptical Probability Curves)
Suppose the situation of two asset prices $S_1(t)$ and $S_2(t)$ for $t > 0$ governed by GBM (3.35), with initial price point $(S_1(0), S_2(0))$. Barriers of a barrier option can be aligned such that the probability of $(S_1(t), S_2(t))$ reaching the barrier has the same constant value. Define $Y_1 := \log S_1$, $Y_2 := \log S_2$.

(a) Show that the curve of constant probability in the (Y_1, Y_2)-plane has an elliptical shape.
(b) Let the covariance matrix be

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}.$$

Calculate its eigenvalues and eigenvectors.
(c) Sketch representative ellipses in a (Y_1, Y_2)-plane. How do they depend on ρ?

5.2 (Cubic B-Spline)
Suppose an equidistant partition of an interval be given with mesh size $h = x_{k+1} - x_k$. Cubic B-splines have a support of four subintervals. In each subinterval the spline is a piece of polynomial of degree three. Apart from special boundary splines, the

cubic B-splines φ_i are determined by the requirements

$$\begin{aligned} \varphi_i(x_i) &= 1 \\ \varphi_i(x) &\equiv 0 \quad \text{for } x < x_{i-2} \\ \varphi_i(x) &\equiv 0 \quad \text{for } x > x_{i+2} \\ \varphi &\in \mathcal{C}^2(-\infty, \infty)\,. \end{aligned}$$

To construct these φ_i proceed as follows:

(a) Construct a spline $S(x)$ that satisfies the above requirements for the special nodes

$$\tilde{x}_k := -2 + k \quad \text{for } k = 0, 1, \ldots, 4\,.$$

(b) Find a transformation $T_i(x)$, such that $\varphi_i = S(T_i(x))$ satisfies the requirements for the original nodes.
(c) For which i, j does $\varphi_i \varphi_j = 0$ hold?

5.3 (Finite-Element Matrices)
For the hat functions φ from Sect. 5.2 calculate for arbitrary subinterval $\mathcal{D}_k$ all nonzero integrals of the form

$$\int \varphi_i \varphi_j \,\mathrm{d}x, \quad \int \varphi_i' \varphi_j \,\mathrm{d}x, \quad \int \varphi_i' \varphi_j' \,\mathrm{d}x$$

and represent them as local 2×2 matrices.

5.4 (Calculating Options with Finite Elements)
Design an algorithm for the pricing of standard options by means of finite elements. To this end proceed as outlined in Sect. 5.3. Start with a simple version using an equidistant discretization step Δx. If this is working properly change the algorithm to a version with nonequidistant x-grid. Distribute the nodes x_i closer around $x = 0$. Always place a node at the strike.

5.5 (Black-Scholes Equation in Divergence-Free Form)

(a) Prove the equivalence of (5.31) and (5.32), where D and b are given by (5.33). Specialize this to the one-dimensional case of the Black–Scholes equation.
(b) Show

$$b^{tr} \nabla u + ru = \nabla \cdot (bu) + \gamma u$$

and determine γ for the two-dimensional case, and for the Black–Scholes equation.

(c) With the transformation

$$x := \log(\frac{S_1}{K_1}),\ y := \log(\frac{S_2}{K_2})$$

and writing $u(x, y, t)$ for V leads to the PDE

$$u_t + \frac{1}{2}\sigma_1^2 u_{xx} + (r - \delta_1 - \frac{1}{2}\sigma_1^2)u_x - ru$$
$$+ \frac{1}{2}\sigma_2^2 u_{yy} + (r - \delta_2 - \frac{1}{2}\sigma_2^2)u_y + \rho\sigma_1\sigma_2 u_{xy} = 0\,.$$

What are the matrix D and the vector b such that we arrive at (5.32)?

5.6 (Outward Normals)
The boundary $\partial\mathcal{D}$ of the trapezoidal domain $\mathcal{D}$ in Fig. 5.5 consists of four straight lines. What are the four unit outward vectors n orthogonal to $\partial\mathcal{D}$? Give a parameter representation of the boundary.

5.7 (Gradient on a Triangle)
Consider hat functions φ on a triangular element $\mathcal{D}_k$ with vertex nodes numbers $\mathcal{I}_k = \{i, j, l\}$, and the local plane on $\mathcal{D}_k$ represented by

$$w(x, y) = w_i\varphi_i(x, y) + w_j\varphi_j(x, y) + w_l\varphi_l(x, y)\,.$$

(a) In the three-dimensional (x, y, w)-space let the plane $w(x, y) = c_1 + c_2\,x + c_3\,y$ interpolate the three points $(x_i,\ y_i,\ w_i)$, $i = 1, 2, 3$ (local node numbering). That is,

$$\begin{pmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{pmatrix} \begin{pmatrix} c_1 \\ c_2 \\ c_3 \end{pmatrix} = \begin{pmatrix} w_1 \\ w_2 \\ w_3 \end{pmatrix},$$

shortly $Ac = w$. Establish a formula for the gradient $\nabla w = (c_2, c_3)^{tr}$, showing that there is a (2×3)-matrix G_k such that

$$\nabla w = G_k w\,.$$

Hint: Use Cramer's rule; $|F_k|$ is the area of the triangle, where

$$F_k := \frac{1}{2}\det(A)\,.$$

(b) Show

$$(\nabla\varphi_i \,|\, \nabla\varphi_j \,|\, \nabla\varphi_l) = G_k \,.$$

(c) Show

$$\int_{\mathcal{D}_k} \nabla\varphi_i^{tr} \nabla\varphi_j \,\mathrm{d}x\,\mathrm{d}y = \nabla\varphi_i^{tr} \nabla\varphi_j \,|F_k| \,,$$

and all nine integrals of the element stiffness matrix are obtained by

$$|F_k| G_k^{tr} G_k \,.$$

5.8 (Assembling)
Consider the domain $\mathcal{D} := \{(x, y) \,|\, x \geq 0,\ y \geq 0,\ \ 1 \leq x + y \leq 2\}$ tiled by 12 triangles $\mathcal{D}_k$, where triangles and vertices are numbered as in Fig. 5.13.

(a) Set up the index set $\mathcal{I}$ with entries $\mathcal{I}_k = \{i_k, j_k, l_k\}$, which assigns node numbers to the kth triangle, for $1 \leq k \leq 12$.
(b) Formulate the assembling algorithm that builds up the global stiffness matrix out of the element stiffness matrices

$$\begin{pmatrix} s_{11}^{(k)} & s_{12}^{(k)} & s_{13}^{(k)} \\ s_{21}^{(k)} & s_{22}^{(k)} & s_{23}^{(k)} \\ s_{31}^{(k)} & s_{32}^{(k)} & s_{33}^{(k)} \end{pmatrix}$$

for a general index set $\mathcal{I}$ and $1 \leq k \leq m$.

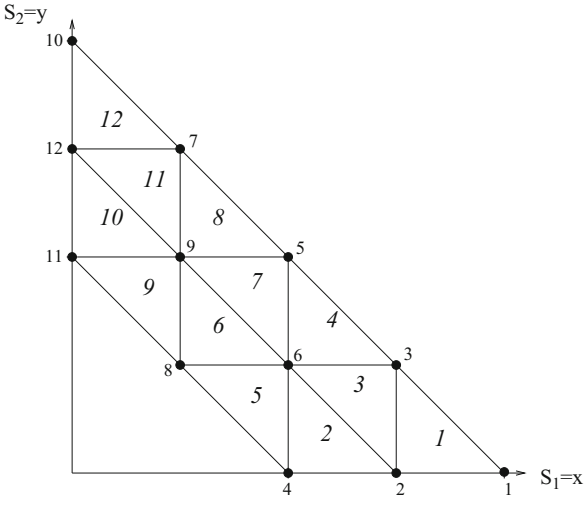

Fig. 5.13 Specific triangulation and numbering, see Exercise 5.8

(c) The example of Fig. 5.13 leads to a banded stiffness matrix. What is the bandwidth?

5.9 (Variable Volatility (Project))
For variable volatility $\sigma(S,t)$ and constant K, T, r, δ, PDEs of the type

$$\frac{\partial y}{\partial \tau} - \frac{1}{2}\hat{\sigma}^2(x,\tau)\left(\frac{\partial^2 y}{\partial x^2} - \frac{1}{4}y\right) = 0$$

are to be solved, with $\tau = T - t$ and transformations $S \leftrightarrow x$, $V \leftrightarrow y$ from the Black–Scholes model given by (A.25), (A.26); consult Appendix A.6.

(a) For an American put, apply these transformations to derive from $V(S,t) \geq (K-S)^+$ an inequality $y(x,\tau) \geq g(x,\tau)$.
(b) Carry out the finite-element formulation for the linear complementarity problem analogously as in Sect. 5.3.4.
(c) Integrals will include local integrals

$$\int \sigma^2(x,\tau)\varphi_i\varphi_j \,\mathrm{d}x\,, \quad \int \sigma^2(x,\tau)\varphi_i'\varphi_j \,\mathrm{d}x\,.$$

Apply Simpson's quadrature rule

$$\int_a^b f(x)dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

to approximate the above local integrals.
(d) Set up a finite-element code, and test it with the artificial function [128]

$$\sigma(S) := 0.3 - \frac{0.2}{\log(S/K)^2 + 1}\,.$$

5.10 Assume a function $v(\zeta)$ with $\alpha \leq \zeta \leq \beta$ and $v(\alpha) = 0$.

(a) Show

$$(v(\zeta))^2 \leq (\zeta - \alpha)\int_\alpha^\zeta (v'(x))^2 \,\mathrm{d}x\,.$$

Hint: Recall $v(\zeta) = \int_\alpha^\zeta v'(x)\,\mathrm{d}x$, and apply the Schwarzian inequality (C.16).
(b) Use (a) to show

$$\int_\alpha^\beta (v(\zeta))^2 \,\mathrm{d}\zeta \leq \frac{1}{2}(\beta-\alpha)^2 \int_\alpha^\beta (v'(x))^2 \,\mathrm{d}x\,.$$

5.11 Prove Lemma 5.13, and for $u \in \mathcal{C}^2$ the assertion $\|u - w_h\|_1 = O(h)$.

Chapter 6
Pricing of Exotic Options

Chapter 4 discussed the pricing of vanilla options (standard options) by means of finite differences. The methods were based on the simple partial differential equation (4.2),

$$\frac{\partial y}{\partial \tau} = \frac{\partial^2 y}{\partial x^2},$$

which was obtained from the Black–Scholes equation (4.1) for $V(S,t)$ via the transformations (4.3). These transformations exploit the simple structure of the Black–Scholes operator and rely on the assumption of constant coefficients.

Exotic options lead to partial differential equations that are not of the simple structure of the basic Black–Scholes equation (4.1). In the general case, the transformations (4.3) are no longer useful and the PDEs must be solved directly. Thereby numerical instabilities or spurious solutions may occur that do not play any role for the methods of Chap. 4. To cope with the "new" difficulties, Chap. 6 introduces ideas and tools not needed in Chap. 4. Exotic options often involve higher-dimensional problems. This significantly adds to the complexity. An exhaustive discussion of the wide field of exotic options is beyond the scope of this book. The aim of this chapter will not be to formulate algorithms, but to give an outlook on several relevant aspects of computation, and on phenomena of stability. In this chapter, we still stick to the GBM model and move within the Black–Scholes world; for more general models see Chap. 7.

Sections 6.1 and 6.2 give a brief overview on important types of exotic options. Section 6.3 introduces approaches for path-dependent options, with the focus on Asian options. Then numerical aspects of convection-diffusion problems are discussed (in Sect. 6.4), and upwind schemes are analyzed (in Sect. 6.5). After these preparations, Sect. 6.6 arrives at a state of the art high-resolution method. Finally, Sect. 6.7 addresses penalty methods, with application to two-asset options.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_6

6.1 Exotic Options

So far, this book has mainly concentrated on standard options. These are the American or European call or put options with vanilla payoff functions (1.1) or (1.2) as discussed in Sect. 1.1, based on a single underlying asset. The options traded on official exchanges are mainly standard options; there are market prices quoted in relevant newspapers.

All nonstandard options are called exotic options. That is, at least one of the features of a standard option is violated. One of the main possible differences between standard and exotic options lies in the payoff; examples are given in this section. Another extension from standard to exotic is an increase in the dimension, from single-factor to multifactor options; this will be discussed in Sect. 6.2. For exotic options, the distinctions between put and call, and between European and American options remain valid.

Financial institutions have been imaginative in designing exotic options to meet the needs of clients. Many of the products have a highly complex structure. Exotic options are traded outside the exchanges (OTC), and often they are illiquid and no market prices are available. Then exotic options must be priced based on models. In general, their parameters are taken from the results obtained when standard options with comparable terms are calibrated to market prices. The simplest models extend the Black–Scholes model, which is summarized by Assumptions 1.2.

Next we list some important types of exotic options. For more explanation we refer to [191, 375].

Binary Option Binary options (or digital options) have a discontinuous payoff. For example, a binary put has the payoff

$$\Psi(S) := c \cdot \begin{cases} 1 & \text{if } S < K \\ 0 & \text{if } S \geq K \end{cases}$$

for a fixed amount c. See Fig. 4.21 for an illustration of a binary call, and Sect. 3.5.5 for a two-dimensional example.

Chooser Option After a specified period of time the holder of a chooser option can choose whether the option is a call or a put. The value of a chooser option at this time is

$$\max\{V_C, V_P\}\,.$$

Compound Option Compound options are options on options. Depending on whether the options are put or call, there are four main types of compound options. For example, the option may be a call on a call.

6.1.1 Path-Dependent Options

Options with payoff depending not only on the current value S_T but also on the path of S_t for previous times $t < T$ are called *path dependent*. Important path-dependent options are the *barrier option*, the *lookback option*, and the *Asian option*.

Barrier Option For a barrier option the payoff is contingent on the underlying asset's price S_t reaching a certain threshold value B, which is called barrier. Barrier options can be classified depending on whether S_t reaches B from above *(down)* or from below *(up)*. Another feature of a barrier option is whether it ceases to exist when B is reached *(knock out)*, or conversely comes into existence *(knock in)*. Obviously, for a down option, $S_0 > B$ and for an up option $S_0 < B$. Depending on whether the barrier option is a put or a call, several different types are possible. For example, the payoff of a European *down-and-out* call is

$$V_T = \begin{cases} (S_T - K)^+ & \text{in case } S_t > B \text{ for all } t\,, \\ 0 & \text{in case } S_t \le B \text{ for some } t\,. \end{cases}$$

In the Black–Merton–Scholes framework, the value of the option before the barrier has been triggered still satisfies the Black–Scholes equation. The details of the barrier feature come in through the specification of boundary conditions [375]. An example of an up-and-out call is illustrated in Fig. 7.3, and a two-asset double barrier is discussed in Example 5.6.

Lookback Option The payoff of a lookback option depends on the maximum or minimum value the asset price S_t reaches during the life of the option. For example, the payoff of a lookback option is

$$\max_t S_t - S_T\,.$$

Average Option/Asian Option The payoff from an Asian option depends on the average price of the underlying asset. This will be discussed in more detail in Sect. 6.3.

The exotic options of the above short list gain in complexity when they are multifactor options.

6.1.2 Pricing of Exotic Options

Several types of exotic options can be reduced to the Black–Scholes equation. In these cases the methods of Chap. 4 or Chap. 5 are adequate. In particular, barrier options under GBM are close to the standard options. For a knock-out option with barrier B, a boundary condition will be $V(B, t) = 0$, which is part of (4.28). Since

their numerical treatment is widely analogous, we will not touch barrier options specifically.

For a number of options of the European type the Black–Scholes evaluation formula (A.13)–(A.17) can be applied. For related reductions of exotic options we refer to [191, 234, 376]. Approximations are possible with binomial methods or with Monte Carlo simulation. The Algorithm 3.6 applies, only the calculation of the payoff (step 2) must be adapted to the exotic option.

6.2 Options Depending on Several Assets

The options listed in Sect. 6.1 depend on one underlying asset. Options depending on several assets are discussed next. Two large groups of multifactor options are the *rainbow options* and the *baskets*. The subdivision into the groups is by their payoff. Assume n underlying assets with prices $S_1, \ldots, S_n$. Different from the notation prevailing in previous chapters, the index refers to the number of the asset. Recall again that two examples of exotic options with two underlyings occurred earlier in this text: Example 3.9 of a binary put, and Sect. 5.4 with a basket-barrier call.

Rainbow options compare the value of individual assets [342]. Examples of payoffs include

$\max(S_1, \ldots, S_n)$	"n-color better-of option,"
$\min(S_1, S_2)$	"two-color worse-of option,"
$(S_2 - S_1)^+$	"outperformance option,"
$(\min(S_1 - K, \ldots, S_n - K))^+$	"min call option,"
$(S_2 - S_1 - K)^+$	"spread call."

Weights are possible too, for instance, in $(c_1 S_2 - c_2 S_1)^+$. The outperformance option is also called spread option. Figure 6.1 (top) illustrates the payoff of a min put, and Fig. 6.2 (bottom) the payoff of a max call. A basket is an option with payoff depending on a portfolio of assets. An example is the payoff of a basket call,

$$\left(\sum_{i=1}^{n} c_i S_i - K \right)^+ ,$$

where the weights c_i are given by the portfolio. To gain a better feeling for such kind of options, it is recommendable to sketch the above payoffs for $n = 2$.

For the pricing of multifactor options the instruments introduced in the previous chapters apply. This holds for the four large classes of methods discussed before, namely, PDE methods, tree methods, evaluation of integrals by quadrature, and Monte Carlo methods. Each class subdivides into further methods.

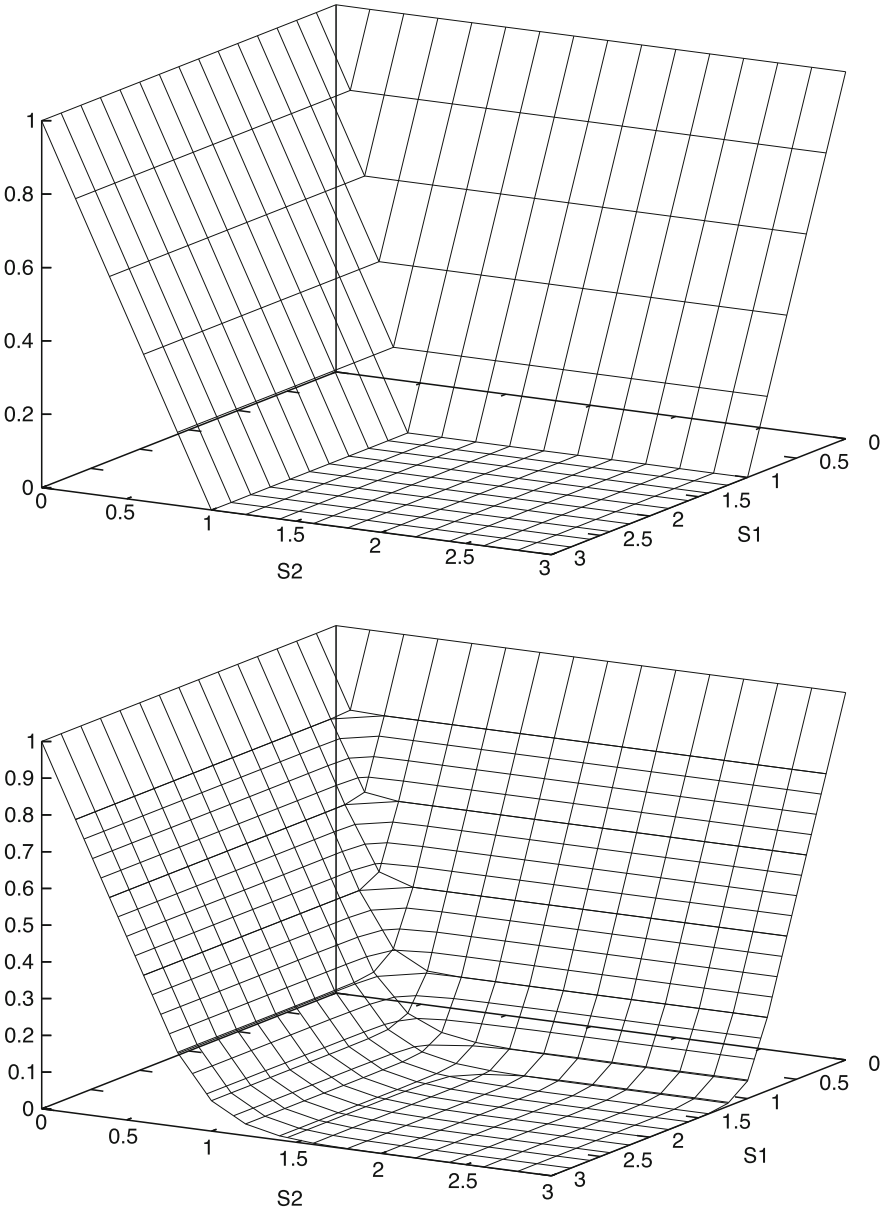

Fig. 6.1 Rainbow option of a put on the minimum of two assets; *top*: payoff $\Psi(S_1, S_2) = (1 - \min(S_1, S_2))^+$; *bottom*: $V(S_1, S_2, 0)$ approximated by a binomial method, level curves for slices with constant values of S_1, S_2, V

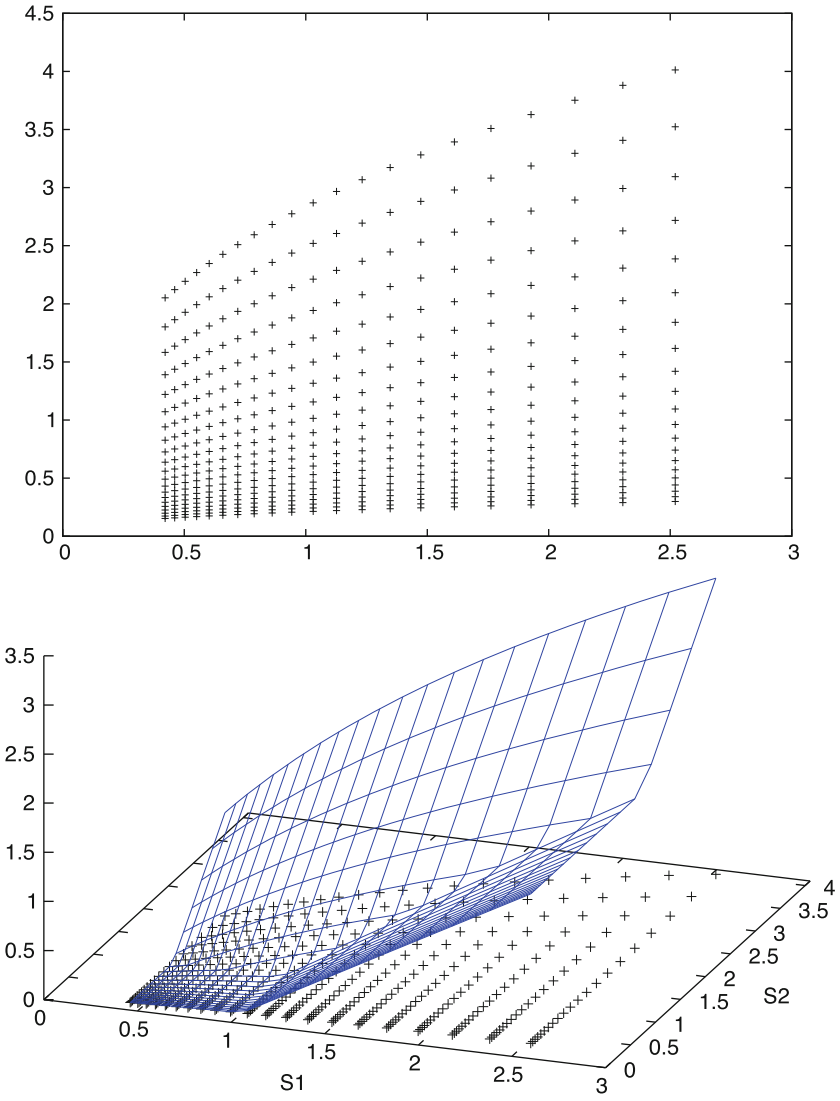

Fig. 6.2 Max call, with payoff $\Psi(S_1, S_2) = (\max(S_1, S_2) - K)^+$; parameters: $K = T = 1$, $\mu_1 = \mu_2 = r = 0.1$, $\sigma_1 = 0.2$, $\sigma_2 = 0.3$, $\rho = 0$, $\delta_1 = \delta_2 = 0$; *top*: (S_1, S_2)-plane with the grid of the tree for the payoff, $t = T$, with $M = 20$; *bottom*: the payoff $\Psi(S_1, S_2)$ above the tree

The dimension n is crucial for the choice of an appropriate method. For large values of n, in particular PDE methods suffer from the curse of dimension ($\longrightarrow$ Exercise 4.21). At present state it is not possible to decide, above which threshold level of n standard discretizations are too expensive.

PDE methods require relevant PDEs *and* boundary conditions. Often a Black–Merton–Scholes scenario is assumed. To extend the one-factor model, an appropriate generalization of geometric Brownian motion is needed. We begin with the two-factor model, with the prices S_1 and S_2 of the two assets. Then the assumption of a constant-coefficient GBM is expressed as

$$\begin{aligned} \mathrm{d}S_1 &= \mu_1 S_1 \,\mathrm{d}t + \sigma_1 S_1 \,\mathrm{d}W^{(1)}\,, \\ \mathrm{d}S_2 &= \mu_2 S_2 \,\mathrm{d}t + \sigma_2 S_2 \,\mathrm{d}W^{(2)}\,, \\ \mathsf{E}(\,\mathrm{d}W^{(1)}\,\mathrm{d}W^{(2)}) &= \rho \,\mathrm{d}t\,, \end{aligned} \tag{6.1}$$

where ρ is the correlation between the two assets, $-1 \le \rho \le 1$. Note that the third equation in (6.1) is equivalent to $\mathsf{Cov}(\,\mathrm{d}W^{(1)},\,\mathrm{d}W^{(2)}) = \rho\,\mathrm{d}t$, because $\mathsf{E}(\,\mathrm{d}W^{(1)}) = \mathsf{E}(\,\mathrm{d}W^{(2)}) = 0$. The correlation ρ is given by the covariance of the returns $\frac{dS}{S}$, since

$$\mathsf{Cov}\left(\frac{\mathrm{d}S_1}{S_1}, \frac{\mathrm{d}S_2}{S_2}\right) = \mathsf{E}(\sigma_1 \,\mathrm{d}W^{(1)} \sigma_2 \,\mathrm{d}W^{(2)}) = \rho\sigma_1\sigma_2 \,\mathrm{d}t\,. \tag{6.2}$$

Compared to the more general system (1.57), the version (6.1) with correlated Wiener processes has pulled out the scaling by the volatilities σ_1, σ_2. Then, following Sect. 2.3.4 and Exercise 2.17, the correlated Wiener processes can be decoupled by Cholesky decomposition of the correlation matrix

$$\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}.$$

This leads to

$$\begin{aligned} \mathrm{d}W^{(1)} &= \mathrm{d}Z_1\,, \\ \mathrm{d}W^{(2)} &= \rho\,\mathrm{d}Z_1 + \sqrt{1-\rho^2}\,\mathrm{d}Z_2\,, \end{aligned} \tag{6.3}$$

where Z_1 and Z_2 are independent standard normally distributed processes. This was already used in (3.35). The resulting two-dimensional Black–Scholes equation was applied in Sect. 5.4, see Eq. (5.31). This is derived by the two-dimensional version of the Itô-Lemma (⟶ Appendix B.2) and by a no-arbitrage argument. The resulting PDE (5.31) has independent variables (S_1, S_2, t). Usually, the time variable is not counted when the dimension is discussed. In this sense, the PDE (5.31) is two-dimensional, whereas the classic Black–Scholes PDE (1.5) is considered as one-dimensional.

The general n-factor model is analogous. The appropriate GBM model is a straightforward generalization of (6.1),

$$\begin{aligned} \mathrm{d}S_i &= (\mu_i - \delta_i) S_i \,\mathrm{d}t + \sigma_i S_i \,\mathrm{d}W^{(i)}\,, \quad i = 1, \ldots, n\,, \\ \mathsf{E}(\mathrm{d}W^{(i)}\mathrm{d}W^{(j)}) &= \rho_{ij}\,\mathrm{d}t\,, \quad i,j = 1, \ldots, n\,, \end{aligned} \tag{6.4}$$

where ρ_{ij} is the correlation between asset i and asset j, and δ_i denotes a dividend flow rate paid by the ith asset. For a simulation of such a stochastic vector process see Sect. 2.3.4. The Black–Scholes-type PDE of the model (6.4) is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sum_{i,j=1}^{n} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{n}(r-\delta_i)S_i\frac{\partial V}{\partial S_i} - rV = 0\,. \tag{6.5}$$

The derivation uses the general Itô formula (B.17), with $m = 1$ ($\longrightarrow$ Exercise 6.1).

Boundary conditions depend on the specific type of option. For example in the "two-dimensional" situation in (S_1, S_2, t)-space, one boundary can be defined by the plane $S_1 = 0$ and the other by the plane $S_2 = 0$. It may be appropriate to apply the Black–Scholes vanilla formulas (A.13)–(A.17) along these planes, or to define one-dimensional sub-PDEs only for the purpose to calculate the values of $V(S_1, 0, t)$ and $V(0, S_2, t)$ along the boundary planes.

After the PDE with boundary conditions is set up, solutions are approximated by numerical methods. Standard discretizations are straightforward and work for small n. As a rule of thumb, for $n = 2$ and $n = 3$, such elementary PDE approaches are competitive to Monte Carlo. For large n, sparse-grid technology or multigrid are better choices, see the references in Sect. 3.5.1 and at the end of Chap. 4. Generally in a multidimensional situation, finite elements are recommendable. But FE methods suffer from the curse of dimension too. Irregular grids have been applied successfully [36].

For **tree methods**, the binomial method can be generalized canonically [48] ($\longrightarrow$ Exercise 6.2). But already for $n = 2$ the recombining standard tree with M time levels requires $\frac{1}{3}M^3 + O(M^2)$ nodes, and for $n = 3$ the number of nodes is of the order $O(M^4)$. Tree methods also suffer from the curse of dimension. But obviously not all of the nodes of the canonical binomial approach are needed. The ultimate aim is to approximate the lognormal distribution, and this can be done with fewer nodes. Nodes in $\mathbb{R}^n$ should be constructed in such a way that the number of nodes grows comparably slower than the quality of the approximation of the distribution function. An example of a two-dimensional approach is presented in [251]. Generalizing the trinomial approach to higher dimensions is not recommendable because of storage requirements, but other geometrical structures as icosahedral volumes can be applied. For different tree approaches, see [266]. For a convergence analysis of tree methods, and for an extension to Lévy processes, consult [136, 255]. A tree approach that makes use of decoupling (similar as in Sect. 2.3.4) has shown to be favorable in multidimensional cases [228].

An advantage of tree methods and of **Monte Carlo methods** is that no boundary conditions are needed. The essential advantage of MC methods is that they are much less affected by high dimensions, see the notes on Sect. 3.6. A correlation is achieved by $\mathrm{d}W = L\mathrm{d}Z$, where LL^{tr} is the Cholesky decomposition of the ρ-matrix. An example of a five-dimensional American-style option is calculated in [59, 247], and one with dimension 30 in [206]. It is most inspiring to perform Monte

Carlo experiments on exotic options. For European-style options, this amounts to a straightforward application of Sect. 3.5 (⟶ Exercise 6.3).

6.3 Asian Options

The price of an Asian option[1] depends on the average price of the underlying and hence on the history of S_t. We choose this type of option to discuss some strategies of how to handle path-dependent options. Let us first define different types of Asian options via their payoff.

6.3.1 The Payoff

There are several ways how an average of past values of S_t can be formed. If the price S_t is observed at discrete time instances t_i, say equidistantly with time interval $h := T/n$, one obtains a times series $S_{t_1}, S_{t_2}, \ldots, S_{t_n}$. An obvious choice of average is the arithmetic mean

$$\frac{1}{n}\sum_{i=1}^{n} S_{t_i} = \frac{1}{T}\, h \sum_{i=1}^{n} S_{t_i}\,.$$

If we imagine the observation as continuously sampled in the time period $0 \le t \le T$, the above mean corresponds to the integral

$$\widehat{S} := \frac{1}{T}\int_0^T S_t \,\mathrm{d}t\,. \tag{6.6}$$

The arithmetic average is used mostly. Sometimes the geometric average is applied, which can be expressed as

$$\left(\prod_{i=1}^{n} S_{t_i}\right)^{1/n} = \exp\left(\frac{1}{n}\log\prod_{i=1}^{n} S_{t_i}\right) = \exp\left(\frac{1}{n}\sum_{i=1}^{n}\log S_{t_i}\right)\,.$$

Hence the continuously sampled geometric average of the price S_t is the integral

$$\widehat{S}_{\mathrm{g}} := \exp\left(\frac{1}{T}\int_0^T \log S_t \,\mathrm{d}t\right)\,.$$

[1] Again, the name has no geographical relevance.

The averages $\widehat{S}$ and $\widehat{S}_{\mathrm{g}}$ are formulated for the time period $0 \le t \le T$, which corresponds to a European option. To allow for early exercise at time $t < T$, $\widehat{S}$ and $\widehat{S}_{\mathrm{g}}$ are modified appropriately, for instance to

$$\widehat{S} := \frac{1}{t} \int_0^t S_\theta \, \mathrm{d}\theta \, .$$

With an average value $\widehat{S}$ like the arithmetic average of (6.6) the payoff of Asian options can be written conveniently:

Definition 6.1 (Asian Option) With an average $\widehat{S}$ of the price evolution S_t the payoff functions of Asian options are defined as

$$\begin{array}{ll} (\widehat{S} - K)^+ & \text{average price call,} \\ (K - \widehat{S})^+ & \text{average price put,} \\ (S_T - \widehat{S})^+ & \text{average strike call,} \\ (\widehat{S} - S_T)^+ & \text{average strike put.} \end{array}$$

The price options are also called *rate options*, or *fixed strike options*; the strike options are also called *floating strike options*. Compared to the vanilla payoffs of (1.1) and (1.2), for an Asian price option the average $\widehat{S}$ replaces S whereas for the Asian strike option $\widehat{S}$ replaces K. The payoffs of Definition 6.1 form surfaces above the quadrant $S > 0, \widehat{S} > 0$. The reader may visualize these payoff surfaces.

6.3.2 Modeling in the Black–Scholes Framework

The above averages can be expressed by means of the integral

$$A_t := \int_0^t f(S_\theta, \theta) \, \mathrm{d}\theta \, , \tag{6.7}$$

where the function $f(S, t)$ depends on the type of chosen average. In particular $f(S, t) = S$ corresponds to the continuous arithmetic average (6.6), up to scaling by the length of interval. For Asian options the price V is a function of S, A and t, which we write $V(S, A, t)$. To derive a partial differential equation for V using a generalization of Itô's Lemma we require a differential equation for A. This is given

by (6.7). Compare with (1.44) to see[2]

$$\begin{aligned} \mathrm{d}A &= a_A(t)\,\mathrm{d}t + b_A\,\mathrm{d}W_t\,, \\ \text{with} \quad a_A(t) &:= f(S_t,t)\,, \quad b_A := 0\,. \end{aligned}$$

For S_t the standard GBM of (1.47) is assumed. By the multidimensional version (B.17) of Itô's Lemma adapted to $Y_t := V(S_t, A_t, t)$, the two terms in (1.60) or (1.61) that involve b_A as factors to $\frac{\partial V}{\partial A}$, $\frac{\partial^2 V}{\partial A^2}$ vanish. Accordingly,

$$\mathrm{d}V_t = \left(\frac{\partial V}{\partial t} + \mu S \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + f(S,t)\frac{\partial V}{\partial A} \right) \mathrm{d}t + \sigma S \frac{\partial V}{\partial S}\,\mathrm{d}W_t\,.$$

The derivation of the Black–Scholes-type PDE goes analogously as outlined in Appendix A.4 for standard options and results in

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} + f(S,t)\frac{\partial V}{\partial A} - rV = 0\,. \tag{6.8}$$

Compared to the original vanilla version (1.5), only one term in (6.8) is new, namely,

$$f(S,t)\,\frac{\partial V}{\partial A}\,.$$

As we will see below, the lack of a second-order derivative with respect to A can cause numerical difficulties. The transformations (4.3) cannot be applied advantageously to (6.8). As an alternative to the definition of A_t in (6.7), one can scale by t. This leads to a different "new term" [⟶ Exercise 6.4(e)].

6.3.3 *Reduction to a One-Dimensional Equation*

Solutions of (6.8) are defined on the domain

$$S > 0\,,\ A > 0\,,\ 0 \le t \le T$$

of the (S, A, t)-space. The extra A-dimension leads to significantly higher costs when (6.8) is solved numerically. This is the general situation. But in some cases it is possible to reduce the dimension. Let us discuss an example, concentrating on the case $f(S,t) = S$ of the arithmetic average.

[2] The ordinary integral A_t is random but has zero quadratic variation [340].

We consider a European arithmetic average strike (floating strike) call with payoff

$$\left(S_T - \frac{1}{T}A_T\right)^+ = S_T\left(1 - \frac{1}{TS_T}\int_0^T S_\theta \,\mathrm{d}\theta\right)^+ .$$

An auxiliary variable R_t is defined by

$$R_t := \frac{1}{S_t}\int_0^t S_\theta \,\mathrm{d}\theta \,,$$

and the payoff is rewritten

$$S_T\left(1 - \frac{1}{T}R_T\right)^+ = S_T \cdot \text{function}(R_T, T) \,. \tag{6.9}$$

This motivates trying a separation of the solution in the form

$$V(S, A, t) = S \cdot H(R, t) \tag{6.10}$$

for some function $H(R, t)$. In this role, R is an independent variable. From (6.9) the payoff follows:

$$H(R_T, T) = (1 - \tfrac{1}{T}R_T)^+ \,. \tag{6.11}$$

Substituting the separation ansatz (6.10) into the PDE (6.8) leads to a PDE for H,

$$\frac{\partial H}{\partial t} + \frac{1}{2}\sigma^2 R^2 \frac{\partial^2 H}{\partial R^2} + (1 - rR)\frac{\partial H}{\partial R} = 0 \tag{6.12}$$

[⟶ Exercise 6.4(c)]. To solve this PDE, boundary conditions are required. Their choice in general is not unique. The following considerations from [376] suggest boundary conditions.

A right-hand boundary condition for $R \to \infty$ follows from the payoff (6.11), which implies $H(R_T, T) = 0$ for $R_T \to \infty$. The integral $A_t = S_t R_t$ is bounded, hence $S \to 0$ for $R \to \infty$. For $S \to 0$ a European call option is not exercised, which suggests to prescribe the boundary condition

$$H(R, t) = 0 \quad \text{for } R \to \infty \text{ and all } t \,. \tag{6.13}$$

At the left-hand boundary $R = 0$ we encounter more difficulties. Note that the integral R_t satisfies the SDE

$$\mathrm{d}R_t = (1 + (\sigma^2 - \mu)R_t)\,\mathrm{d}t - \sigma R_t \,\mathrm{d}W_t$$

[⟶ Exercise 6.4(d)]. Even if $R_0 = 0$ holds, this SDE shows that $\mathrm{d}R_0 = \mathrm{d}t$ and R_t will not stay at 0. So there is no reason to expect $R_T = 0$, and the value of the payoff cannot be predicted. Another kind of boundary condition is required.

To this end, we start from the PDE (6.12), which for $R \to 0$ is equivalent to

$$\frac{\partial H}{\partial t} + \frac{1}{2}\sigma^2 R^2 \frac{\partial^2 H}{\partial R^2} + \frac{\partial H}{\partial R} = 0\,.$$

Assuming that H is bounded, one can prove that the term

$$R^2 \frac{\partial^2 H}{\partial R^2}$$

vanishes for $R \to 0$. The resulting boundary condition is

$$\frac{\partial H}{\partial t} + \frac{\partial H}{\partial R} = 0 \quad \text{for} \quad R \to 0\,. \tag{6.14}$$

The vanishing of the second-order derivative term is shown by contradiction: Assuming a nonzero value of $R^2 \frac{\partial^2 H}{\partial R^2}$ leads to

$$\frac{\partial^2 H}{\partial R^2} = O\left(\frac{1}{R^2}\right)\,,$$

which can be integrated twice to

$$H = O(\log R) + c_1 R + c_2\,.$$

This contradicts the boundedness of H for $R \to 0$.

For a numerical realization of the boundary condition (6.14) in the finite-difference framework of Chap. 4, we may use the second-order formula

$$\left.\frac{\partial H}{\partial R}\right|_{0,\nu} = \frac{-3H_{0,\nu} + 4H_{1,\nu} - H_{2,\nu}}{2\Delta R} + O(\Delta R^2)\,. \tag{6.15}$$

The indices have the same meaning as in Chap. 4. In summary, the boundary-value problem of PDEs is

$$\begin{aligned}
&\frac{\partial H}{\partial t} + \frac{1}{2}\sigma^2 R^2 \frac{\partial^2 H}{\partial R^2} + (1 - rR)\frac{\partial H}{\partial R} = 0\,,\\
&H(R_T, T) = \left(1 - \tfrac{R_T}{T}\right)^+\,,\\
&H = 0 \quad \text{for } R \to \infty\,,\\
&\frac{\partial H}{\partial t} + \frac{\partial H}{\partial R} = 0 \quad \text{for } R = 0\,.
\end{aligned} \tag{6.16}$$

Solving this problem numerically for $0 \le t \le T$, $R \ge 0$, gives $H(R, t)$, and via (6.10) the required values of V.

6.3.4 Discrete Monitoring

Instead of defining a continuous averaging as in (6.6), a realistic scenario is to assume that the average is monitored only at discrete time instances

$$t_1, t_2, \ldots, t_M .$$

These time instances are not to be confused with the grid times of the numerical discretization. The discretely sampled arithmetic average at t_k is given by

$$A_{t_k} := \frac{1}{k} \sum_{i=1}^{k} S_{t_i}, \; k = 1, \ldots, M . \tag{6.17}$$

A new average is updated from the previous one by

$$A_{t_k} = A_{t_{k-1}} + \frac{1}{k}(S_{t_k} - A_{t_{k-1}})$$

or

$$A_{t_{k-1}} = A_{t_k} + \frac{1}{k-1}(A_{t_k} - S_{t_k}) .$$

The latter of these update formulas is relevant to us, because we integrate backwards in time. The discretely sampled A_t is constant between consecutive sampling times, and A jumps at t_k with the step

$$\frac{1}{k-1}(A_{t_k} - S_{t_k}) .$$

For each k this jump can be written

$$A^-(S) = A^+(S) + \frac{1}{k-1}(A^+(S) - S), \text{ where } S = S_{t_k} . \tag{6.18}$$

A^- and A^+ denote the values of A immediately before and immediately after sampling at t_k. The no-arbitrage principle implies continuity of V at the sampling instances t_k in the sense of continuity of $V(S_t, A_t, t)$ for any realization of a random walk. In our setting, this continuity is written

$$V(S, A^+, t_k) = V(S, A^-, t_k) . \tag{6.19}$$

But for a *fixed* (S, A) the Eqs. (6.18)/(6.19) define a jump of V at t_k.

The numerical application of the jump condition (6.18)/(6.19) is as follows: The A-axis is discretized into discrete values A_j, $j = 1, \ldots, J$. For each time

period between two consecutive sampling instances, say for $t_{k+1} \rightarrow t_k$, the option's value is independent of A because in our discretized setting A_t is piecewise constant; accordingly $\frac{\partial V}{\partial A} = 0$ in (6.8). Based on this semidiscretization, J one-dimensional Black–Scholes equations are integrated separately and independently for the short time interval from t_{k+1} to t_k, one BS-equation for each j. Each of the one-dimensional Black–Scholes problems has its own "terminal" condition to start from. For each A_j, the "first" terminal condition for $t_M = T$ is taken from the payoff surface. Proceeding backwards in time, at each sampling time t_k the J parallel one-dimensional Black–Scholes problems are halted because new terminal conditions must be derived from the jump condition (6.18)/(6.19). The new values for $V(S, A_j, t_k)$ that serve as terminal values (starting values for the backward integration) for the next time period $t_k \rightarrow t_{k-1}$, are defined by the jump condition. Since $A_j + \frac{1}{k-1}(A_j - S)$ in general does not agree with one of the node values A_j, interpolation is applied. Hence the starting function for the next BS-step for $A = A_j$ can be written

$$V^{\text{interpol}}(S,\, A + \frac{1}{k-1}(A - S),\, t_k)\,.$$

Only at these sampling times t_k the J standard one-dimensional Black–Scholes problems are coupled; the coupling is provided by the interpolation. In this way, a sequence of surfaces $V(S, A, t_k)$ is approximated for $t_M = T, \ldots, t_1 = 0$ in a line-wise fashion. Figures 6.3 and 6.4 show[3] the payoff and three surfaces calculated for an Asian European fixed strike put. As this illustration indicates, there is a kind of rotation of this surface as t varies from T to 0.

6.4 Numerical Aspects

A direct numerical approach to the PDE (6.8) for functions $V(S, A, t)$ depending on three independent variables requires more effort than in the $V(S, t)$-case. For example, a finite-difference approach uses a three-dimensional grid. And a separation ansatz as in Sect. 5.3 applies with two-dimensional basis functions. Although much of the required technology is widely analogous to the approaches discussed in Chaps. 4 and 5, a thorough numerical treatment of higher-dimensional PDEs is beyond the scope of this book. Here we confine ourselves to PDEs with two independent variables, as in (6.12).

[3] After interpolation; MATLAB graphics; similar [385].

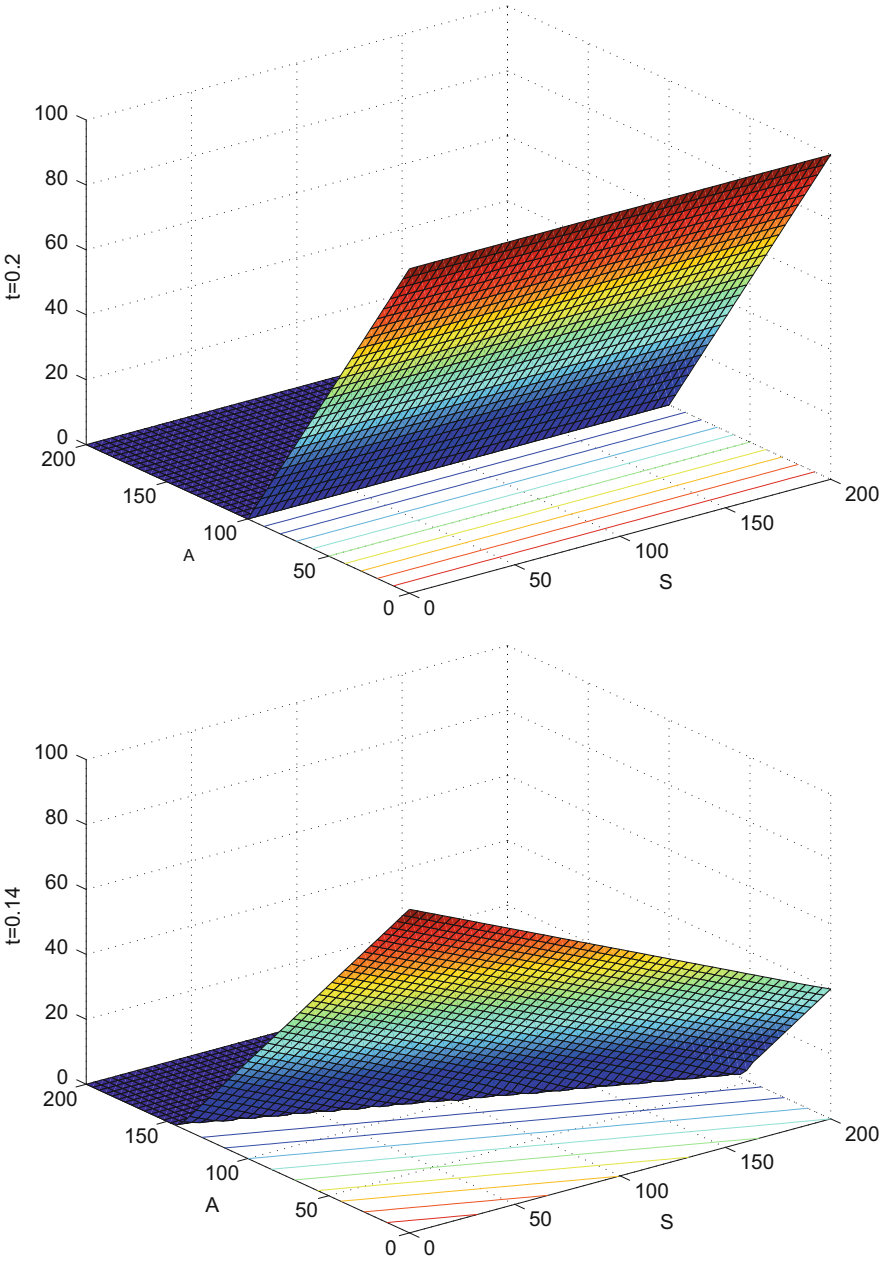

Fig. 6.3 Asian European fixed strike put, $K = 100$, $T = 0.2$, $r = 0.05$, $\sigma = 0.25$. The *top figure* shows the payoff $V(S, A, t)$ for $t = T = 0.2$; the *bottom figure* shows the solution surface $V(S, A, t)$ for $t = 0.14$. For the solution surfaces $t = 0.06$, and $t = 0$ see Fig. 6.4. With kind permission of Sebastian Göbel

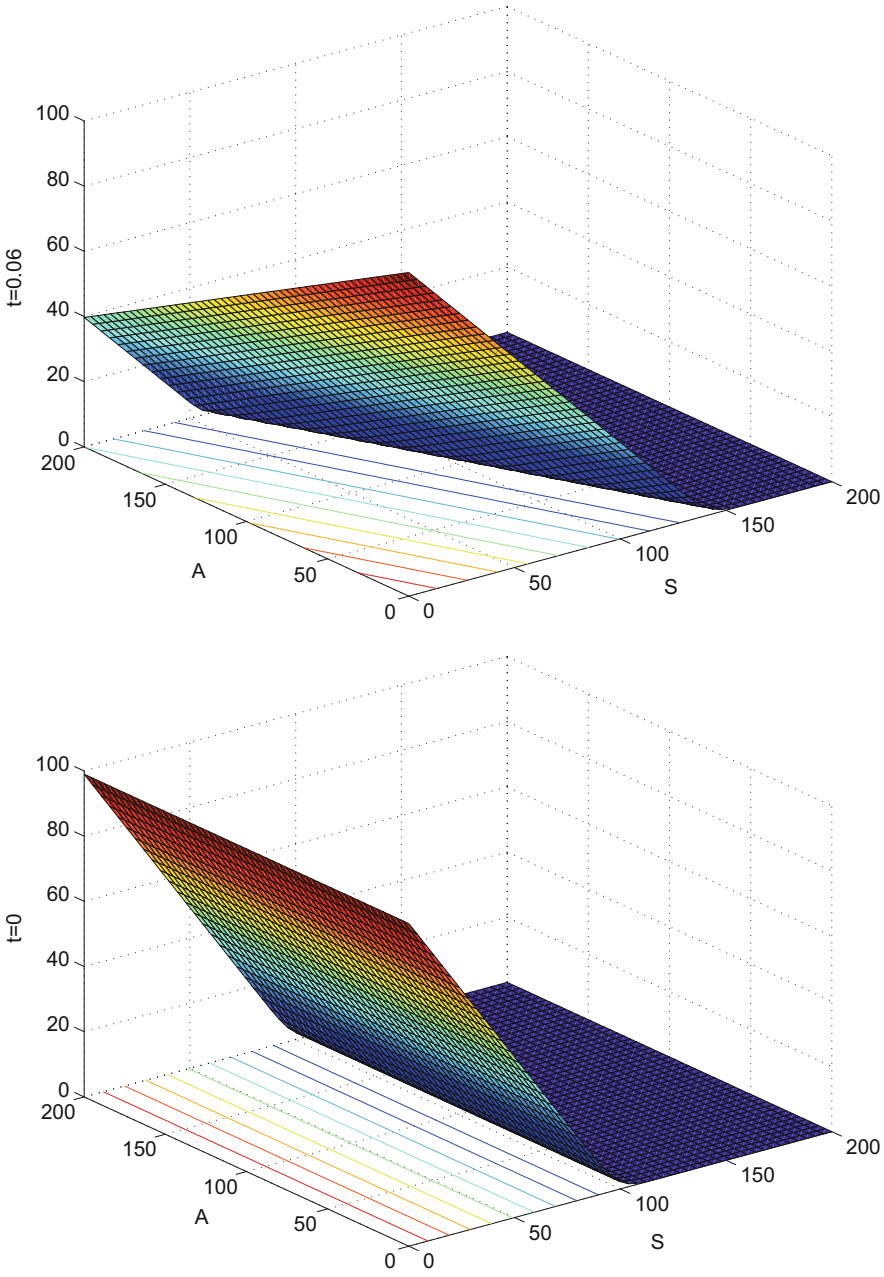

Fig. 6.4 Figure 6.3 continued, with solution surfaces $V(S, A, t)$ for $t = 0.06$ (*top*), and $t = 0$ (*bottom*)

6.4.1 Convection-Diffusion Problems

Before entering a discussion on how to solve a PDE like (6.12) numerically without using transformations like (4.3), we perform an experiment with our familiar Black–Scholes equation (1.5). In contrast to the procedure of Chap. 4 we directly apply finite-difference quotients to (1.5). Here we use the second-order differences of Sect. 4.2.1 for a European call, and compare the numerical approximation with the exact solution (A.13)–(A.17). Figure 6.5 shows the result for $V(S, 0)$. The lower part of the figure depicts an oscillating error, which seems to be small. But differentiating magnifies oscillations. This is clearly visible in Fig. 6.6, where the important hedge variable delta$= \frac{\partial V}{\partial S}$ is depicted. The wiggles are even worse for the second-order derivative gamma. These oscillations are financially unrealistic and are not tolerable, and we have to find its causes. The oscillations are *spurious* in that they are produced by the numerical scheme and are not solutions of the differential equation. The spurious oscillations do not exist for the transformed version $y_\tau = y_{xx}$, which is illustrated by Fig. 6.7.

In order to understand possible reasons why spurious oscillations may occur, we invoke elementary fluid dynamics, where so-called convection-diffusion equations play an important role. For such equations, the second-order term is responsible

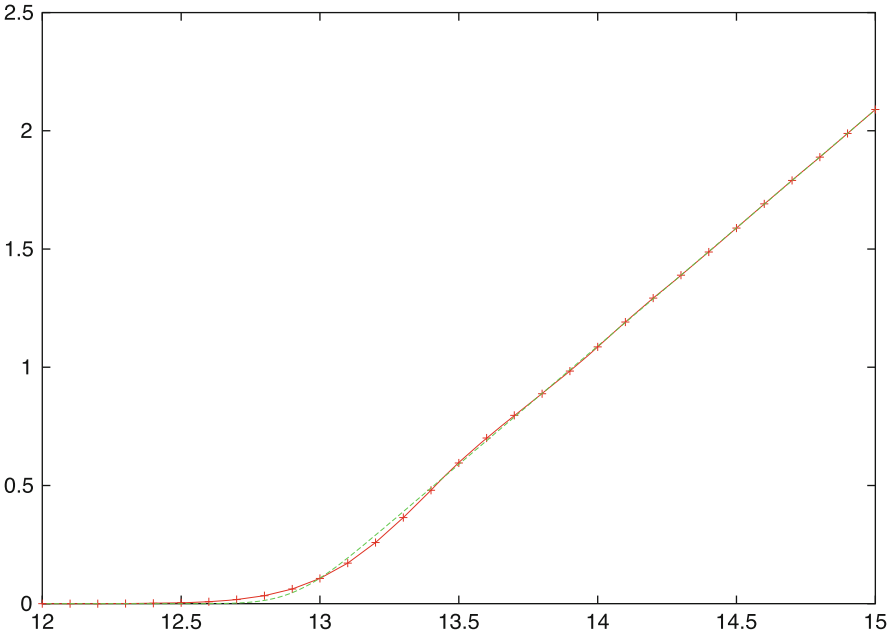

Fig. 6.5 European call, $K = 13$, $r = 0.15$, $\sigma = 0.01$, $T = 1$. Crank–Nicolson approximation $V(S, 0)$ with $\Delta t = 0.01$, $\Delta S = 0.1$ and centered difference scheme for $\frac{\partial V}{\partial S}$ (in *red*). Comparison with the exact Black–Scholes values (*green*)

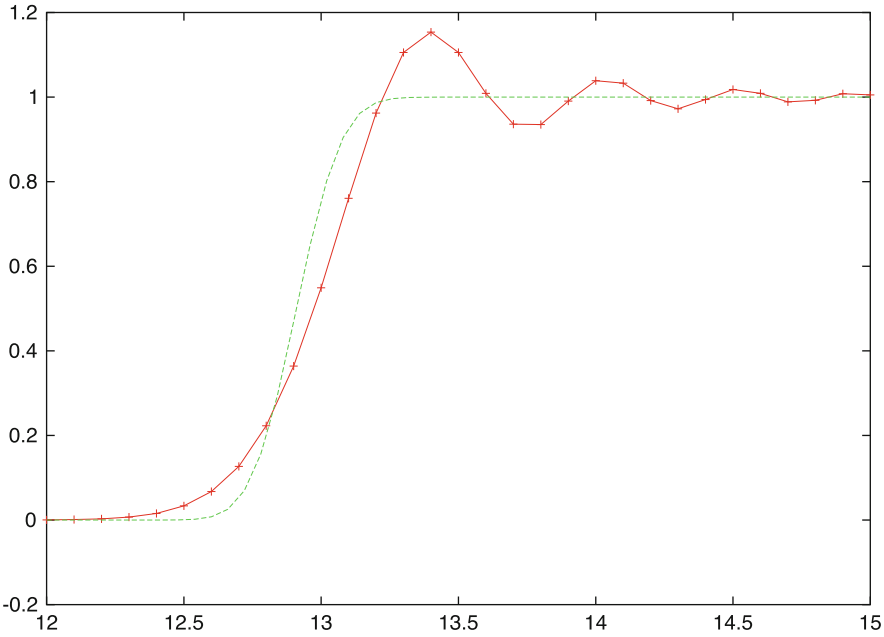

Fig. 6.6 Delta= $\frac{\partial V}{\partial S}$, otherwise the same data as in Fig. 6.5

for diffusion and the first-order term for convection. The ratio of convection to diffusion (their coefficients, scaled by a characteristic length) is the *Péclet number*, a dimensionless parameter characterizing the convection-diffusion problem. It turns out that the Péclet number is relevant for the understanding of underlying phenomena. Let us see what the Péclet numbers are for several PDEs discussed so far in the text.

As a first example we take the original Black–Scholes equation (1.5), with

$$\text{diffusion term:} \quad \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}$$

$$\text{convection term:} \quad rS\frac{\partial V}{\partial S}$$

$$\text{length scale:} \quad \Delta S$$

When the coefficients—not the derivatives—enter the Péclet number, and ΔS is taken as characteristic length, the number is

$$\frac{rS}{\frac{1}{2}\sigma^2 S^2}\Delta S = \frac{2r}{\sigma^2}\ \frac{\Delta S}{S}\,.$$

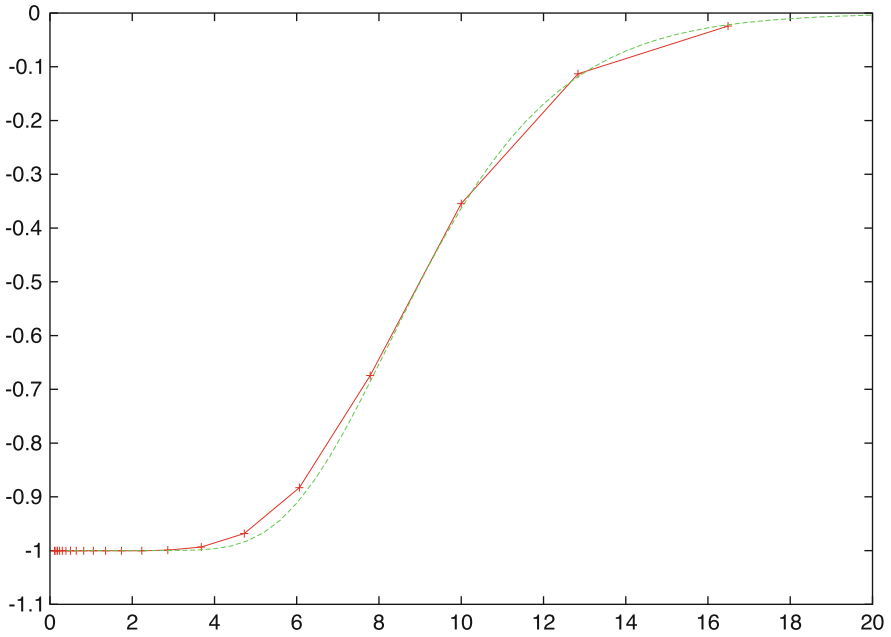

Fig. 6.7 European put, $K = 10$, $r = 0.06$, $\sigma = 0.30$, $T = 1$. Approximation delta$= \frac{\partial V}{\partial S}(S, 0)$ based on $y_\tau = y_{xx}$ with $m = 40$ (in *red*). Comparison with the exact Black–Scholes values (in *green*)

Since this dimensionless parameter involves the mesh size ΔS it is also called *mesh Péclet number*.[4] Experimental evidence indicates that the higher the Péclet number, the higher the danger that the numerical solution exhibits oscillations.

Next we examine other PDEs for their Péclet numbers: The PDE $y_\tau = y_{xx}$ has no convection term, hence its Péclet number is zero. Asian options described by the PDE (6.8) have a cumbersome situation: With respect to A there is no diffusion term (i.e., no second-order derivative), hence its Péclet number is ∞! For the original Black–Scholes equation the Péclet number basically amounts to r/σ^2. It may become large when a small volatility σ is not compensated by a small riskless interest rate r. And for the reduced PDE (6.12), the Péclet number is

$$\frac{\Delta R(1 - rR)}{\frac{1}{2}\sigma^2 R^2},$$

here a small σ can not be compensated by a small r.

[4] In case of a continuous dividend flow δ, replace r by $r - \delta$.

These investigations of the Péclet numbers do not yet explain *why* spurious oscillations occur, but should open our eyes to the relation between convection and diffusion in the different PDEs. Let us discuss causes of the oscillations by means of a **model problem**. The model problem is the pure initial-value problem for a scalar function u defined on $t \geq 0$, $x \in \mathbb{R}$,

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = b\frac{\partial^2 u}{\partial x^2}\,, \quad u(x,0) = u_0(x)\,. \tag{6.20}$$

We assume $b \geq 0$. This sign of b does not contradict the signs in (6.12) since there we have a terminal condition for $t = T$, whereas (6.20) prescribes an initial condition for $t = 0$. The Eq. (6.20) is meant to be integrated in forward time with discretization step size $\Delta t > 0$. So the Eq. (6.20) is a model problem representing a large class of convection-diffusion problems, to which the Eq. (6.12) belongs. For the Black–Scholes equation, the simple transformation $S = K\mathrm{e}^x$, $t = T - \tau$, which works even for variable coefficients r, σ, produces (6.20) except for a further term $-ru$ on the right-hand side (compare Exercise 1.4). And for constant r, σ the transformed equation $y_\tau = y_{xx}$ is a member of the class (6.20), although it lacks convection. Discussing the stability properties of the model problem (6.20) will help us understanding how discretizations of (1.5) or (6.12) behave. For the analysis assume an equidistant grid on the x-range, with grid size $\Delta x > 0$ and nodes $x_j = j\Delta x$ for integers j. And for sake of simplicity, assume a and b are constants.

6.4.2 Von Neumann Stability Analysis

First we apply to (6.20) the standard second-order centered space difference schemes in x-direction together with a forward time step, leading to

$$\frac{w_{j,\nu+1} - w_{j,\nu}}{\Delta t} + a\frac{w_{j+1,\nu} - w_{j-1,\nu}}{2\Delta x} = b\delta_{xx}w_{j,\nu} \tag{6.21}$$

with $\delta_{xx}w_{j,\nu}$ defined as in (4.21). This scheme is called *Forward Time Centered Space* (FTCS). "Forward time" reflects the explicit (forward) Euler step, and "centered space" refers to our well-established second-order difference quotients. Instead of performing an eigenvalue-based stability analysis as in Chap. 4, we now apply the von Neumann stability analysis. This method expresses the approximations $w_{j,\nu}$ of the νth time level by a sum of *eigenmodes* or Fourier modes,

$$w_{j,\nu} = \sum_k c_k^{(\nu)} \mathrm{e}^{\mathrm{i}k\eta j\Delta x}\,, \tag{6.22}$$

where i denotes the imaginary unit, and $k\eta$ are the *wave numbers* with fundamental wave number[5] $\eta := 2\pi/L$. A set of coefficients $c_k^{(\nu)}$ in (6.22) exists for each time level t_ν, it is the basis of the discrete Fourier transform (C.7), which takes numbers w_j into coefficients c_k, and back. Substituting the expression (6.22) into the FTCS-difference scheme (6.21) leads to a corresponding sum for $w_{j,\nu+1}$ with coefficients $c_k^{(\nu+1)}$ ($\longrightarrow$ Exercise 6.5). The linearity of the scheme (6.21) allows to find a relation

$$c_k^{(\nu+1)} = G_k c_k^{(\nu)} ,$$

where G_k is the *growth factor* of the mode with wave number k. In case $|G_k| \le 1$ holds, it is guaranteed that the modes $\mathrm{e}^{\mathrm{i}kx}$ in (6.22) are not amplified, which means the method is stable. This parallels Lemma 4.2 without the need of calculating eigenvalues.

Applying the von Neumann stability analysis to (6.21) leads to

$$G_k = 1 - 2\lambda + \left(\tfrac{\gamma}{2} + \lambda\right) \mathrm{e}^{-\mathrm{i}k\eta\Delta x} + \left(\lambda - \tfrac{\gamma}{2}\right) \mathrm{e}^{\mathrm{i}k\eta\Delta x} ,$$

where we use the abbreviations

$$\gamma := \frac{a\Delta t}{\Delta x} , \quad \lambda := \frac{b\Delta t}{\Delta x^2} , \quad \beta := \frac{a\Delta x}{b} . \tag{6.23}$$

Here $\gamma = \beta\lambda$ is the famous *Courant number*, and β is the *mesh Péclet number*. For a finite value of the latter, assume $b > 0$. Using $\mathrm{e}^{\mathrm{i}\alpha} = \cos\alpha + \mathrm{i}\sin\alpha$ and

$$s := \sin\frac{k\eta\Delta x}{2} , \quad \cos k\eta\Delta x = 1 - 2s^2 , \quad \sin k\eta\Delta x = 2s\sqrt{1-s^2} ,$$

we arrive at

$$G_k = 1 - 2\lambda + 2\lambda\cos k\eta\Delta x - \mathrm{i}\beta\lambda \sin k\eta\Delta x \tag{6.24}$$

and

$$|G_k|^2 = (1 - 4\lambda s^2)^2 + 4\beta^2\lambda^2 s^2(1 - s^2) .$$

The last expression for $|G_k|^2$ is a polynomial on $0 \le s^2 \le 1$. A straightforward discussion of this polynomial reveals that $|G_k| \le 1$ for

$$0 \le \lambda \le \tfrac{1}{2} , \quad \lambda\beta^2 \le 2 . \tag{6.25}$$

[5] L stands for the wave length or the length of the interval. In case of a partition into n steps of size Δx, $\eta\Delta x = 2\pi/n$. Without loss of generality, we may set $L = 2\pi$, so $\eta = 1$ for the following analysis. It will be sufficient to study the propagation of $\mathrm{e}^{\mathrm{i}kx}$.

The inequality $0 \leq \lambda \leq \frac{1}{2}$ brings back the stability criterion of Sect. 4.2.4. And the inequality $\lambda\beta^2 \leq 2$ is an additional restriction to the parameters λ and β. Because of

$$\lambda\beta^2 = \frac{a^2 \Delta t}{b}$$

this restriction depends on the discretization steps Δt, Δx, and on the convection parameter a and the diffusion parameter b as defined in (6.23). The restriction due to the convection becomes apparent when we, for example, choose $\lambda = \frac{1}{2}$ for a maximal time step Δt. Then $|\beta| \leq 2$ is a bound imposed on the mesh Péclet number, which restricts Δx to $\Delta x \leq 2b/|a|$. A violation of this bound might be an explanation why the difference schemes of (6.21) applied to the Black–Scholes equation (1.5) exhibit faulty oscillations.[6] The bounds on $|\beta|$ and Δx are not active for problems without convection ($a = 0$). Note that the bounds give a severe restriction on problems with small values of the diffusion constant b. For $b \to 0$ (no diffusion) and $a \neq 0$ we encounter the consequence $\Delta t \to 0$, and the scheme (6.21) can not be applied at all. Although the constant-coefficient model problem (6.20) is not the same as the Black–Scholes equation (1.5) or the Eq. (6.12), the above analysis reflects the core of the difficulties. We emphasize that small values of the volatility represent small diffusion. So other methods than the standard finite-difference approach (6.21) are needed.

6.5 Upwind Schemes and Other Methods

The instability analyzed for the model combination (6.20)/(6.21) occurs when the mesh Péclet number is high and because the symmetric and centered difference quotient is applied to the first-order derivative. Next we discuss the extreme case of an infinite Péclet number of the model problem, namely, $b = 0$. The resulting PDE is the prototypical equation

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0\,. \tag{6.26}$$

6.5.1 Upwind Scheme

The standard FTCS approach for (6.26) does not lead to a stable scheme. The PDE (6.26) has solutions in the form of *traveling waves*,

$$u(x,t) = W(x - at)\,,$$

[6] In fact, the situation is more subtle. We postpone an outline of how *dispersion* is responsible for the oscillations to Sect. 6.5.2.

where $W(\xi) = u_0(\xi)$ in case initial conditions $u(x,0) = u_0(x)$ are incorporated. For $a > 0$, the profile $W(\xi)$ drifts in positive x-direction: the "wind blows to the right." Seen from a grid point (j,ν), the neighboring node $(j-1,\nu)$ lies *upwind* and $(j+1,\nu)$ lies *downwind*. Here the j indicates the node x_j and ν the time instant t_ν. Information flows from upstream to downstream nodes. Accordingly, the first-order difference scheme

$$\frac{w_{j,\nu+1} - w_{j,\nu}}{\Delta t} + a\frac{w_{j,\nu} - w_{j-1,\nu}}{\Delta x} = 0 \tag{6.27}$$

is called *upwind discretization* ($a > 0$). The scheme (6.27) is also called Forward Time Backward Space (FTBS) scheme.

Applying the von Neumann stability analysis to the scheme (6.27) leads to growth factors given by

$$G_k := 1 - \gamma + \gamma \mathrm{e}^{-\mathrm{i}k\eta\Delta x}\,. \tag{6.28}$$

Here $\gamma = \frac{a\Delta t}{\Delta x}$ is the Courant number from (6.23). As in Sect. 6.4.2, the stability requirement $|G_k| \le 1$ should hold such that the coefficients $c_k^{(\nu)}$ remain bounded for all k and $\nu \to \infty$. It is easy to see that

$$\gamma \le 1 \;\Rightarrow\; |G_k| \le 1\,.$$

(The reader may sketch the complex G-plane to realize the situation.) The condition $|\gamma| \le 1$ is called the **Courant–Friedrichs–Lewy (CFL) condition**. The above analysis shows that this condition is sufficient to ensure stability of the upwind-scheme (6.27) applied to the PDE (6.26) with prescribed initial conditions.

In case $a < 0$, the scheme in (6.27) is no longer an upwind scheme. The upwind scheme for $a < 0$ is

$$\frac{w_{j,\nu+1} - w_{j,\nu}}{\Delta t} + a\frac{w_{j+1,\nu} - w_{j,\nu}}{\Delta x} = 0\,. \tag{6.29}$$

The von Neumann stability analysis leads to the restriction $|\gamma| \le 1$, or $\lambda|\beta| \le 1$ if expressed in terms of the mesh Péclet number, see (6.23). This again emphasizes the importance of small Péclet numbers.

We note in passing that the FTCS scheme for $u_t + au_x = 0$, which is unstable, can be cured by replacing $w_{j,\nu}$ by the average of its two neighbors. The resulting scheme

$$w_{j,\nu+1} = \tfrac{1}{2}(w_{j+1,\nu} + w_{j-1,\nu}) - \tfrac{1}{2}\gamma(w_{j+1,\nu} - w_{j-1,\nu}) \tag{6.30}$$

is called *Lax–Friedrichs scheme*. It is stable if and only if the CFL condition is satisfied. A simple calculation shows that the Lax–Friedrichs scheme (6.30)

can be rewritten in the form

$$\frac{w_{j,\nu+1} - w_{j,\nu}}{\Delta t} = -a\frac{w_{j+1,\nu} - w_{j-1,\nu}}{2\Delta x} + \frac{1}{2\Delta t}\left(w_{j+1,\nu} - 2w_{j,\nu} + w_{j-1,\nu}\right). \tag{6.31}$$

This is a FTCS scheme with the additional term

$$\frac{(\Delta x)^2}{2\Delta t}\,\delta_{xx} w_{j,\nu}\,,$$

representing the PDE

$$u_t + a u_x = \zeta u_{xx} \text{ with } \zeta = \Delta x^2/2\Delta t\,.$$

That is, the stabilization is accomplished by adding artificial diffusion ζu_{xx}. The scheme (6.31) is said to have *numerical dissipation*.

We return to the model problem (6.20) with $b > 0$. For the discretization of the $a\frac{\partial u}{\partial x}$ term we now apply the appropriate upwind scheme from (6.27) or (6.29), depending on the sign of the convection constant a. This noncentered first-order difference scheme can be written

$$\begin{aligned} w_{j,\nu+1} = w_{j,\nu} &- \gamma\tfrac{1-\mathrm{sign}(a)}{2}\left(w_{j+1,\nu} - w_{j,\nu}\right) \\ &- \gamma\tfrac{1+\mathrm{sign}(a)}{2}\left(w_{j,\nu} - w_{j-1,\nu}\right) \\ &+ \lambda(w_{j+1,\nu} - 2w_{j,\nu} + w_{j-1,\nu}) \end{aligned} \tag{6.32}$$

with parameters γ, λ as defined in (6.23). For $a > 0$ the growth factors are

$$G_k = 1 - \lambda(2+\beta)(1 - \cos k\eta\Delta x) - \mathrm{i}\lambda\beta \sin k\eta\Delta x\,.$$

The analysis follows the lines of Sect. 6.4 and leads to the single stability criterion

$$\lambda \le \frac{1}{2 + |\beta|}\,. \tag{6.33}$$

This inequality is valid for both signs of a ($\longrightarrow$ Exercise 6.6). For $\lambda \ll \beta$ the inequality (6.33) is less restrictive than (6.25). For example, a hypothetical value of $\lambda = \frac{1}{50}$ leads to the bound $|\beta| \le 10$ for the FTCS scheme (6.21) and to the bound $|\beta| \le 48$ for the upwind scheme (6.32).

The Figs. 6.8 and 6.9 show the Black–Scholes solution and an approximation obtained by using the upwind scheme as in (6.32). No oscillations are visible. But the low order of the approximation can be seen from the moderate gradient, which does not reflect the steep gradient of the reality. The spurious wiggles have disappeared but the steep profile is heavily smeared. So the upwind scheme discussed above is a motivation to look for better methods (in Sect. 6.6).

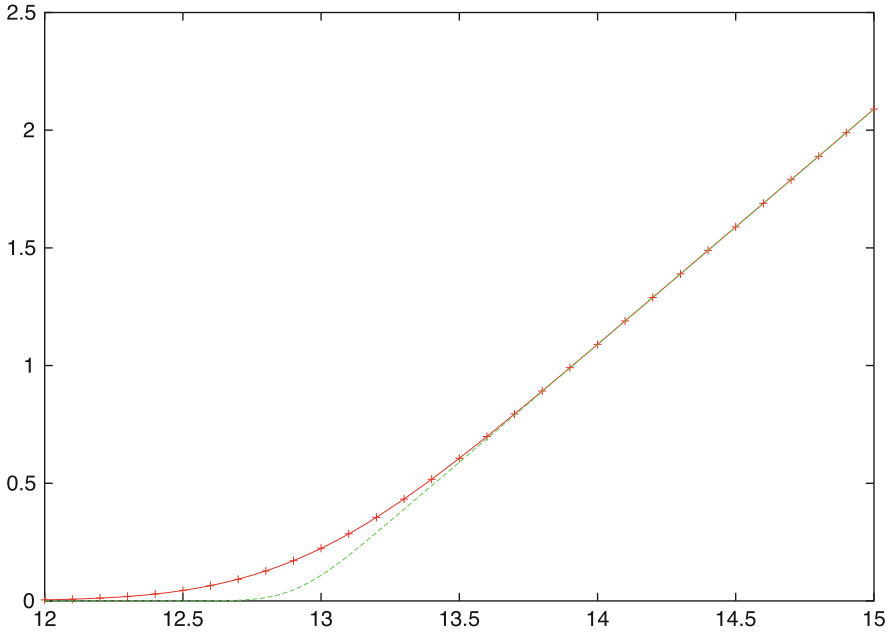

Fig. 6.8 European call, $K = 13$, $r = 0.15$, $\sigma = 0.01$, $T = 1$. Approximation $V(S, 0)$ (in *red*), calculated with upwind scheme for $\frac{\partial V}{\partial S}$ and $\Delta t = 0.01$, $\Delta S = 0.1$. Comparison with the exact Black–Scholes values (*green*)

6.5.2 Dispersion

The spurious wiggles are attributed to *dispersion*. Dispersion is not due to rounding errors. Rather, dispersion is the phenomenon of different modes traveling at different speeds. We explain dispersion for the simple PDE $u_t + au_x = 0$. Consider for $t = 0$ an initial profile u represented by a sum of Fourier modes, as in (6.22). Because of the linearity of the PDE it is sufficient to study how the kth mode e^{ikx} is conveyed for $t > 0$. The differential equation $u_t + au_x = 0$ conveys the mode without deformation, because $e^{ik[x-at]}$ is a solution. For an observer who travels with speed a along the x-axis, the mode appears "frozen."

This does not hold for the numerical scheme. Here the amplitude and the phase of the kth mode may change. That is, the special initial profile of the Fourier mode

$$e^{ikx} = 1 \cdot e^{ik[x-0]}$$

deforms to

$$c(t) \cdot e^{ik[x-d(t)]},$$

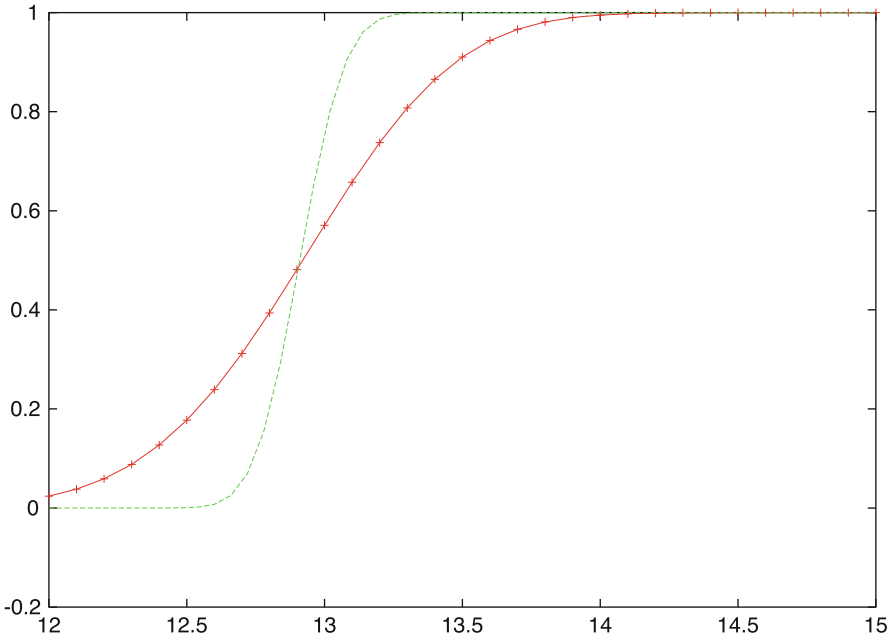

Fig. 6.9 Delta= $\frac{\partial V}{\partial S}(S,0)$, same data as in Fig. 6.8

where $c(t)$ is the amplitude and $d(t)$ the phase (up to the traveler distance at). Their values must be compared to those of the exact solution.

To be specific, we study the upwind scheme for $u_t + au_x = 0$ $(a > 0)$,

$$\frac{w(x,t+\Delta t) - w(x,t)}{\Delta t} + a\frac{w(x,t) - w(x-\Delta x,t)}{\Delta x} = 0\,.$$

Let $w(x,t)$ denote the exact solution of this difference equation for specified values of Δx, Δt. Apply Taylor's expansion to derive the *equivalent differential equation*

$$w_t + aw_x = \zeta w_{xx} + \xi w_{xxx} + O(\Delta^2)\,,$$

with the coefficients

$$\begin{aligned}\zeta &:= \frac{a}{2}(\Delta x - a\Delta t) &&= \frac{a}{2}\Delta x(1-\gamma)\,,\\ \xi &:= \frac{a}{6}(-\Delta x^2 + 3a\Delta t\Delta x - 2a^2\Delta t^2) &&= \frac{a}{6}\Delta x^2(1-\gamma)(2\gamma-1)\end{aligned}$$

depending on $\Delta x, \Delta t$. (Recall $\gamma = a\Delta t/\Delta x$.) A solution can be obtained for the truncated PDE $w_t + aw_x = \zeta w_{xx} + \xi w_{xxx}$. Substituting $w = \mathrm{e}^{\mathrm{i}(\omega t + kx)}$ with

undetermined frequency ω gives $\mathrm{i}\omega = -\zeta k^2 - \mathrm{i}k(a - \xi k^2)$ and thus

$$w = \exp\{-\zeta k^2 t\} \cdot \exp\{\mathrm{i}k[x - t(\xi k^2 + a)]\}$$

as solution of the truncated PDE. This defines amplitudes $c(t)$ and phase shifts $d(t)$,

$$\begin{aligned} c_k(t) &= \exp\{-\zeta k^2 t\}\,, \\ d_k(t) &= \xi k^2 t\,. \end{aligned}$$

The $w = c_k(t)\mathrm{e}^{\mathrm{i}k[x-at-d_k(t)]}$ represents the solution of the upwind scheme. It is compared to the exact solution $u = \mathrm{e}^{\mathrm{i}k[x-at]}$ of the model problem, for which all modes propagate with the same speed a and without decay of the amplitude. The phase shift d_k in w due to a nonzero ξ becomes more relevant if the wave number k gets larger. That is, modes with different wave numbers drift across the finite-difference grid at different rates. Consequently, an initial signal represented by a sum of modes, changes its shape as it travels. The different propagation speeds of different modes $\mathrm{e}^{\mathrm{i}kx}$ give rise to oscillations. This phenomenon is called dispersion. (Note that in our scenario of the simple model problem with upwind scheme, for $\gamma = 1$ and $\gamma = \frac{1}{2}$ we have $\xi = 0$ and dispersion vanishes.)

A value of $|c(t)| < 1$ amounts to dissipation. If a high phase shift is compensated by heavy dissipation ($c \approx 0$), then the dispersion is damped and may be hardly noticeable. Generally, for reasonable damping require ζ to be large compared to ξ.

For several numerical schemes, related values of ζ and ξ have been investigated. For the influence of dispersion or dissipation see, for example, [308, 350, 353, 356]. Dispersion is to be expected for numerical schemes that operate on those versions of the Black–Scholes equation that have a convection term. This holds in particular for the θ-methods as described in Sect. 4.6.1, and for the upwind scheme. In contrast, numerical schemes for the convection-free version $y_\tau = y_{xx}$ do not suffer from dispersion since $a = 0$.

6.6 High-Resolution Methods

The naive FTCS approach of the scheme (6.21) is only first-order in t-direction and suffers from severe stability restrictions. There are second-order approaches with better properties. A large class of schemes has been developed for so-called *conservation laws*, which in the one-dimensional situation are written

$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x} f(u) = 0\,. \tag{6.34}$$

The function $f(u)$ represents the *flux* in the Eq. (6.34), which originally was tailored to applications in fluid dynamics. We introduce the method of Lax and Wendroff for the flux-conservative equation (6.34). Then we present basic ideas of high-resolution methods.

6.6.1 Lax–Wendroff Method

The Lax–Wendroff scheme is based on

$$u_{j,\nu+1} = u_{j,\nu} + \Delta t \frac{\partial u_{j,\nu}}{\partial t} + O(\Delta t^2) = u_{j,\nu} - \Delta t \frac{\partial f(u_{j,\nu})}{\partial x} + O(\Delta t^2)\,.$$

This expression makes use of (6.34) and replaces time derivatives by space derivatives. For suitably adapted indices the basic scheme is applied three times on a *staggered grid*. The staggered grid (see Fig. 6.10) uses half steps of lengths $\frac{1}{2}\Delta x$ and $\frac{1}{2}\Delta t$ and intermediate node numbers $j - \frac{1}{2}, j + \frac{1}{2}, \nu + \frac{1}{2}$. The main step is the second-order centered step (CTCS) with the center in the node $(j, \nu + \frac{1}{2})$ (square in Fig. 6.10). This main step needs the flux function f evaluated at approximations w obtained for the two intermediate nodes $\left(j \pm \frac{1}{2}, \nu + \frac{1}{2}\right)$, which are marked by crosses in Fig. 6.10. These two intermediate values are provided by the Lax–Friedrichs steps (6.30).

Algorithm 6.2 (Lax–Wendroff)

$$\begin{aligned} w_{j+\frac{1}{2},\nu+\frac{1}{2}} &:= \tfrac{1}{2}(w_{j,\nu} + w_{j+1,\nu}) - \tfrac{\Delta t}{2\Delta x}\left(f(w_{j+1,\nu}) - f(w_{j,\nu})\right) \\ w_{j-\frac{1}{2},\nu+\frac{1}{2}} &:= \tfrac{1}{2}(w_{j-1,\nu} + w_{j,\nu}) - \tfrac{\Delta t}{2\Delta x}\left(f(w_{j,\nu}) - f(w_{j-1,\nu})\right) \\ w_{j,\nu+1} &:= w_{j,\nu} - \tfrac{\Delta t}{\Delta x}\left(f(w_{j+\frac{1}{2},\nu+\frac{1}{2}}) - f(w_{j-\frac{1}{2},\nu+\frac{1}{2}})\right) \end{aligned} \tag{6.35}$$

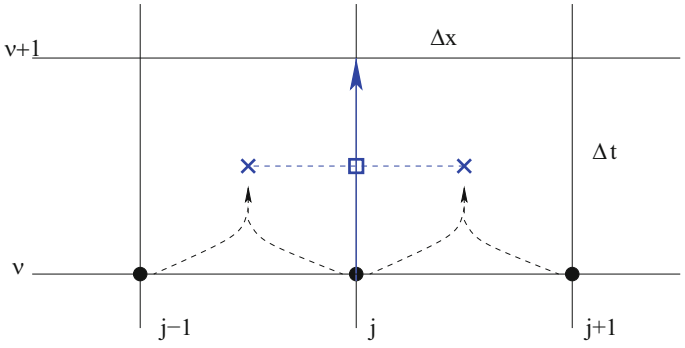

Fig. 6.10 (x, t)-plane: staggered grid for the Lax–Wendroff scheme

The half-step values $w_{j+\frac{1}{2},\nu+\frac{1}{2}}$ and $w_{j-\frac{1}{2},\nu+\frac{1}{2}}$ are provisional and discarded after $w_{j,\nu+1}$ is calculated. A stability analysis for the special case $f(u) = au$ in Eq. (6.34) [that is, of Eq. (6.26)] leads to the CFL condition as before. The Lax–Wendroff step is centered and of second order in both x and t. This explicit method fits well discontinuities and steep fronts as the Black–Scholes delta-profile in Figs. 6.6 and 6.9. But there are still spurious wiggles in the vicinity of steep gradients. The Lax–Wendroff scheme produces oscillations near sharp fronts. We need to find a way to damp out the oscillations.

6.6.2 *Total Variation Diminishing*

Since $u_t + au_x$ convects an initial profile $W(x)$ with velocity a, a monotonicity of W will be preserved for all $t > 0$. So it makes sense to require also a numerical scheme to be *monotonicity preserving*. That is,

$$w_{j,0} \le w_{j+1,0} \text{ for all } j \;\Rightarrow\; w_{j,\nu} \le w_{j+1,\nu} \text{ for all } j, \nu \ge 1\,,$$
$$w_{j,0} \ge w_{j+1,0} \text{ for all } j \;\Rightarrow\; w_{j,\nu} \ge w_{j+1,\nu} \text{ for all } j, \nu \ge 1\,.$$

A stronger requirement is that oscillations be diminished. To this end we define the *total variation* of the approximation vector $w^{(\nu)}$ at the νth time level as

$$\mathrm{TV}(w^{(\nu)}) := \sum_j |w_{j+1,\nu} - w_{j,\nu}|\,. \tag{6.36}$$

The aim is to construct a method that is *total variation diminishing* (TVD),

$$\mathrm{TV}(w^{(\nu+1)}) \le \mathrm{TV}(w^{(\nu)}) \text{ for all } \nu\,.$$

Before we come to a criterion for TVD, note that the schemes discussed in this section are explicit and of the form

$$w_{j,\nu+1} = \sum_l d_l\, w_{j+l,\nu}\,. \tag{6.37}$$

For example, the upwind scheme (6.27) for $a > 0$

$$w_{j,\nu+1} = (1-\gamma)w_{j,\nu} + \gamma w_{j-1,\nu}$$

has two coefficients in (6.37), $d_{-1} = \gamma$ and $d_0 = 1 - \gamma$. The coefficients d_l decide whether the scheme (6.37) is monotonicity preserving or TVD.

Lemma 6.3 (Monotonicity and TVD)

(a) The scheme (6.37) is monotonicity preserving if and only if $d_l \geq 0$ *for all* d_l.
(b) The scheme (6.37) is total variation diminishing (TVD) if and only if

$$d_l \geq 0 \text{ for all } d_l\,, \text{ and } \sum_l d_l \leq 1\,.$$

The proof of (a) is left to the reader; for proving (b) the reader may find help in [374], see also [232]. As a consequence of Lemma 6.3 note that TVD implies monotonicity preservation.

The criterion of Lemma 6.3 is straightforward to check. For example, we now can be certain about the upwind scheme's monotonicity preservation shown in Figs. 6.8 and 6.9. The Lax–Wendroff scheme satisfies $d_l \geq 0$ for all l only in the exceptional case $\gamma = 1$. For practical purposes, in view of nonconstant coefficients a, the Lax–Wendroff scheme is not TVD. For $f(u) = au$, the upwind scheme (6.27) and the Lax–Friedrichs scheme (6.30) are TVD for $|\gamma| \leq 1$ ($\longrightarrow$ Exercise 6.7).

6.6.3 Numerical Dissipation

For clarity we continue to discuss the matters for the linear scalar equation (6.26),

$$u_t + au_x = 0\,, \text{ for } a > 0\,.$$

For this equation it is easy to substitute the two provisional half-step values of the Lax–Wendroff algorithm into the equation for $w_{j,\nu+1}$. Then a straightforward calculation shows that the Lax–Wendroff scheme can be obtained by adding a diffusion term to the upwind scheme (6.27). To show this, make use of the difference operator

$$\delta_x^- w_{j,\nu} := w_{j,\nu} - w_{j-1,\nu} \tag{6.38}$$

and rewrite the upwind scheme as

$$w_{j,\nu+1} = w_{j,\nu} - \gamma \delta_x^- w_{j,\nu}\,, \quad \gamma = \frac{a\,\Delta t}{\Delta x}\,.$$

The reader may check that the Lax–Wendroff scheme is obtained by adding the term

$$-\delta_x^- \{\tfrac{1}{2}\gamma(1-\gamma)(w_{j+1,\nu} - w_{j,\nu})\} \tag{6.39}$$

to the upwind scheme. So the Lax–Wendroff scheme is rewritten

$$w_{j,\nu+1} = w_{j,\nu} - \gamma\delta_x^- w_{j,\nu} - \delta_x^-\{\tfrac{1}{2}\gamma(1-\gamma)(w_{j+1,\nu} - w_{j,\nu})\}\,.$$

That is, the Lax–Wendroff scheme is the first-order upwind scheme plus the term (6.39), which is

$$-\tfrac{1}{2}\gamma(1-\gamma)(w_{j+1,\nu} - 2w_{j,\nu} + w_{j-1,\nu})\,.$$

Hence the added term is—similar as for the Lax–Friedrichs scheme (6.31)—the discretized analogue of the artificial diffusion

$$-\tfrac{1}{2}a\Delta t(\Delta x - a\Delta t)u_{xx}\,.$$

Adding this artificial dissipation term (6.39) to the upwind scheme makes the scheme a second-order method.

The aim is to find a scheme that will give us neither the wiggles of the Lax–Wendroff scheme nor the smearing and low accuracy of the upwind scheme. On the other hand, we wish to benefit both from the second-order accuracy of the Lax–Wendroff scheme and from the smoothing capabilities of the upwind scheme. A core idea is not to add the same amount of dissipation everywhere along the x-axis, but to add artificial dissipation in the right amount where it is needed. This flexibility is achieved by a proper factor on the diffusion (6.39). The resulting hybrid scheme will be of Lax–Wendroff type when the gradient is flat, and will be upwind-like at strong gradients of the solution. The decision on how much dissipation to add will be based on the solution.

In order to meet the goals, high-resolution methods control the artificial dissipation by introducing a *limiter* $\ell_{j,\nu}$ such that

$$w_{j,\nu+1} = w_{j,\nu} - \gamma\delta_x^- w_{j,\nu} - \delta_x^-\{\,\ell_{j,\nu}\,\tfrac{1}{2}\gamma(1-\gamma)(w_{j+1,\nu} - w_{j,\nu})\}\,. \tag{6.40}$$

Obviously this hybrid scheme specializes to the upwind scheme for $\ell_{j,\nu} = 0$ and is identical to the Lax–Wendroff scheme for $\ell_{j,\nu} = 1$. Accordingly, $\ell_{j,\nu} = 0$ should be chosen for strong gradients in the solution profile and $\ell_{j,\nu} = 1$ for smooth sections. To check the smoothness of the solution one defines the *smoothness parameter*

$$q_{j,\nu} := \frac{w_{j,\nu} - w_{j-1,\nu}}{w_{j+1,\nu} - w_{j,\nu}}\,. \tag{6.41}$$

The limiter $\ell_{j,\nu}$ will be a function of $q_{j,\nu}$. We now drop the indices j, ν. For $q \approx 1$ the solution will be considered smooth, so we require the function $\ell = \ell(q)$ to satisfy $\ell(1) = 1$ to reproduce the Lax–Wendroff scheme. Several strategies have been suggested to choose the limiter function $\ell(q)$ such that the scheme (6.40) is total variation diminishing. For a thorough discussion of this matter we refer to [232, 352, 357]. One example of a limiter function is the van Leer limiter, which is

defined by

$$\ell(q) = \begin{cases} 0 & , \ q \le 0\,, \\ \frac{2q}{1+q} & , \ q > 0\,. \end{cases} \tag{6.42}$$

The above principles of high-resolution methods have been applied successfully to financial engineering. The transfer of ideas from the simple model problem (6.26) to the Black–Scholes world is quite involved. The methods are TVD for the Black–Scholes equation, which is in nonconservative form. Further, the methods can be applied to nonuniform grids, and to implicit methods. The application of the Crank–Nicolson approach can be recommended. Equations (6.41), (6.42) introduce a nonlinearity in $w^{(\nu+1)}$. Hence nonlinear equations are solved for each time step ν; Newton's method is applied to calculate the approximation $w^{(\nu+1)}$ [383].

6.7 Penalty Method for American Options

As we have seen in Chap. 4, the PDE description of an American-style option leads to a linear complementarity problem (LCP), which was restated in Problem 4.12 as an equation under an inequality as side condition. Such problems can be solved numerically by imposing a penalty in case the inequality is violated. For motivation see Sect. 4.5.4, and study the simple setting of Exercise 6.8. Penalty methods have been applied repeatedly for the pricing of American options, see for instance [133, 230, 288]. Here we describe the approach of [289].

6.7.1 LCP Formulation

Similar as in Sect. 4.5.3 we denote the n-dimensional Black–Scholes operator of (6.5)

$$\mathcal{L}_{\mathrm{BS}}(V) := \frac{1}{2}\sum_{i,j=1}^{n} \rho_{ij}\,\sigma_i\sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} + \sum_{i=1}^{n}(r-\delta_i)S_i\frac{\partial V}{\partial S_i} - rV \tag{6.43}$$

and the payoff by $\Psi(S_1,\ldots,S_n)$. For example, for a basket put,

$$\Psi(S_1,\ldots,S_n) = \left(K - \sum_{i=1}^{n} c_i S_i\right)^+ ,$$

with non-negative weights $c_1, \ldots, c_n$. The LCP is

$$\begin{aligned} &(V - \Psi)\left(\frac{\partial V}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V)\right) = 0\,, \\ &\frac{\partial V}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V) \le 0\,, \\ &V \ge \Psi\,. \end{aligned} \tag{6.44}$$

In addition, with vector notation $S := (S_1, \ldots, S_n)$, the terminal condition $V(S, T) = \Psi(S)$ must hold, and boundary conditions. Since the domain is $S_i > 0$ for $i = 1, \ldots, n$, there are n bounding planes given by $S_i = 0$. For each i let

$$\mathcal{D}_i := \{\, (S_1, \ldots, S_{i-1}, 0, S_{i+1}, \ldots, S_n) \mid S_j > 0 \text{ for } j \neq i \,\}$$

denote the domain of the associated $(n-1)$-dimensional American option problem with the same terms, and $G_i(S, t)$ for $S \in \mathcal{D}_i$ be its solution. Then the boundary conditions for the bounding planes $S_i = 0$ are defined by

$$V(S, t) = G_i(S, t) \quad \text{for } S \in \mathcal{D}_i \tag{6.45}$$

for all $i = 1, \ldots, n$. Note that these boundary conditions amount to the recursive solution of all lower-dimensional American option problems. This is an enormous amount of work for larger n, and limits the approach to small values of the dimension. The final item to be specified are the boundary conditions for $S_i \to \infty$. For the case of a put,

$$\lim_{S_i \to \infty} V(S, t) = 0 \quad \text{for all } i\,.$$

The above equations define the LCP for an n-asset American option under the Black–Scholes model.

6.7.2 Penalty Formulation

In the following, we stay with the American put with a basket payoff. For a penalty approach, replace the LCP formulation (6.44) by

$$\begin{aligned} &\frac{\partial V^{\epsilon,C}}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V^{\epsilon,C}) + \frac{\epsilon\, C}{V^{\epsilon,C} + \epsilon - q} = 0 \\ &\text{with } q := K - \sum_{i=1}^{n} c_i S_i\,. \end{aligned} \tag{6.46}$$

q is the basic part of the basket's payoff. We call the solution of the penalty formulation (6.46) $V^{\epsilon,C}$; it is supposed to approximate V. Clearly, the value function V and its approximation $V^{\epsilon,C}$ should both satisfy $V \geq q$. The parameter ϵ in the penalty term

$$p := \frac{\epsilon\, C}{V^{\epsilon,C} + \epsilon - q} \tag{6.47}$$

must be chosen small with $0 < \epsilon \ll 1$. The parameter $C > 0$ is a tune factor to be fixed later. For $V^{\epsilon,C} \gg q$, the penalty term is of the order ϵ, and (6.46) approximates the Black–Scholes equation. As $V^{\epsilon,C}$ approaches the payoff, $V^{\epsilon,C} \approx q$, the penalty term p approaches the value $C > 0$, and

$$\frac{\partial V^{\epsilon,C}}{\partial t} + \mathcal{L}_{\mathrm{BS}}(V^{\epsilon,C}) \approx -C < 0\,.$$

This reflects the complementarity of American options. Note that the Eq. (6.46) is nonlinear in V.[7]

6.7.3 *Discretization of the Two-Factor Model*

For the discretization of the American-style basket put we restrict ourselves to the case $n = 2$. Then the lower-dimensional American put problems are the plain-vanilla cases discussed in Chap. 4, and the corresponding standard value functions $G_1(S_2, t)$ for $S_1 = 0$ and $G_2(S_1, t)$ for $S_2 = 0$ can be considered "known" or delegated to a subalgorithm. The functions G_1 and G_2 are defined by the Black–Scholes equation/inequality, and by their payoff and volatility:

$$G_1(S_2, t) \text{ with payoff } (K - c_2 S_2)^+, \text{ volatility } \sigma_2\,,$$
$$G_2(S_1, t) \text{ with payoff } (K - c_1 S_1)^+, \text{ volatility } \sigma_1\,.$$

Here we apply a standard finite-difference scheme, widely analogous as in Chap. 4. The nonlinearity of the PDE (6.46) prevents a transformation such as (4.3). Hence the discretization is applied to (6.46) directly. For ease of notation, we use the variables

$$x := S_1,\ y := S_2\,,$$

[7] Actually, the LCP (6.44) is nonlinear as well, which is not correctly reflected by the name "LCP".

and ρ for ρ_{12}. Then the penalty problem (6.46) for $V^{\epsilon,C}(x,y,t)$ is restated as (the superscript ϵ, C of $V^{\epsilon,C}$ is dropped)

$$\begin{aligned}&\frac{\partial V}{\partial t}+\frac{1}{2}\sigma_1^2x^2\frac{\partial^2V}{\partial x^2}+\frac{1}{2}\sigma_2^2y^2\frac{\partial^2V}{\partial y^2}+\rho\sigma_1\sigma_2xy\frac{\partial^2V}{\partial x\partial y}\\&+(r-\delta_1)x\frac{\partial V}{\partial x}+(r-\delta_2)y\frac{\partial V}{\partial y}-rV+\frac{\epsilon\, C}{V+\epsilon-q}=0\end{aligned}\tag{6.48}$$

with terminal and boundary conditions. For a put with basket payoff these are:

$$\begin{aligned}&q(x,y):=K-c_1x-c_2y\\&\Psi(x,y):=(q(x,y))^+\\&V^{\epsilon,C}(x,y,T)=\Psi(x,y)\\&V^{\epsilon,C}(x,0,t)=G_2(x,t)\\&V^{\epsilon,C}(0,y,t)=G_1(y,t)\\&\lim_{x\to\infty}V^{\epsilon,C}(x,y,t)=\lim_{y\to\infty}V^{\epsilon,C}(x,y,t)=0\,,\end{aligned}$$

for $0\le t\le T$, $x\ge 0$, $y\ge 0$. An equidistant grid on the truncated domain

$$0\le x\le x_{\max}\,,\ 0\le y\le y_{\max}\,,\ 0\le t\le T$$

is defined by $i_{\max}$, $j_{\max}$ and $\nu_{\max}$ subintervals,

$$\begin{aligned}&\Delta x:=\frac{x_{\max}}{i_{\max}},\ x_i:=i\Delta x,\ i=0,\ldots,i_{\max}\,,\\&\Delta y:=\frac{y_{\max}}{j_{\max}},\ y_j:=j\Delta y,\ j=0,\ldots,j_{\max}\,,\\&\Delta t:=\frac{T}{\nu_{\max}},\ t_\nu:=\nu\Delta t,\ \nu=\nu_{\max},\ldots,0\,.\end{aligned}$$

Furthermore, we use the notations

$$\begin{aligned}&q_{i,j}:=q(x_i,y_j)\,,\\&w^\nu_{i,j}\text{ approximation to }V^{\epsilon,C}(x_i,y_j,t_\nu)\,.\end{aligned}$$

To simplify the exposition, we choose $i_{\max}=j_{\max}$, $x_{\max}=y_{\max}$ and use the notation $h:=\Delta x=\Delta y$. The difference quotients are defined in Chap. 4, except for the mixed second-order derivative, which is discretized by the second-order term

$$\delta_{xy}w^\nu_{i,j}:=\frac{1}{2h^2}(w^\nu_{i+1,j+1}-w^\nu_{i+1,j}-w^\nu_{i,j+1}+2w^\nu_{i,j}-w^\nu_{i-1,j}-w^\nu_{i,j-1}+w^\nu_{i-1,j-1}).$$

By stability reasons (⟶ Sects. 6.4 and 6.5) the first-order derivatives with respect to x and y are discretized by upwind schemes. For $\delta_1 \le r$, $\delta_2 \le r$, the upwind schemes are

$$\delta_x w^\nu_{i,j} := \frac{w^\nu_{i+1,j} - w^\nu_{i,j}}{h},$$
$$\delta_y w^\nu_{i,j} := \frac{w^\nu_{i,j+1} - w^\nu_{i,j}}{h},$$

since the integration is backward in time.[8] Substituting all difference quotients into (6.48) is routine.

As in Chap. 4, we may choose among explicit or implicit schemes. The difference quotient

$$\delta_t w^\nu_{i,j} := \frac{w^{\nu+1}_{i,j} - w^\nu_{i,j}}{\Delta t}$$

for the time derivative $\frac{\partial V}{\partial t}$ leads to an explicit scheme when the difference quotients with respect to x, y are evaluated at level $\nu + 1$, and leads to an implicit scheme when the evaluation is at level ν. In the latter case, since we integrate backwards in time, $w^{\nu+1}$ is considered as calculated and the w^ν are to be calculated next. For the explicit scheme, stability requirements lead to severe restrictions on the step size Δt, and to a slow algorithm; it will not be discussed further.

But for the implicit scheme, the nonlinear penalty term (6.47) makes a difference. In case we plug in $w^\nu_{i,j}$ for $V^{\epsilon,C}$, the equation to be solved at time level t_ν is nonlinear and requires an iterative solution. To speed up a Newton iteration, good initial guesses must be made available. These are given by the previous time level, provided the time steps Δt are small. Such a restriction on Δt *due to the nonlinearity* may make the method expensive. But there is an alternative. When $w^{\nu+1}_{i,j}$ is used for $V^{\epsilon,C}$ in the penalty term, then the nonlinearity at time level t_ν is known, and for each ν only a linear system needs to be solved. This procedure is called *semi-implicit* or linear-implicit. The alternative of a fully nonlinear equation [with $w^\nu_{i,j}$ in (6.47)] is referred to as fully implicit.

The semi-implicit scheme now reads

$$\frac{w^{\nu+1}_{i,j} - w^\nu_{i,j}}{\Delta t} + \frac{1}{2}\sigma_1^2 x_i^2\, \delta_{xx} w^\nu_{i,j} + \frac{1}{2}\sigma_2^2 y_j^2\, \delta_{yy} w^\nu_{i,j} + \rho\sigma_1\sigma_2 x_i y_j\, \delta_{xy} w^\nu_{i,j}$$
$$+(r-\delta_1) x_i\, \delta_x w^\nu_{i,j} + (r-\delta_2) y_j\, \delta_y w^\nu_{i,j} - r w^\nu_{i,j} + \frac{\epsilon\, C}{w^{\nu+1}_{i,j} + \epsilon - q_{i,j}} = 0$$

[8] For $\delta_1 > r$ or $\delta_2 > r$ the "other" quotient is upwind.

for $\nu = \nu_{\max} - 1, \ldots, 0$, and $w_{i,j}^{\nu_{\max}} = \Psi(x_i, y_j)$. We leave it to the reader to plug in the difference quotients, to organize the equation, and to introduce a matrix-vector notation for the equation to be solved at time level t_ν. The matrix is sparse, banded, and block-tridiagonal, which calls for specific linear-equation solvers [326].

In [289] the convergence of the explicit, the semi-implicit, and the fully implicit schemes were analyzed for the uncorrelated case $\rho = 0$. In numerical experiments it turns out that the semi-implicit variant is recommendable in terms of accuracy and costs. In case

$$C \geq rK\,,\ \Delta t \leq \frac{\epsilon}{rK} \tag{6.49}$$

holds, the semi-implicit method satisfies the required inequality

$$w_{i,j}^{\nu} \geq \Psi(x_i, y_j)$$

for all ν, see [289]. This restricts the step size Δt to a relatively small value. Hence, one will not choose a too small value of ϵ and do without high demands on the accuracy of $V^{\epsilon,C}$. For example, one chooses $\epsilon = 0.01$ or $\epsilon = 0.001$. But for the fully implicit method the step size Δt must be restricted too in order to maintain the convergence of the Newton method. And the mild bound on Δt in (6.49) does not depend on h (as would do the bound of the explicit method). Our experiments indicate an $O(\epsilon)$ error of $V^{\epsilon,C}$.

6.8 Notes and Comments

On Sect. 6.1

For barrier options we refer, for example, to [15, 304, 346, 385, 386]. Monte Carlo for path-dependent options is discussed in [207]. Dai and Lyuu [96] suggests a tree method with an initial trinomial step tuned so that the following tree has layers coinciding with the barrier. For lookback options we mention [95, 135, 218]. Haug [172] is a rich source of analytical formula for option pricing.

On Sect. 6.2

To see how the multidimensional volatilities of the model enter into a lumped volatility, consult [340]. Other multidimensional PDEs arise when stochastic volatilities are modeled with SDEs, see [20, 163, 185, 293, 384], or Example 5.8. A list of exotic options with various payoffs is presented in Sect. 19.2 of [102]. Also the n-dimensional PDEs can be transformed to simpler forms. For $n = 2$ and $n = 3$ this

is shown in [197]. For the n-dimensional Black–Scholes problem, see [2, 65, 234]. An ADI method is applied to American options on two stocks in [370]. Refined ADI methods work with non-equidistant grids [163]. Consult also the efficient operator splitting method [195], which decouples the treatment of the early-exercise constraint and the solution of the linear system. Further higher-dimensional PDEs related to finance can be found in [353].

On Sect. 6.3

PDEs in the context of Asian options were introduced in [196, 220, 320]. A reduction as in (6.12) from $V(S, A, t)$ to $H(R, t)$ is called *similarity reduction*. The derivation of the boundary-value problem (6.16) follows [376]. For the discrete sampling discussed in Sect. 6.3.4 see [376, 385]. The strategies introduced for Asian options work similarly for other path-dependent options. An overview on methods for Asian options, and a semianalytical method are found in [379].

On Sect. 6.4

The von Neumann stability analysis is tailored to linear schemes and pure initial-value problems. It does not rigorously treat effects caused by boundary conditions. In this sense it provides a necessary stability condition for boundary-value problems. For a rigorous treatment of stability see [356, 357]. The stability analysis based on eigenvalues of iteration matrices as used in Chap. 4 is an alternative to the von Neumann analysis.

Spurious oscillations are special solutions of the difference equations and do not correspond to solutions of the differential equation. The spurious oscillations are not related to rounding errors. This may be studied analytically for the simple ODE model boundary-value problem $au' = bu''$, which is the steady state of (6.20), along with boundary conditions $u(0) = 0$, $u(1) = 1$. Here for mesh Péclet numbers $\frac{a\Delta x}{b} > 2$ the analytical solution of the discrete centered-space analog is oscillatory, whereas the solution $u(x)$ of the differential equation is monotone, see [281]. The model problem is extensively studied in [281, 298]. The mesh Péclet number is also called "algebraic Reynold's number of the mesh."

On Sect. 6.5

It is recommendable to derive the equivalent differential equation in Sect. 6.5.2.

On Sect. 6.6

The Lax–Wendroff method is an example of a *finite-volume method*. Another second-order scheme for (6.26) is the *leapfrog* scheme $\delta_t^2 w + a\delta_x^2 w = 0$, which involves three time levels. The discussion of monotonicity is based on investigations of Godunov, see [232, 374]. The Lax–Wendroff scheme for (6.26) and $\gamma \geq 0$ can also be written

$$w_j^{\nu+1} = w_j^\nu - \tfrac{1}{2}\gamma(w_{j+1}^\nu - w_{j-1}^\nu) + \tfrac{1}{2}\gamma^2(w_{j+1}^\nu - 2w_j^\nu + w_{j-1}^\nu)\,.$$

(This version adopts the frequent notation w_j^ν for our $w_{j,\nu}$.) Here the diffusion term has a slightly different factor than (6.39). The numerical dissipation term is also called *artificial viscosity*. In [374, p. 348], the Lax–Wendroff scheme is embedded in a family of schemes. A special choice of the family parameter yields a third-order scheme. The TVD criterion can be extended to implicit schemes and to schemes that involve more than two time levels. For the general analysis of numerical schemes for conservation laws (6.34) we refer to [232].

On Sect. 6.7

In [289] the linear systems were solved iteratively with the bi-conjugate gradient method Bi-CGSTAB [326, 366]. Choosing Δt small provides good initial guesses for the next time level, which accelerates the iteration. Hence the limitation $\Delta t \leq \frac{\epsilon}{rK}$ is not too severe in practice. In our experiments, the penalty method did not achieve better results than a simple binomial-tree method. For the convergence of penalty methods consult [133]. A penalty method with a smooth penalty has been implemented with finite elements in [230]. The weak formulation (compare Sect. 5.4) works with the relatively simple choice of boundary conditions $V = \Psi$ along the boundary. Exercise 6.8 follows [288].

On Other Methods

Computational methods for exotic options are under rapid development. The universal binomial method can be adapted to exotic options [203, 224]. Tavella and Randall [353] gives an overview on a class of PDE solvers. For barrier options see [144, 385, 386]. For two-factor barrier options and their finite-element solution, see [304]. PDEs for lookback options are given in [21]. Using Monte Carlo for path-dependent options, considerable efficiency gains are possible with bridge techniques [315, 316]. For Lévy process models, see, for example, [7, 84]. We recommend to consult, for example, the issues of the *Journal of Computational Finance*.

6.9 Exercises

6.1 (Towards the Black–Scholes Equation)

(a) For the model equation (6.4) set up the vector a and the matrix b for the general vector notation (1.57).

(b) Let LL^{tr} be the Cholesky decomposition of the ρ-matrix, and $\tilde{b} := bL$. Show

$$\text{trace}(\tilde{b}\tilde{b}^{tr}V_{SS}) = \sum_{i,j=1}^{n} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} .$$

(c) Show

$$\begin{aligned} \mathrm{d}V = & \left[\frac{\partial V}{\partial t} + \sum_{i=1}^{n} (\mu_i - \delta_i) S_i \frac{\partial V}{\partial S_i} + \frac{1}{2} \sum_{i,j=1}^{n} \rho_{ij}\sigma_i\sigma_j S_i S_j \frac{\partial^2 V}{\partial S_i \partial S_j} \right] \mathrm{d}t \\ & + \sum_{i=1}^{n} \sigma_i S_i \frac{\partial V}{\partial S_i} \mathrm{d}W^{(i)} . \end{aligned}$$

6.2 (Tree for Two Assets)
A two-asset extension of the binomial tree with (x, y)-coordinates representing the assets, and time-coordinate t, is assumed to develop as follows: Each node with position (x, y) may develop for $t \to t + \Delta t$ with equal probabilities 0.25 to one of the four positions

$$(xu, yA),\ (xu, yB),\ (xd, yC),\ (xd, yD) , \tag{6.50}$$

for constants u, d, A, B, C, D.

(a) Show that the tree is recombining for $AD = BC$.
Hint: Sketch the possible values in a (x, y)-plane.

Following [323], a tree is defined for interest rate r, asset volatilities σ_1, σ_2, correlation ρ, and dividend yield rates δ_1, δ_2, by

$$\begin{aligned} \mu_i &:= r - \delta_i - \sigma_i^2/2 \ \text{ for } i = 1, 2 \\ u &:= \exp(\mu_1 \Delta t + \sigma_1 \sqrt{\Delta t}) \\ d &:= \exp(\mu_1 \Delta t - \sigma_1 \sqrt{\Delta t}) \\ A &:= \exp(\mu_2 \Delta t + \sigma_2 \sqrt{\Delta t}\,[\rho + \sqrt{1-\rho^2}]) \\ B &:= \exp(\mu_2 \Delta t + \sigma_2 \sqrt{\Delta t}\,[\rho - \sqrt{1-\rho^2}]) \\ C &:= \exp(\mu_2 \Delta t - \sigma_2 \sqrt{\Delta t}\,[\rho - \sqrt{1-\rho^2}]) \\ D &:= \exp(\mu_2 \Delta t - \sigma_2 \sqrt{\Delta t}\,[\rho + \sqrt{1-\rho^2}]) . \end{aligned}$$

For initial prices $x^0 := S_1^0$, $y^0 := S_2^0$, and time level $t_\nu := \nu\Delta t$, the S_1-components of the grid according to (6.50) distribute in the same way as for the one-dimensional tree,

$$x_i^\nu := S_1^0 u^i d^{\nu-i} \quad \text{for } i = 0, \ldots, \nu\,.$$

(b) Verify that the choice of A, B, C, D sets up a recombining tree.

(c) Show that the second (S_2-)components belonging to x_i^ν are

$$y_{i,j}^\nu := S_2^0 \exp(\mu_2 \nu \Delta t)\, \exp\left(\sigma_2 \sqrt{\Delta t}\left[\rho(2i-\nu) + \sqrt{1-\rho^2}(2j-\nu)\right]\right)$$

for $j = 0, \ldots, \nu$.

Hint: For $\nu \to \nu+1$, u corresponds to $i \to i+1$, and d corresponds to $i \to i$.

(d) Show that the first two moments of the continuous and the discrete model match: Verify that for the log-variables $\Delta Y_1 := \log u$ or $\log d$, and $\Delta Y_2 := \log A$, $\log B$, $\log C$, or $\log D$ the five equations

$$\mathsf{E}(\Delta Y_i) = \mu_i \Delta t\,, \quad \mathsf{Var}(\Delta Y_i) = \sigma_i^2 \Delta t\,, \quad \mathsf{Cov}(\Delta Y_1, \Delta Y_2) = \rho\sigma_1\sigma_2\Delta t$$

hold [for $i = 1, 2$ and the four probabilities $\frac{1}{4}$ associated to (6.50)].

(e) Set up a computer program that implements this binomial method. Analogously as in Sect. 1.4 work in a backward recursion for $\nu = M, \ldots, 0$. For each time level t_ν set up the (x, y)-grid with the above rules and $\Delta t = T/M$. For $t_M = T$ fix V by the payoff Ψ, and use for $\nu < M$

$$V_{i,j}^{\text{cont}} = \exp(-r\Delta t)\,\frac{1}{4}(V_{i,j}^{\nu+1} + V_{i+1,j}^{\nu+1} + V_{i,j+1}^{\nu+1} + V_{i+1,j+1}^{\nu+1})\,.$$

Test example: max call with $\Psi(S_1, S_2) = (\max(S_1, S_2) - K)^+$, $S_1^0 = S_2^0 = K = T = 1$, $r = 0.1$, $\sigma_1 = 0.2$, $\sigma_2 = 0.3$, $\rho = 0.25$, $\delta_1 = 0.05$, $\delta_2 = 0.3$. For $M = 2000$ an approximation of the American-style option is 0.130302, and for the European style 0.120036.

6.3 (Project: Monte Carlo Valuation of Exotic Options)
Perform Monte Carlo valuations of barrier options, basket options, and Asian options, each European style.

6.4 (PDEs for Arithmetic Asian Options)

(a) Use the higher-dimensional Itô-formula (⟶ Appendix B.2) to show that the value function $V(S, A, t)$ of an Asian option satisfies

$$\mathrm{d}V = \left(\frac{\partial V}{\partial t} + S\frac{\partial V}{\partial A} + \mu S\frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t + \sigma S \frac{\partial V}{\partial S}\,\mathrm{d}W\,,$$

where S is the price of the asset and A its average.

(b) Construct a suitable riskless portfolio and derive the Black–Scholes equation

$$\frac{\partial V}{\partial t} + S\frac{\partial V}{\partial A} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,.$$

(c) Use the transformation $V(S,A,t) = \widetilde{V}(S,R,t) = SH(R,t)$, with $R = \frac{A}{S}$ and transform the Black–Scholes equation (6.8) to

$$\frac{\partial H}{\partial t} + \frac{1}{2}\sigma^2 R^2 \frac{\partial^2 H}{\partial R^2} + (1 - rR)\frac{\partial H}{\partial R} = 0\,.$$

(d) From

$$R_{t+\,\mathrm{d}t} = R_t + dR_t\,, \quad \mathrm{d}S_t = \mu S_t\,\mathrm{d}t + \sigma S_t\,\mathrm{d}W_t$$

derive the SDE

$$\mathrm{d}R_t = (1 + (\sigma^2 - \mu)R_t)\,\mathrm{d}t - \sigma R_t\,\mathrm{d}W_t\,.$$

(e) For

$$A_t := \frac{1}{t}\int_0^t S_\theta\,\mathrm{d}\theta$$

show $\mathrm{d}A = \frac{1}{t}(S - A)\,\mathrm{d}t$ and derive the PDE

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

6.5 (Neumann Stability Analysis)
Assume a difference scheme in the form (6.37)

$$w_j^{(\nu+1)} = \sum_l d_l\, w_{j+l}^{(\nu)}$$

and make use of the Fourier transform (6.22)

$$w_j^{(\nu)} = \sum_{k=0}^{n-1} c_k^{(\nu)} \mathrm{e}^{\mathrm{i}k\eta j\Delta x} \quad \text{for } \eta = \frac{2\pi}{n\Delta x}\,.$$

(a) What are the coefficients d_l for the FTCS method (6.21)?
(b) Prove linear independence

$$\sum_{k=0}^{n-1} \alpha_k \exp[\mathrm{i}\frac{2\pi}{n}kj] = 0 \quad \Longrightarrow \quad \alpha_k = 0 \text{ for all } k$$

Hint: FFT equivalence (C.7).
(c) Show

$$c_k^{(\nu+1)} = c_k^{(\nu)} \sum_l d_l \, \mathrm{e}^{\mathrm{i}k\eta l \Delta x} .$$

6.6 (Upwind Scheme)
Apply von Neumann's stability analysis to

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = b\frac{\partial^2 u}{\partial x^2} , \; a > 0, \, b > 0$$

using the upwind scheme for the left-hand side and the centered second-order difference quotient for the right-hand side.

6.7 (TVD of a Model Problem)
Analyze whether the upwind scheme (6.27), the Lax–Friedrichs scheme (6.30) and the Lax–Wendroff scheme (6.35) applied to the scalar partial differential equation

$$u_t + a u_x = 0 , \quad a > 0, \; t \geq 0, \; x \in \mathbb{R}$$

satisfy the TVD property.
Hint: Apply Lemma 6.3.

6.8 (Initial-Value Problem with Penalty Term)
Consider the ODE initial-value problem

$$u' = -u , \quad u(0) = 2$$

with the additional constraint

$$u(t) \geq 1 .$$

(a) Give an analytical solution.
(b) Discuss for a value of ϵ with $0 < \epsilon \ll 1$ the initial-value problem

$$v' = -v + \frac{\epsilon}{v - 1 - \epsilon}\,, \quad v(0) = 2\,.$$

Hint: Do some numerical experiments.
(c) Show that the solution $v(t)$ of the initial-value problem in (b) satisfies

$$1 \le v \le 2\,, \quad v' \le 0\,, \quad v'' \ge 0\,,$$

for $t \ge 0$.

Chapter 7
Beyond Black and Scholes

The Black–Scholes (BS) model for the value $V(S,t)$ of a vanilla option is based on some assumptions on the market. In particular, the BS model assumes the price S_t of the asset on which the option is written, follows a geometric Brownian motion with a constant volatility σ. Further, transaction costs are neglected, and trading of the underlying is supposed to have no influence on the price S_t. As has been discussed extensively, the value function $V(S,t)$ for standard options ("plain vanilla") of the European type, satisfies the Black–Scholes equation (1.5),

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,. \tag{7.1}$$

Solutions of this linear equation are subject to the terminal condition $V(S,T) = \Psi(S)$, where Ψ defines the payoff.

The BS-model is the core example of a complete market. In these idealized markets, the risk exposure to variations in the underlying can be hedged away. The corresponding risk strategy is unique. Hence vanilla options modeled by Assumptions 1.2 have a unique price, given by the costs of the replication strategy (⟶ Appendix A.4). Essentially, Chaps. 4 through 6 have applied numerical methods to complete markets.

For the more realistic incomplete markets, there are no perfect hedges, and a risk remains. Each hedging strategy leads to a specific model with its own price [84]. The hedger compensates the remaining risk in incomplete markets by charging an additional risk premium. Hence the value function or expected value is not the price for which the option is sold. Depending on the way how the comfortable assumption of completeness of the BS-market is lost, different models are set up, calling for different numerical approaches. This Chap. 7 is devoted to computational tools for incomplete markets.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0_7

Relaxing several of the assumptions of the Black–Scholes market, *nonlinear* extensions of the BS equation can be derived. These "nonlinear Black–Scholes type equations" are of the form

$$\frac{\partial V}{\partial t} + \frac{1}{2}\left[\hat{\sigma}(S,t,\frac{\partial^2 V}{\partial S^2})\right]^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,. \tag{7.2}$$

In this class of models, the volatility $\hat{\sigma}$ is a function that may incorporate several types of nonlinearity. The standard PDE (7.1) is included for $\hat{\sigma} \equiv \sigma$. In Sect. 7.1 we describe three scenarios leading to three different functions $\hat{\sigma}$ of the volatility. A nonlinear PDE as (7.2) requires special numerical treatment, which will be the focus of Sect. 7.2.

Another stream of research beyond Black and Scholes is devoted to jump processes (Sect. 7.3). One of the numerical approaches is based on partial integro-differential equations (PIDE). Some highly efficient methods apply the Fourier transform; a basic approach will be discussed in Sect. 7.4.

7.1 Nonlinearities in Models for Financial Options

In this section we briefly discuss three sources of nonlinearity of $\hat{\sigma}$ in (7.2). We start with transaction costs based on Leland's approach [245], and touch the more sophisticated model of Barles and Soner [24]. Then we turn to specifying ranges of volatility. Finally we address feedback by market illiquidity.

7.1.1 Leland's Model of Transaction Costs

Basic for the Black–Scholes model is the idea of rebalancing the portfolio continuously. But in financial reality this continuous trading would cause arbitrarily high trading costs. Keeping transaction costs low forces to abandon the optimal Black–Scholes hedging. But without the ideal BS hedging, the model suffers from hedging errors. To compromise, the hedger searches a balance between keeping both the transaction costs low *and* the hedging errors low.

Suppose that instead of rebalancing continuously, trading is only possible at discrete time instances with time step Δt apart (Δt fixed and finite). We assume a transaction cost rate proportional to the trading volume νS:

$$\text{trading } \nu \text{ assets costs the amount } c|\nu|S$$

for some cost parameter c.

Here we sketch a heuristic derivation of a model due to [187, 245]. The discussion of this model parallels that for the Black–Scholes model, now adapted to the discrete scenario.[1] The stochastic changes of the asset with price S and of a riskless bond with price B are

$$\begin{aligned} \Delta S &= \mu S \Delta t + \sigma S \Delta W \,, \\ \Delta B &= r B \Delta t \,. \end{aligned}$$

The portfolio with value Π is taken in the form

$$\Pi = \alpha S + \beta B \,,$$

with α units of the asset and β units of the bond. Suppose the portfolio is self-financing in the sense $S\Delta\alpha + B\Delta\beta = 0$, which is sufficient for $\Delta\Pi = \alpha\Delta S + \beta\Delta B$. Further assume that trading is such that the portfolio Π replicates the value of the option.

By definition, $\nu = \Delta\alpha$. After one time interval, $\nu = \Delta\alpha$ assets are traded, with transaction costs $cS|\Delta\alpha|$. The change in the value of the portfolio is

$$\begin{aligned} \Delta\Pi &= \alpha\Delta S + \beta\Delta B - cS|\Delta\alpha| \\ &= (\alpha\mu S + \beta r B)\Delta t + \alpha\sigma S\Delta W - cS|\Delta\alpha| \,. \end{aligned} \tag{7.3}$$

Let V be the value function of the option. Itô's lemma adapted to the discrete scenario gives

$$\Delta V = \frac{\partial V}{\partial S}\,\Delta S + \left(\frac{\partial V}{\partial t} + \frac{\sigma^2}{2} S^2 \frac{\partial^2 V}{\partial S^2}\right)\Delta t \,.$$

By the no-arbitrage principle $\Delta V = \Delta\Pi$ holds for the replicating and self-financing portfolio. And coefficient matching will give further information. But first let us approximate the $\Delta\alpha$-term.

From BS theory we expect $\alpha \approx \frac{\partial V}{\partial S}$. So $\nu = \Delta\alpha$ will be approximated by

$$\begin{aligned} &\frac{\partial V(S+\Delta S, t+\Delta t)}{\partial S} - \frac{\partial V(S,t)}{\partial S} \\ &= \frac{\partial^2 V(S,t)}{\partial S^2}\,\Delta S + \frac{\partial^2 V(S,t)}{\partial S\,\partial t}\,\Delta t + \text{t.h.o.} \,, \end{aligned}$$

[1] All other BS-assumptions remain untouched [234]. The following analysis uses or modifies Appendix A.4 with (A.3), (A.5), (A.10). Here Δ means the increment, and *not* the greek $\frac{\partial V}{\partial S}$.

invoking Taylor's expansion. After substituting ΔS we realize that the term of lowest order is

$$\sigma S \frac{\partial^2 V(S,t)}{\partial S^2}\, \Delta W\,.$$

In summary, by (7.3) the transaction costs in $\Delta\Pi$ can be approximated by

$$-cS|\Delta\alpha| = -c\sigma S^2 \left|\frac{\partial^2 V(S,t)}{\partial S^2}\right| |\Delta W| + \text{t.h.o.}\,,$$

which is path-dependent. Leland [245] boldly suggested to approximate $|\Delta W| \approx \mathsf{E}(|\Delta W|)$. Exercise 7.1 tells

$$\mathsf{E}(|\Delta W|) = \sqrt{\Delta t}\,\sqrt{\frac{2}{\pi}}\,.$$

In this way, the trading cost term $-cS\,|\Delta\alpha|$ is approximated by the deterministic expression

$$-\,c\sigma S^2 \left|\frac{\partial^2 V(S,t)}{\partial S^2}\right| \sqrt{\Delta t}\,\sqrt{\frac{2}{\pi}}\,. \tag{7.4}$$

This may be seen as further assumption, motivated by the above arguing. The approximation (7.4) of the transaction costs and its artificial parameter $\sqrt{2/\pi} \approx 0.8$ reflect the lack of a unique price in incomplete markets.

With this somewhat artificial approximation (7.4) of the trading costs $-cS|\Delta\alpha|$, coefficient matching of $\Delta V = \Delta\Pi$ leads to match the remaining stochastic terms,

$$\alpha\sigma S\Delta W = \sigma S\frac{\partial V}{\partial S}\,\Delta W\,,$$

or $\alpha = \frac{\partial V}{\partial S}$, which is the famous "delta hedging," consistent with the modeling of $\Delta\alpha$ above. The remaining terms are deterministic. Use $\beta B + S\frac{\partial V}{\partial S} = \Pi = V$ to obtain

$$\begin{aligned}&\left(\mu S\frac{\partial V}{\partial S} + rV - rS\frac{\partial V}{\partial S}\right)\Delta t - cS|\Delta\alpha| \\ &= \left(\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2} + \mu S\frac{\partial V}{\partial S}\right)\Delta t\,.\end{aligned} \tag{7.5}$$

The μ-terms cancel out. Equation (7.5) with transaction costs replaced by (7.4) leads to the variant of the Black–Scholes equation. With the coefficient

$$\gamma := \sqrt{\frac{2}{\pi}}\left(\frac{2c}{\sigma\sqrt{\Delta t}}\right) \tag{7.6}$$

the resulting equation is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + \frac{1}{2}\sigma^2 S^2 \gamma \left|\frac{\partial^2 V}{\partial S^2}\right| + rS\frac{\partial V}{\partial S} - rV = 0\,. \tag{7.7}$$

Formally, this becomes the standard Black–Scholes equation with a modified volatility

$$\hat{\sigma}^2(\Gamma) := \sigma^2[1 + \gamma\ \mathrm{sign}(\Gamma)]\,, \tag{7.8}$$

with $\Gamma := \frac{\partial^2 V}{\partial S^2}$. For convex payoff, $\mathrm{sign}(\Gamma) = 1$. This amounts to augment the volatility to a constant $\hat{\sigma} > \sigma$ (Leland's scenario). In this case the PDE (7.7) is again linear. But note that for instance for barrier options, Γ does change sign, and the PDE is nonlinear and of the general type of Eq. (7.2). For $c = 0$ (no transaction costs) (7.7) specializes to the BS-equation. To have a well-posed PDE, Δt must be such that $\gamma < 1$. In particular, $\Delta t \to 0$ does not make sense.

7.1.2 The Barles and Soner Model of Transaction Costs

Barles and Soner [24] assume a price process $\mathrm{d}S_t = S_t(\mu\,\mathrm{d}t + \sigma\,\mathrm{d}W_t)$, with constant volatility σ, $0 \le t \le T$, and model transactions using the following variables:

α_t shares of the asset with price S_t,
β_t shares of the bond,
L_t cumulative transfer form cash to stock, nondecreasing, $L(0) = 0$,
M_t cumulative transfer from stock to cash, nondecreasing, $M(0) = 0$.

Consequently,

$$\begin{aligned}
\alpha_t &= \alpha_0 + L_t - M_t\,,\\
\beta_t &= \beta_0 - \int_0^t S_\tau \cdot (1+c)\,\mathrm{d}L_\tau + \int_0^t S_\tau \cdot (1-c)\,\mathrm{d}M_\tau + \int_0^t r\beta_\tau\,\mathrm{d}\tau\,.
\end{aligned}$$

That is, in both the cases buying and selling of stocks, transaction costs $\int S_\tau c$ are charged to β, where c again denotes proportional transaction costs. The further derivation of [24] is based on a utility function. The final result is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \cdot \left[1 + f\left(\mathrm{e}^{r(T-t)} a^2 S^2 \frac{\partial^2 V}{\partial S^2}\right)\right] + rS\frac{\partial V}{\partial S} - rV = 0\,, \tag{7.9}$$

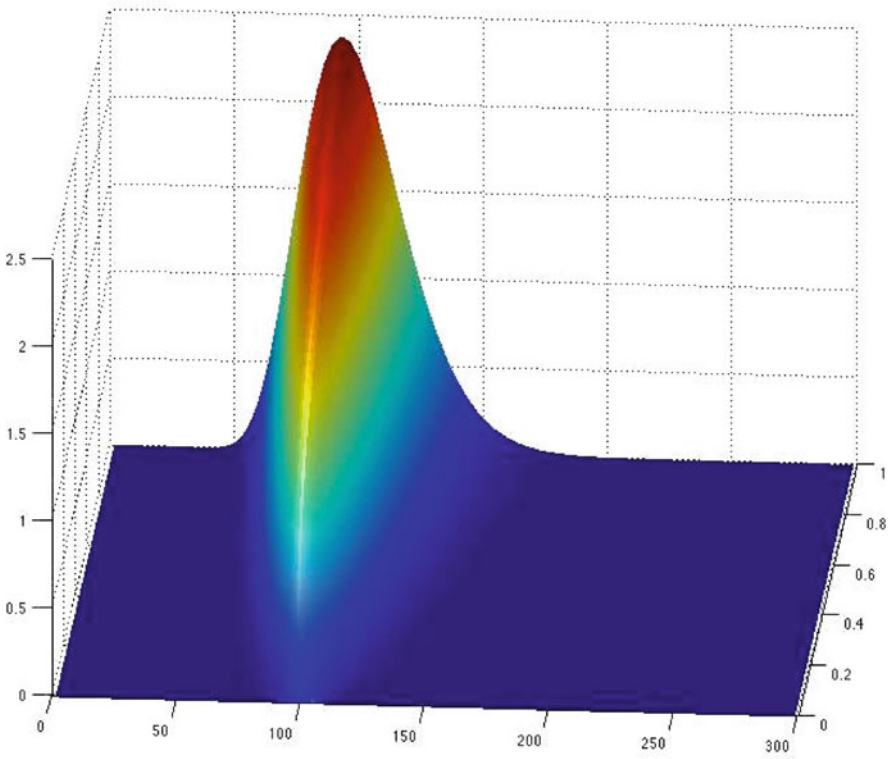

Fig. 7.1 $V(S, T-t)$: difference between the solution of the Black–Scholes equation (7.1) and the solution of (7.9); $K = 100, r = 0.1, \sigma = 0.2, a = 0.02, T = 1$. With kind permission of Pascal Heider

where a is a parameter representing proportional transaction costs and risk aversion. The function f is the unique solution of the ODE

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = \frac{f(x)+1}{2\sqrt{xf(x)}-x} \quad \text{with } f(0) = 0\,.$$

The resulting function f is singular at $x = 0$ ($\longrightarrow$ Exercise 7.2). Figure 7.1 shows the difference between the BS-solution and the solution of the corresponding nonlinear model (7.9).

7.1.3 Specifying a Range of Volatility

The two above models of transaction costs come up with a nonlinear volatility function $\hat{\sigma}(\Gamma)$. Usually this function is not known, and is subject to speculation (modeling). It will be easier to specify a *range* of volatility, assuming that $\hat{\sigma}$ lies

within an interval or band

$$0 < \sigma_{\min} \le \sigma \le \sigma_{\max} < 1 \,.$$

This is the uncertain-volatility model of [16, 17, 250].

The derivation starts as above, leading to (7.5) with $c = 0$. (Here transaction costs are not considered.) Formally, the result is the Black–Scholes equation (BSE), except that σ is no constant, but is considered as a stochastic variable $\sigma(t)$:

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

This is a PDE with stochastic control parameter $\sigma(t)$. There is an ambitious theory for such controlled diffusion processes, see the monograph [233]. To avoid the use of this methodology, we adopt a simplified arguing, similar as in [375].

Using an argumentation of Black and Scholes, we construct a portfolio of one option (value V), and hedge it with $-\alpha$ units of the underlying asset,

$$\Pi = V - \alpha S \,.$$

Assuming a change in the value of this portfolio in the form $\Delta\Pi = \Delta V - \alpha\Delta S$, we have as above

$$\Delta\Pi = \frac{\partial V}{\partial S}\Delta S + \left(\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}\right)\Delta t - \alpha\Delta S \,.$$

The choice $\alpha = \frac{\partial V}{\partial S}$ eliminates the risk represented by the ΔW-terms. This results in

$$\Delta\Pi = \left(\frac{\partial V}{\partial t} + \frac{\sigma^2}{2}S^2\frac{\partial^2 V}{\partial S^2}\right)\Delta t \,. \tag{7.10}$$

Note that the return $\Delta\Pi$ of the portfolio still depends on the unknown stochastic $\sigma(t)$, we write $\Delta\Pi(\sigma)$.

Now we define artificially two specific functions $\sigma^+(t)$ and $\sigma^-(t)$ chosen such that the return $\Delta\Pi(\sigma)$ increases by the maximum amount, or by the least amount:

- $\sigma^+(t)$ chosen such that $\Delta\Pi(\sigma^+)$ is a maximum,
- $\sigma^-(t)$ chosen such that $\Delta\Pi(\sigma^-)$ is a minimum.

These returns reflect the best case and the worst case as seen by the holder. For every function $\sigma(t)$ the no-arbitrage principle holds. Hence both cases $\sigma^+(t)$ and $\sigma^-(t)$ result in a return $\Delta\Pi = r\Pi\Delta t$. This can be summarized as

$$\begin{aligned} \sigma^+ \text{ maximizes} \quad & \max_{\sigma_{\min}\le\sigma\le\sigma_{\max}} \Delta\Pi(\sigma) = r\Pi\Delta t \,, \\ \sigma^- \text{ minimizes} \quad & \min_{\sigma_{\min}\le\sigma\le\sigma_{\max}} \Delta\Pi(\sigma) = r\Pi\Delta t \,. \end{aligned}$$

In view of the expression (7.10) for $\Delta\Pi(\sigma)$, the two artificial functions σ^+, σ^- enter via the term

$$\sigma^2 \frac{\partial^2 V}{\partial S^2}\,.$$

For $\Delta\Pi$ to become a maximum or minimum, σ^+ (or σ^-) will equal $\sigma_{\min}$ or $\sigma_{\max}$, depending on the sign of $\Gamma = \frac{\partial^2 V}{\partial S^2}$. To become a maximum, set

$$\sigma^+(\Gamma) := \begin{cases} \sigma_{\max} & \text{if } \Gamma \geq 0\,, \\ \sigma_{\min} & \text{if } \Gamma < 0\,. \end{cases} \tag{7.11}$$

And to become a minimum, set

$$\sigma^-(\Gamma) := \begin{cases} \sigma_{\max} & \text{if } \Gamma < 0\,, \\ \sigma_{\min} & \text{if } \Gamma \geq 0\,. \end{cases} \tag{7.12}$$

Equations (7.11) and (7.12) define two specific control functions σ, which after substitution into the PDE $\Delta\Pi(\sigma) = r\Pi\Delta t$ yields *two* nonlinear PDEs

$$\frac{\partial V}{\partial t} + \frac{1}{2}\hat{\sigma}(\Gamma)^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,, \tag{7.13}$$

with $\hat{\sigma} = \sigma^+$ and $\hat{\sigma} = \sigma^-$ from (7.11)/(7.12). Let us denote the corresponding solutions V^+ and V^-. Since σ^+ yields the maximum return, we expect $V \leq V^+$, and similarly, $V^- \leq V$. This provides the range $V^- \leq V \leq V^+$ for the option price.

In the special case of vanilla options, the convexity of $V(S,.)$ implies $\Gamma \geq 0$ and hence $\sigma^+ = \sigma_{\max}$ and $\sigma^- = \sigma_{\min}$; the nonlinearity is not effective then. The monotonicity of V with respect to σ is clear for vanilla options, but is not valid, for example, for barrier options. And convexity of $V(S,.)$ is lost for barrier options, butterfly spreads, digital options, and many other options [303]. The great potential of the uncertain-volatility model is illustrated by Fig. 7.2. For the example of a butterfly option, and an uncertainty interval $0.15 \leq \sigma \leq 0.25$ we show the band $V^- \leq V \leq V^+$, with two Black–Scholes curves therein. The payoff of a butterfly spread is illustrated schematically in Fig. 1.25d, see also Exercise 7.3. The functions V^-, V^+ were calculated with the methods to be explained in Sect. 7.2. For barrier options, the success of the method is doubtful because of the high sensitivity w.r.t. σ close to the barrier. Then the bandwidth may be so large that it is not of practical use. Such an example is shown in Fig. 7.3.

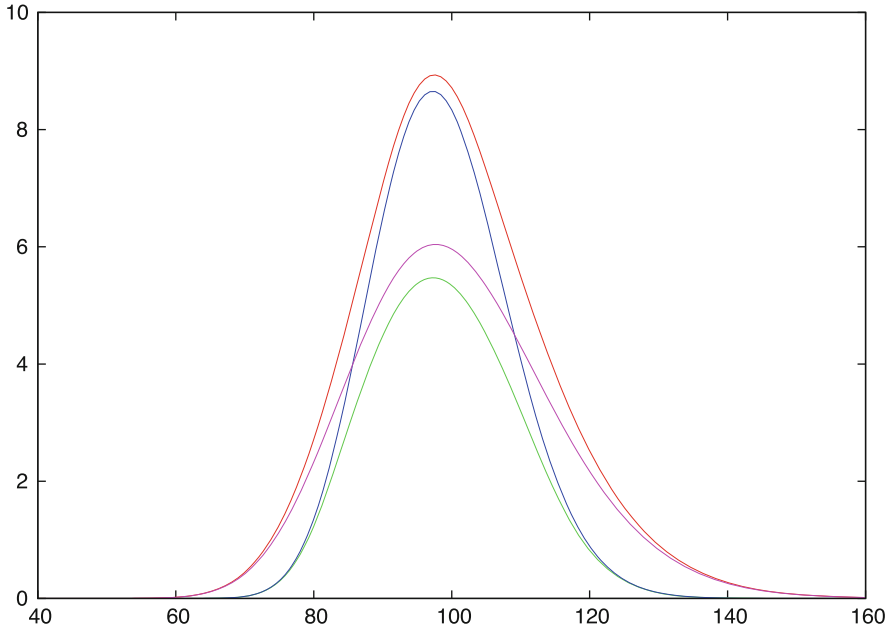

Fig. 7.2 $V(S,0)$ of a European butterfly spread, uncertain-volatility model of Avellaneda et al., Sect. 7.1.3; with $K = 100$, $K_1 = 85$, $K_2 = 115$, $r = 0.13$, $\sigma_{\min} = 0.15$, $\sigma_{\max} = 0.25$, $\delta = 0.03$, $T = 0.27$. Four curves are shown: the bounding functions V^+ (*orange curve*) and V^- (*green curve*), and V of the standard Black–Scholes model with constant volatilities $\sigma = 0.15$ (the steeper curve, in *blue*) and $\sigma = 0.25$ (the lower profile, in *violet*)

7.1.4 Market Illiquidity

As pointed out by [140, 141, 330], the assumption that a big investor can trade large amounts of an asset without affecting its price, is not realistic. There will be a feedback, and the assumption of an infinite market liquidity may fail. Frey and Stremme [141], Schönbucher and Wilmott [330] introduce a market liquidity parameter λ, with $0 \le \lambda \le 1$, and derive the nonlinear PDE

$$\frac{\partial V}{\partial t} + \frac{1}{2}\frac{\sigma^2 S^2}{(1 - \lambda \frac{\partial^2 V}{\partial S^2})^2}\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,. \tag{7.14}$$

Here we do not discuss further details. Note that this model is also of the form of Eq. (7.2).

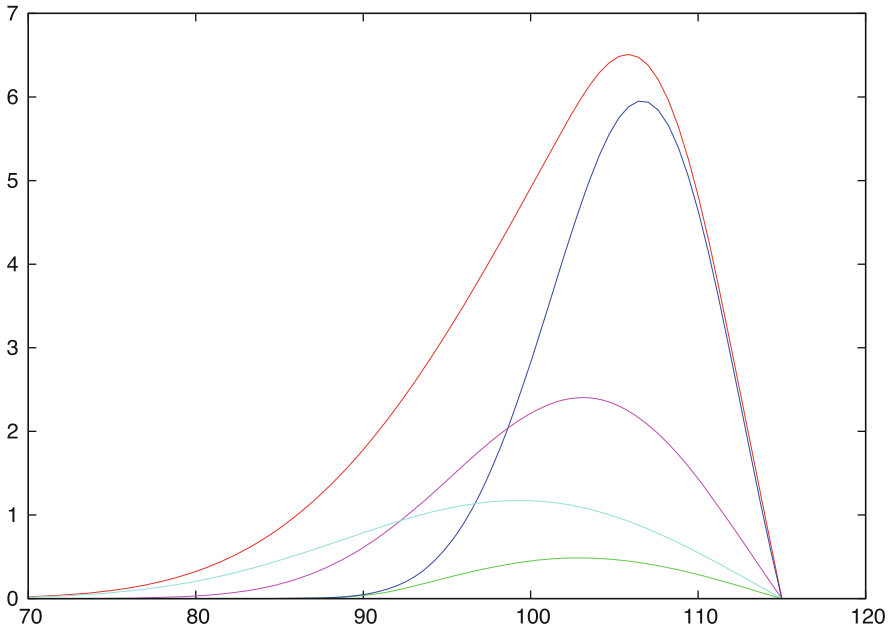

Fig. 7.3 $V(S,0)$ of a European up-and-out barrier call, uncertain-volatility model of Avellaneda et al., Sect. 7.1.3; with barrier $B = 115$, and $K = 100$, $r = 0.1$, $\sigma_{\min} = 0.1$, $\sigma_{\max} = 0.3$, $\delta = 0$, $T = 0.2$. In addition to the two bounding curves V^+ (*orange*) and V^- (*green*) three V curves are shown of the standard Black–Scholes model with constant volatilities $\sigma = 0.1$ (*blue*) and $\sigma = 0.2,\ 0.3$

7.2 Numerical Solution of Nonlinear Black–Scholes Equations

All the nonlinear PDEs of Sect. 7.1 fall under the general type of equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\hat{\sigma}^2(S,t,\frac{\partial^2 V}{\partial S^2})S^2\frac{\partial^2 V}{\partial S^2} + (r-\delta)S\frac{\partial V}{\partial S} - rV = 0\,, \tag{7.15}$$

which we are going to solve next. In this form, Eq. (7.15) represents the value of a European-style option. There is no analytical solution known for (7.15), so a numerical approach is needed also in the European case.

For an American-style option, a penalization can be applied, and an additional nonlinear term appears in (7.15). A penalty approach (e.g., [119, 133]) is to add the penalty $\hat{p}\max(\Psi - V,\ 0)$, where Ψ denotes the payoff, and the penalty parameter $\hat{p}$ is chosen large, say, $\hat{p} = 10^6$. The resulting PDE is

$$\frac{\partial V}{\partial t} + \frac{1}{2}\hat{\sigma}^2(S,t,\frac{\partial^2 V}{\partial S^2})S^2\frac{\partial^2 V}{\partial S^2} + (r-\delta)S\frac{\partial V}{\partial S} - rV + \hat{p}\max(\Psi - V,\ 0) = 0\,. \tag{7.16}$$

In the continuation region, for $V \geq \Psi$, the penalty term is zero, and (7.15) results. For $\hat{p} \to \infty$, think of dividing the equation by $\hat{p}$ to be convinced that V sticks close to Ψ. In Chap. 4, we could preserve the linear equation by the elegant complementarity approach. In (7.16) the PDE is nonlinear by the volatility function $\hat{\sigma}$, and thus the nonlinear penalty term does not cause further harm.

7.2.1 Transformation

The transformation (4.3) of Chap. 4 is not valid here, because the volatility $\hat{\sigma}$ is no longer constant. But assuming constant r, δ, the independent variables S, t can be transformed similarly. The transformation from variables S, t, V to x, τ, u is

$$x := \log \frac{S}{K}\,, \quad \tau := \frac{1}{2}\sigma_0^2 \cdot (T-t)\,, \quad u(x,\tau) := \mathrm{e}^{-x}\frac{V(S,t)}{K}\,. \tag{7.17}$$

σ_0 is a scaling parameter. As a result of the transformation, $V_S = u + u_x$ and $SV_{SS} = u_x + u_{xx}$. Here we use the notations $V_S, V_{SS}, u_\tau, u_x, u_{xx}$ for partial derivatives. And (7.15) becomes

$$-u_\tau + \tilde{\sigma}^2(x,\tau,u_x,u_{xx})(u_x+u_{xx}) + \frac{2(r-\delta)}{\sigma_0^2}u_x - \frac{2\delta}{\sigma_0^2}u = 0 \tag{7.18}$$

with

$$\tilde{\sigma} := \frac{1}{\sigma_0}\,\hat{\sigma}\left(S,t,\frac{\partial^2 V}{\partial S^2}\right) = \frac{1}{\sigma_0}\,\hat{\sigma}\left(K\mathrm{e}^x,\; T-\frac{2\tau}{\sigma_0^2},\; \frac{\mathrm{e}^{-x}}{K}(u_x+u_{xx})\right)\,. \tag{7.19}$$

(Transform (7.16) in Exercise 7.4.) For example, for Leland's model,

$$\tilde{\sigma}^2 = 1 + \gamma\,\mathrm{sign}(u_x+u_{xx})\,.$$

For all of the models of Sect. 7.1 the nonlinearity is of the type

$$\tilde{\sigma}^2(x,\tau,s)\cdot s \;\text{ with }\; s := u_x + u_{xx}\,, \tag{7.20}$$

with $\tilde{\sigma}$ from (7.19).

The payoffs Ψ of the options are transformed as well. Let u^* denote the transformed payoff. For the payoff of a vanilla put,

$$V(S,T) = K\mathrm{e}^x u(x,0) = (K-S)^+ = K(1-\mathrm{e}^x)^+$$

and hence

$$u(x,0) = u^*(x) := (\mathrm{e}^{-x}-1)^+\,.$$

Similarly, for a vanilla call,

$$u(x,0) = u^*(x) := (1 - \mathrm{e}^{-x})^+ .$$

This is similar for exotic options ($\longrightarrow$ Exercise 7.3).

Finally, boundary conditions are chosen (as in Sect. 4.4) and transformed. For example, applying (4.27) for a vanilla call of the European type,

$$\begin{aligned} u(x_{\max},\tau) &= \frac{\mathrm{e}^{-x_{\max}}}{K} V(S_{\max},t) \\ &= \frac{\mathrm{e}^{-x_{\max}}}{K} (S_{\max}\mathrm{e}^{-\delta(T-t)} - K\mathrm{e}^{-r(T-t)}) \\ &= \mathrm{e}^{-\delta(T-t)} - \exp(-r(T-t) - x_{\max}) \\ &= \exp(-\tau\frac{2\delta}{\sigma_0^2}) - \exp(-\tau\frac{2r}{\sigma_0^2} - x_{\max}) , \\ u(x_{\min},\tau) &= 0 . \end{aligned}$$

For a vanilla put and $S_{\min} \approx 0$ one may choose

$$\begin{aligned} u(x_{\min},\tau) &= \frac{1}{K}\mathrm{e}^{-x_{\min}} K\mathrm{e}^{-r(T-t)} = \exp(-\tau\frac{2r}{\sigma_0^2} - x_{\min}) , \\ u(x_{\max},\tau) &= 0 . \end{aligned}$$

For vanilla American-style options with penalty formulation (7.16), the nonzero boundary conditions are just that u is in contact with the payoff,

$$\begin{aligned} u(x_{\min}) &= u^*(x_{\min}) = \mathrm{e}^{-x_{\min}} - 1 \qquad \text{for a put, and} \\ u(x_{\max}) &= u^*(x_{\max}) = 1 - \mathrm{e}^{-x_{\max}} \qquad \text{for a call.} \end{aligned}$$

7.2.2 *Discretization*

Finite differences in a standard fashion as in Chap. 4, with the same grid, lead to nonlinear equations for the vector $w^{(\nu)}$ of approximate values at time level $\tau_\nu = \tau_{\nu-1} + \Delta\tau$. The equidistant x-spacing with mesh size Δx consists of m subintervals, see Sect. 4.2.2. As before, the components w_0 and w_m are defined by boundary conditions. The finite differences include

$$\begin{aligned} \delta_x w_{i,\nu} &:= \frac{w_{i+1,\nu} - w_{i-1,\nu}}{2\Delta x} , \\ \delta_{xx} w_{i,\nu} &:= \frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2} , \end{aligned}$$

where Δx^2 is understood as $(\Delta x)^2$. For the discretization replace s of (7.20) by $\bar{s}$ with

$$\bar{s}_{i,\nu} := (\delta_x + \delta_{xx}) w_{i,\nu} = \frac{w_{i+1,\nu} - w_{i-1,\nu}}{2\Delta x} + \frac{w_{i+1,\nu} - 2w_{i,\nu} + w_{i-1,\nu}}{\Delta x^2} .$$

Substituting into the PDEs is the next step. Here we confine ourselves to the European case (7.15); the discretization of (7.16) is analogous and left to the reader. Define

$$\begin{aligned} \mathcal{L}_{i,\nu} : = \tilde{\sigma}^2(x_i, \tau_\nu, \delta_x w_{i,\nu}, \delta_{xx} w_{i,\nu})(\delta_x w_{i,\nu} + \delta_{xx} w_{i,\nu}) \\ + \frac{2(r-\delta)}{\sigma_0^2} \delta_x w_{i,\nu} - \frac{2\delta}{\sigma_0^2} w_{i,\nu} \end{aligned}$$

to arrive at the θ-approach

$$\frac{-w_{i,\nu+1} + w_{i,\nu}}{\Delta\tau} + \theta \mathcal{L}_{i,\nu+1} + (1-\theta)\mathcal{L}_{i,\nu} = 0 . \tag{7.21}$$

Recall that this includes Crank–Nicolson for $\theta = \frac{1}{2}$, and for $\theta = 1$ the fully implicit Euler (BDF). The $\tilde{\sigma}$ of the above examples is represented by the discretization $\tilde{\sigma}(x_i, \tau_\nu, \bar{s}_{i,\nu})$ with

$$\begin{aligned} \bar{s}_{i,\nu} &= w_{i-1,\nu} \left(-\frac{1}{2\Delta x} + \frac{1}{\Delta x^2} \right) - \frac{2}{\Delta x^2} w_{i,\nu} + w_{i+1,\nu} \left(\frac{1}{2\Delta x} + \frac{1}{\Delta x^2} \right) \\ &= \alpha\, w_{i-1,\nu} - \frac{2}{\Delta x^2}\, w_{i,\nu} + \beta\, w_{i+1,\nu} , \end{aligned} \tag{7.22}$$

where we denote

$$\alpha := -\frac{1}{2\Delta x} + \frac{1}{\Delta x^2} , \quad \beta := \frac{1}{2\Delta x} + \frac{1}{\Delta x^2} , \tag{7.23}$$

and reuse the notation $\tilde{\sigma}$ for the three-argument version. Now the discretized version of the operator $\mathcal{L}_{i,\nu}$ is

$$\mathcal{L}_{i,\nu} = \tilde{\sigma}^2(x_i, \tau_\nu, \bar{s}_{i,\nu}) \bar{s}_{i,\nu} + \frac{r-\delta}{\sigma_0^2 \Delta x}(w_{i+1,\nu} - w_{i-1,\nu}) - \frac{2\delta}{\sigma_0^2} w_{i,\nu} \tag{7.24}$$

and the θ-method reads

$$-w_{i,\nu+1} + w_{i,\nu} + \theta \Delta\tau \mathcal{L}_{i,\nu+1} + (1-\theta)\Delta\tau \mathcal{L}_{i,\nu} = 0 . \tag{7.25}$$

With the vector notation $w^{(\nu)}$ as in Chap. 4 and a vector function F this is written

$$F(w^{(\nu+1)}, w^{(\nu)}) = 0 .$$

For the fully implicit BDF method ($\theta = 1$), the ith equation of the vector equation $F = 0$ reads

$$\begin{aligned} F_i = & - w_i^{(\nu+1)} + w_i^{(\nu)} \\ & + \Delta\tau \Big[\tilde{\sigma}^2(x_i, \tau_{\nu+1}, \alpha w_{i-1}^{(\nu+1)} - \frac{2}{\Delta x^2} w_i^{(\nu+1)} + \beta w_{i+1}^{(\nu+1)}) \cdot \\ & \qquad (\alpha w_{i-1}^{(\nu+1)} - \frac{2}{\Delta x^2} w_i^{(\nu+1)} + \beta w_{i+1}^{(\nu+1)}) \\ & - \frac{r-\delta}{\sigma_0^2 \Delta x} w_{i-1}^{(\nu+1)} - \frac{2\delta}{\sigma_0^2} w_i^{(\nu+1)} + \frac{r-\delta}{\sigma_0^2 \Delta x} w_{i+1}^{(\nu+1)} \Big] = 0 \,. \end{aligned} \tag{7.26}$$

For $i = 0$ and $i = m$, boundary conditions enter. Their basic structure is

$$\begin{aligned} F_0^{(\nu)} &:= u(x_{\min}, \tau_\nu) - w_0^{(\nu)} \,, \\ F_m^{(\nu)} &:= u(x_{\max}, \tau_\nu) - w_m^{(\nu)} \,. \end{aligned} \tag{7.27}$$

In the θ-method (7.25) boundary conditions enter in the form $\theta F^{(\nu+1)} + (1 - \theta)F^{(\nu)}$. The nonlinear equation $F(w^{(\nu+1)}, w^{(\nu)}) = 0$ with components defined by (7.26)/(7.27) represents a discretization of (7.15). It is solved iteratively by Newton's method.

7.2.3 *Convergence of the Discrete Equations*

The above numerical scheme is of the form

$$F(\Delta\tau, \Delta x, \nu, i, w_{i,\nu}, \tilde{w}) = 0$$

where $\tilde{w}$ stands for the vector of all $w_{k,l}$. For such a scheme convergence to the unique viscosity solution (⟶ Appendix C.5) can be proved, provided F satisfies three conditions [23], namely,

- stability,
- consistency, and
- monotonicity.

Not for the numerical scheme but for the equation an additional property must be assumed, namely, the strong uniqueness. For the uniqueness we refer to the special literature [89].

The proof that for a particular numerical scheme all of these three criteria are satisfied, can be quite involved [176, 177, 303]. Checking stability and consistency is rather standard, and has been widely discussed in previous chapters. Here we concentrate on the monotonicity of the scheme, which is a new aspect as compared to the investigations for the linear equation in Chap. 4.

Definition 7.1 (Monotone Scheme) A discretization $F(w^{(\nu+1)}, w^{(\nu)})$ is *monotone* if for all $i = 0, \ldots, m$

$$\begin{aligned} \text{(a)} \qquad & F_i(w^{(\nu+1)} + \epsilon^{(\nu+1)},\, w^{(\nu)} + \epsilon^{(\nu)}) \geq F_i(w^{(\nu+1)}, w^{(\nu)}) \ \text{ for all} \\ & \epsilon^{(\nu+1)} := (0, \ldots, 0, \epsilon_{i-1}^{(\nu+1)}, 0, \epsilon_{i+1}^{(\nu+1)}, 0, \ldots, 0) \geq 0 \ \text{ and} \\ & \epsilon^{(\nu)} := (0, \ldots, 0, \epsilon_{i-1}^{(\nu)}, \epsilon_i^{(\nu)}, \epsilon_{i+1}^{(\nu)}, 0, \ldots, 0) \geq 0\,, \end{aligned}$$

and

$$\begin{aligned} \text{(b)} \qquad & F_i(w^{(\nu+1)} + \epsilon^{(\nu+1)},\, w^{(\nu)}) \leq F_i(w^{(\nu+1)}, w^{(\nu)}) \ \text{ for all} \\ & \epsilon^{(\nu+1)} := (0, \ldots, 0, \epsilon_i^{(\nu+1)}, 0, \ldots, 0) \geq 0\,. \end{aligned}$$

Translated into the fully implicit scheme (7.26)/(7.27), the condition (a) of monotonicity reads

$$\begin{aligned} & F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)} + \epsilon_1,\, w_{i+1}^{(\nu+1)} + \epsilon_2,\, w_i^{(\nu)} + \epsilon_3) \geq \\ & F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)},\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)}) \end{aligned}$$

for scalar $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon$. Because of transitivity, it suffices to show separately

$$\begin{aligned} \text{(a1)} \quad & F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)} + \epsilon,\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)}) \geq F_i(w_i^{(\nu+1)}, w_{i-1}^{(\nu+1)}, w_{i+1}^{(\nu+1)}, w_i^{(\nu)}) \\ \text{(a2)} \quad & F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)},\, w_{i+1}^{(\nu+1)} + \epsilon,\, w_i^{(\nu)}) \geq F_i(w_i^{(\nu+1)}, w_{i-1}^{(\nu+1)}, w_{i+1}^{(\nu+1)}, w_i^{(\nu)}) \\ \text{(a3)} \quad & F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)},\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)} + \epsilon) \geq F_i(w_i^{(\nu+1)}, w_{i-1}^{(\nu+1)}, w_{i+1}^{(\nu+1)}, w_i^{(\nu)}) \end{aligned}$$

for (a) to hold, and for (b)

$$F_i(w_i^{(\nu+1)} + \epsilon,\, w_{i-1}^{(\nu+1)},\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)}) \leq F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)},\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)})\,.$$

Next we check under which conditions the scheme (7.26)/(7.27) is monotone. Heider [176] has shown that the scheme converges whenever the nonlinear term $\tilde{\sigma}^2(x, \tau, s)s$ satisfies conditions (i)–(iii) of the following Theorem 7.2:

Theorem 7.2 (Convergence) *Assume $\tilde{\sigma}^2(x, \tau, u_x, u_{xx})$ in the form $\tilde{\sigma}^2(x, \tau, s)$, with $s = u_x + u_{xx}$ from (7.20), and*

(i) $\tilde{\sigma}^2(x, \tau, s)s$ is continuous and monotone increasing in s,
(ii) there exists a constant $c_+ > 0$ such that for all s and $\epsilon > 0$

$$\tilde{\sigma}^2(x, \tau, s + \epsilon) \cdot (s + \epsilon) \geq \tilde{\sigma}^2(x, \tau, s) \cdot s + c_+ \epsilon\,, \quad \textit{and}$$

(iii) Δx is small enough such that

$$c_+\frac{2-\Delta x}{\Delta x}-\frac{2(r-\delta)}{\sigma_0^2}\geq 0 \;\; \textit{and} \;\; c_+\frac{2+\Delta x}{\Delta x}+\frac{2(r-\delta)}{\sigma_0^2}\geq 0\,.$$

Then the fully implicit BDF scheme (7.26)/(7.27) converges to the viscosity solution of (7.15).

Proof Here we confine ourselves to the proof of monotonicity. As noted above, we can proceed componentwise and check (a1), (a2), (a3), and (b) separately. We begin with $0<i<m$.

To show (a1), perturb $w_{i-1}^{(\nu+1)}\to w_{i-1}^{(\nu+1)}+\epsilon$ for $\epsilon>0$. Then $\bar{s}_{i,\nu}\to\bar{s}_{i,\nu}+\alpha\epsilon$, and

$$\begin{aligned}
&F_i(w_i^{(\nu+1)},\, w_{i-1}^{(\nu+1)}+\epsilon,\, w_{i+1}^{(\nu+1)},\, w_i^{(\nu)}) = \\
&\quad -w_i^{(\nu+1)}+w_i^{(\nu)}+\Delta\tau\Big[\tilde{\sigma}^2(x_i,\tau_{\nu+1},\bar{s}_{i,\nu}+\alpha\epsilon)(\bar{s}_{i,\nu}+\alpha\epsilon) \\
&\quad -\frac{r-\delta}{\sigma_0^2\Delta x}(w_{i-1}^{(\nu+1)}+\epsilon)-\frac{2\delta}{\sigma_0^2}w_i^{(\nu+1)}+\frac{r-\delta}{\sigma_0^2\Delta x}w_{i+1}^{(\nu+1)}\Big] \\
&\geq -w_i^{(\nu+1)}+w_i^{(\nu)}+\Delta\tau\Big[\tilde{\sigma}^2(x_i,\tau_{\nu+1},\bar{s}_{i,\nu})\bar{s}_{i,\nu}+c_+\epsilon\alpha \\
&\quad -\frac{r-\delta}{\sigma_0^2\Delta x}w_{i-1}^{(\nu+1)}-\frac{2\delta}{\sigma_0^2}w_i^{(\nu+1)}+\frac{r-\delta}{\sigma_0^2\Delta x}w_{i+1}^{(\nu+1)}-\frac{r-\delta}{\sigma_0^2\Delta x}\epsilon\Big],
\end{aligned}$$

where the inequality is due to (ii). Compare with F_i in (7.26)/(7.27) and realize two extra terms. By (iii), with α from (7.23), they are

$$c_+\epsilon\alpha-\frac{r-\delta}{\sigma_0^2\Delta x}\epsilon=\frac{\epsilon}{2\Delta x}\left[c_+\frac{2-\Delta x}{\Delta x}-\frac{2(r-\delta)}{\sigma_0^2}\right]\geq 0\,.$$

So we have shown (a1), the first of the four criteria of monotonicity.

To show (a2), perturb $w_{i+1}^{(\nu+1)}\to w_{i+1}^{(\nu+1)}+\epsilon$. Then $\bar{s}_{i,\nu}\to\bar{s}_{i,\nu}+\epsilon\beta$ and the perturbed F_i is

$$\begin{aligned}
&-w_i^{(\nu+1)}+w_i^{(\nu)}+\Delta\tau\Big[\tilde{\sigma}^2(x_i,\tau_{\nu+1},\bar{s}_{i,\nu}+\beta\epsilon)(\bar{s}_{i,\nu}+\beta\epsilon) \\
&-\frac{r-\delta}{\sigma_0^2\Delta x}w_{i-1}^{(\nu+1)}-\frac{2\delta}{\sigma_0^2}w_i^{(\nu+1)}+\frac{r-\delta}{\sigma_0^2\Delta x}w_{i+1}^{(\nu+1)}+\epsilon\frac{r-\delta}{\sigma_0^2\Delta x}\Big].
\end{aligned}$$

Again we obtain a lower bound by (ii), and arrive at the sum of two extra terms

$$c_+\epsilon\beta+\epsilon\frac{r-\delta}{\sigma_0^2\Delta x}\,,$$

which is ≥ 0 by (iii). So the perturbed F_i is larger or equal the unperturbed F_i, and (a2) is satisfied.

The assertion (a3) is clearly satisfied since the perturbation $w_i^{(\nu)} \to w_i^{(\nu)} + \epsilon$ only affects the term outside the brackets.

To show (b), perturb $w_i^{(\nu+1)} \to w_i^{(\nu+1)} + \epsilon$. Then $\bar{s}_{i,\nu} \to \bar{s}_{i,\nu} - \frac{2\epsilon}{\Delta x^2}$, and F_i is perturbed to

$$-w_i^{(\nu+1)} - \epsilon + w_i^{(\nu)} + \Delta\tau\Big[\tilde{\sigma}^2(x_i, \tau_{\nu+1}, \bar{s}_{i,\nu} - \epsilon\frac{2}{\Delta x^2})(\bar{s}_{i,\nu} - \epsilon\frac{2}{\Delta x^2})$$
$$-\frac{r-\delta}{\sigma_0^2\Delta x}w_{i-1}^{(\nu+1)} - \frac{2\delta}{\sigma_0^2}w_i^{(\nu+1)} - \frac{2\delta}{\sigma_0^2}\epsilon + \frac{r-\delta}{\sigma_0^2\Delta x}w_{i+1}^{(\nu+1)}\Big].$$

By the monotonicity (i) and by $\epsilon > 0$, $\delta \geq 0$, the above is smaller or equal to the unperturbed F_i—that is, (b) holds true.

Finally, monotonicity must be checked for F_0 and F_m. For $\theta = 1$, F_0 depends on $w_0^{(\nu+1)}$ and F_m depends on $w_m^{(\nu+1)}$. Hence only (b) needs to be checked, which is clearly satisfied.

This ends the proof that the conditions (i), (ii), (iii) imply monotonicity of the fully implicit scheme.

Example 7.3 (Leland's Model) Let us inspect whether the criteria (i), (ii), (iii) of Theorem 7.2 are satisfied for Leland's model of transaction costs. For (i) we require $|\gamma| < 1$. With some simple manipulations, one shows that (ii) is satisfied with $c_+ = 1 - \gamma$. And for (iii) to hold, the grid size Δx must be small enough ($\longrightarrow$ Exercise 7.5). Specifically, for zero dividend rate $\delta = 0$, the θ-method is

$$-w_i^{(\nu+1)} + w_i^{(\nu)} + \Delta\tau\cdot\theta\,[\tilde{\sigma}^2(\bar{s}_i^{(\nu+1)})\bar{s}_i^{(\nu+1)} + \frac{2r}{\sigma_0^2}\delta_x w_i^{(\nu+1)}]$$
$$+\Delta\tau(1-\theta)\,[\tilde{\sigma}^2(\bar{s}_i^{(\nu)})\bar{s}_i^{(\nu)} + \frac{2r}{\sigma_0^2}\delta_x w_i^{(\nu)}] = 0\,.$$

Sufficient conditions for the Crank–Nicolson scheme ($\theta = 1/2$) to converge include (i), (ii), (iii), and in addition (iv) and (v):
(iv) There exists a constant $c_- > 0$ such that for all $\epsilon > 0$ and s

$$\tilde{\sigma}^2(x, \tau, s-\epsilon)(s-\epsilon) \geq \tilde{\sigma}^2(x, \tau, s)s - c_-\epsilon\,,$$

(v)

$$\Delta\tau \leq \frac{\Delta x^2}{c_-}\,\frac{\sigma_0^2}{\sigma_0^2 + \Delta x\,\delta}\,,$$

see [176, 177]. Condition (iv) holds for Leland's model with $c_- = 1 + \gamma$, and for the uncertain-volatility model with $c_- = \sigma_{\max}^2$. Conditions (iii) and (iv) amount to

stability bounds. We emphasize that in the case of nonlinear models, unconditional stability does *not* hold!

The above has discussed convergence towards the viscosity solution. An application of the uncertain-volatility model to a butterfly is shown in Fig. 7.2. Another illustration is the barrier option in Fig. 7.3. When in case of an American-style option a penalty approach is applied, further assumptions are needed to assert convergence to the solution for $\hat{p} \to \infty$, even though one keeps $\hat{p}$ fixed.

7.3 Option Valuation Under Jump Processes

In this section, we sketch some instruments of Lévy processes as background to the application of partial integro-differential equations. The focus is on one important example, namely Merton's jump diffusion, and on strategies for a numerical valuation of options under such processes. This is no introduction to Lévy processes; for expositions on Lévy processes consult, for instance, [84, 328, 339].

For a Lévy process X_t, all increments $X_{t+\Delta t} - X_t$ are stochastically independent. Further, they are stationary, which means that all increments have the distribution of X_t. Instead of requiring continuity, Lévy processes must be "càdlàg"[2]: For all t, the process X_t is right-continuous ($X_t = X_{t+}$), and the left limit X_{t-} exists. Important examples of Lévy processes are the Wiener process (Sect. 1.6.1), and the Poisson process (Sect. 1.9).

7.3.1 Characteristic Functions

A classification of Lévy processes X_t is based on the Fourier transformation[3]

$$\phi_{X_t}(\zeta) := \mathsf{E}(\exp(\mathrm{i}\zeta X_t)) \,. \tag{7.28}$$

The function ϕ_{X_t} singles out characteristic properties of a random variable X_t. ϕ_{X_t} is called *characteristic function* of X_t, and $\psi_{X_t}(\zeta)$ [shorter: $\psi(\zeta)$] defined by $\exp(t\psi(\zeta)) = \phi_{X_t}(\zeta)$ is the *characteristic exponent*. It suffices to take $t = 1$, since the distribution of X_1 characterizes the process. The characteristic exponent $\psi(\zeta)$ satisfies the **Lévy–Khinchin representation**

$$\psi(\zeta) = \mathrm{i}\gamma\zeta - \frac{1}{2}\sigma^2\zeta^2 + \int_{-\infty}^{\infty} \left(\exp(\mathrm{i}\zeta x) - 1 - \mathrm{i}\zeta x\, \mathbf{1}_{\{|x|\le 1\}}\right) \nu(\mathrm{d}x) \,. \tag{7.29}$$

[2] French for "continu à droite avec limites à gauche".

[3] For the Fourier transform, see Sect. 7.4.

The three terms in this representation characterize different aspects of X_t. $\gamma \in \mathbb{R}$ corresponds to a deterministic trend, σ^2 to the variance of a diffusion (Brownian-motion) part of X_t, and ν is a measure on $\mathbb{R}$ characterizing the activity of jumps $\Delta X_t := X_t - X_{t-}$,

$$\nu(A) := \mathsf{E}\left[\#\{t \in [0,1] \mid \Delta X_t \neq 0,\ \Delta X_t \in A\}\right].$$

The Lévy measure $\nu(A)$ counts the (expected) number of jumps of "size" within A per unit time [84]. $\nu(A)$ is not a probability measure. For the Lévy measure ν, require $\int_{\mathbb{R}} \min(x^2, 1)\, \nu(\mathrm{d}x) < \infty$ and $\nu(\{0\}) = 0$. In the integrand of (7.29), the subtracted term $\mathrm{i}\zeta x\, \mathbf{1}_{\{|x|\leq 1\}}$ causes the integrand to be of the order $O(|x|^2)$ for $x \to 0$. This compensation along with the constraints on ν implies existence of the integral. For many important Lévy processes, $\nu(\mathrm{d}x)$ has a convenient representation

$$\nu(\mathrm{d}x) = f_{\mathrm{L}}(x)\, \mathrm{d}x \tag{7.30}$$

with a Lévy density f_{L}. The three items γ, σ^2, ν ("characteristic triplet") characterize a Lévy process in a unique way.

Example 7.4 (Compound Poisson Process) For a Poisson process J_t with jump intensity λ, a compound Poisson process is

$$X_t := \sum_{j=1}^{J_t} \Delta X_{\tau_j},$$

where the jump sizes ΔX_{τ_j} are assumed i.i.d. with distribution density f, and independent of the Poisson process J. The characteristic function $\phi_{X_t}(\zeta)$ of the compound Poisson process (cP) is

$$\begin{aligned} \mathsf{E}(\exp[\mathrm{i}\zeta X_t)) &= \exp[\lambda t\, (\phi_{\Delta X}(\zeta) - 1)] \\ &= \exp\left[t \int_{\mathbb{R}} (\mathrm{e}^{\mathrm{i}\zeta x} - 1)\nu(\mathrm{d}x)\right] \end{aligned} \tag{7.31}$$

with Lévy measure $\nu(\mathrm{d}x) = \lambda f(x)\, \mathrm{d}x$. The first of the equations in (7.31) uses rules of the conditional expectation [84], whereas the second just applies (7.28) with the definition (B.4) of the expectation, including $\int_{\mathbb{R}} \nu(\mathrm{d}x) = \lambda$. The characteristic exponent ψ_{cP} is the integral in (7.31), $\gamma = \sigma = 0$.

As in (1.65), financial models typically arise in exponential form. For such exponential Lévy processes there is a useful criterion for the martingale property, and hence for risk-neutral valuation:

Lemma 7.5 (Martingale Criterion) *Let X_t be a Lévy process. e^{X_t} is a martingale if and only if $\psi_X(-\mathrm{i}) = 0$ and $\mathsf{E}(\mathrm{e}^{X_t}) < \infty$.*

Proof We extend ζ to complex numbers, and note that

$$\mathsf{E}(\mathrm{e}^{X_t}) = \mathsf{E}(\mathrm{e}^{-\mathrm{i}\mathrm{i}X_t}) = \phi_{X_t}(-\mathrm{i}) = \mathrm{e}^{t\psi(-\mathrm{i})}\,.$$

Then by independence and stationarity,

$$\mathsf{E}(\mathrm{e}^{X_t} \mid \mathcal{F}_s) - \mathrm{e}^{X_s} = \mathsf{E}(\mathrm{e}^{X_{t-s}}) - \mathrm{e}^{X_0} = \mathrm{e}^{(t-s)\psi(-\mathrm{i})} - 1\,.$$

(⟶ Exercise 7.6) □

In finance applications, with an asset price S_t for $t \geq 0$, the absence of arbitrage implies that the discounted $\mathrm{e}^{-rt}S_t$ is a martingale with respect to a risk-neutral measure. This suggests to represent S_t in the form $S_t = S_0 \exp(rt + X_t)$. Then the discounted S_t is the situation to which the Lemma 7.5 applies.

Example 7.6 (Brownian Motion with Drift) A Lévy process X_t is Brownian motion if and only if $\nu \equiv 0$ (no jump). For ease of comparison with (1.71) and (1.76) we take the drift γ in the form $\gamma = \mu - \frac{1}{2}\sigma^2$. For the Brownian motion with drift (Bwd) $X_t := \gamma t + \sigma W_t$ we use a result from probability[4] and conclude for the characteristic exponent

$$\psi_{\mathrm{Bwd}}(\zeta) = \mathrm{i}(\mu - \frac{1}{2}\sigma^2)\zeta - \frac{1}{2}\sigma^2\zeta^2\,.$$

Clearly, $\psi_{\mathrm{Bwd}}(-\mathrm{i}) = \mu$. Hence by Lemma 7.5 e_t^X is martingale for $\mu = 0$. Hence the discounted

$$S_0\mathrm{e}^{-rt}\exp(rt + X_t) = S_0\mathrm{e}^{-rt}\exp[(r - \tfrac{1}{2}\sigma^2)t + \sigma W_t]$$

is martingale. This recovers the well-known riskless drift rate r for a numerical simulation of GBM in the Black-Scholes model.

Example 7.7 (Merton' s Jump Diffusion) We now combine Examples 7.4 and 7.6. As a special case of Example 7.4 we choose as in Sect. 1.9 the jump sizes ΔY in the log process $Y_t := \log S_t$ to be normally distributed, $\Delta Y \sim \mathcal{N}(\mu_{\mathrm{J}}, \sigma_{\mathrm{J}}^2)$ ($\log q$ in Sect. 1.9). Furnished with a drifted Brownian motion, this is Merton's jump-diffusion model (1.74) with jump intensity λ and $\gamma = \mu - \frac{1}{2}\sigma^2$. The Lévy density of the compound Poisson process is λ times the density of the normal distribution,

$$f_{\mathrm{L}}(x) = f_{\mathrm{cP}}(x) := \lambda\,\frac{1}{\sigma_{\mathrm{J}}\sqrt{2\pi}}\,\exp\left[-\frac{(x-\mu_{\mathrm{J}})^2}{2\sigma_{\mathrm{J}}^2}\right]. \tag{7.32}$$

[4] $\mathsf{E}(\mathrm{e}^{\mathrm{i}\zeta X}) = \exp(\mathrm{i}\zeta\gamma - \zeta^2\sigma^2/2)$ holds for $X \sim \mathcal{N}(\gamma, \sigma^2)$, see [199, p. 108].

Since the two processes are independent, and by the exponential structure in (7.28), the two characteristic exponents add:

$$\begin{aligned}\psi(\zeta) &= \psi_{\mathrm{Bwd}}(\zeta) + \psi_{\mathrm{cP}}(\zeta) \\ &= \mathrm{i}\gamma\zeta - \frac{1}{2}\sigma^2\zeta^2 + \int_{\mathbb{R}} (\mathrm{e}^{\mathrm{i}\zeta x} - 1)\nu(\mathrm{d}x)\end{aligned}$$

and

$$\psi(-\mathrm{i}) = \gamma + \frac{1}{2}\sigma^2 + \int_{\mathbb{R}} (\mathrm{e}^{x} - 1)\nu(\mathrm{d}x)\,.$$

Similar as in Exercise 1.22 we calculate the integral

$$\int_{-\infty}^{\infty} (\mathrm{e}^{x} - 1) f_{\mathrm{cP}}(x)\,\mathrm{d}x = \lambda\left(\exp\left[\mathrm{i}\mu_{\mathrm{J}}\zeta - \frac{1}{2}\sigma_{\mathrm{J}}^2\zeta^2\right] - 1\right)\,.$$

Hence, to see whether $S_t = \exp(Y_t)$ is a martingale, check $\psi(-\mathrm{i}) = \gamma + \frac{1}{2}\sigma^2 + \lambda(\exp[\mu_{\mathrm{J}} + \frac{1}{2}\sigma_{\mathrm{J}}^2] - 1)$. By Lemma 7.5, a martingale can be obtained by choosing a drift with

$$\gamma = -\frac{\sigma^2}{2} - \lambda\left(\exp\left[\mu_{\mathrm{J}} + \frac{1}{2}\sigma_{\mathrm{J}}^2\right] - 1\right)\,.$$

This makes $S_0\mathrm{e}^{-rt}\exp(rt + \gamma t + \sigma W_t + \sum_{j=1}^{J_t}\log q_j)$ a martingale. When applied to simulation of SDEs under the risk-neutral measure for Monte Carlo, this risk-neutral valuation amounts to the drift rate in Example 1.21. That is, the SDE is

$$\frac{\mathrm{d}S}{S} = (r - \lambda(\exp[\mu_{\mathrm{J}} + \tfrac{1}{2}\sigma_{\mathrm{J}}^2] - 1))\,\mathrm{d}t + \sigma\,\mathrm{d}W_t\,.$$

In case of a dividend yield with rate δ, the term $\delta\mathrm{d}t$ is subtracted on the right-hand side, similar as in Sect. 3.5.

For other models, a risk-neutral growth rate can be obtained in an analogous way. A table of risk-neutral drift rates is given in [332, p. 80]. For a jump diffusion, jumps are comparably "rare," there is only a finite number of them in any time interval. Apart from Merton's model another jump-diffusion model is Kou's model, which works with an asymmetric double exponential distribution of jump sizes [229].

There are Lévy processes of infinite activity: Then in every time interval an infinite number of jumps occurs. Examples include the VG-process (Variance Gamma) [253], the NIG-process (Normal Inverse Gaussian), the hyperbolic process [114] and the CGMY process [67]. Specifically for VG and NIG, see also [155]. Time deformation plays an important role for constructing Lévy processes. For example, with a Wiener process W_t and a Gamma process G_t as *subordinator*

replacing time, VG can be represented as

$$S_t = S_0 \mathrm{e}^{rt+X_t} \text{ with } X_t = \theta G_t + \sigma W_{G_t} .$$

This includes GBM with the standard time $G_t = t$ and parameter $\theta = -\sigma^2/2$. Such a subordinating process G_t can be regarded as "business time," which runs faster than the calendar time when the trading volume is high, and slower otherwise. Then, for a Wiener process W_t, a class of Lévy processes is defined by W_{G_t}. With a t-grid as in Algorithm 1.8, a time-changed process can be generated as $W_j = W_{j-1} + Z\sqrt{G_{j\Delta t} - G_{(j-1)\Delta t}}$ ($\longrightarrow$ Exercise 2.11).

7.3.2 Option Valuation with PIDEs

Assume European options based on a price process $S_t = S_0 \exp(rt + X_t)$, where X_t is a Lévy process such that e^{X_t} is a martingale, with Lévy measure ν, and the integral $\int_{|y|\geq 1} \mathrm{e}^{2y}\nu(\mathrm{d}y)$ exists. Then the value function $V(S,t)$ satisfies

$$\begin{aligned} &\frac{\partial V(S,t)}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV \\ &+ \int_{\mathbb{R}} \left[V(S\mathrm{e}^y, t) - V(S,t) - (\mathrm{e}^y - 1)S\frac{\partial V(S,t)}{\partial S} \right] \nu(\mathrm{d}y) = 0 \end{aligned} \tag{7.33}$$

A proof can be found in [84, pp. 385–387].

Definition 7.8 (PIDE) An equation of the above type (7.33) is called *partial integro-differential equation (PIDE)*.

The integral term in (7.33) complicates the numerical solution since it is a nonlocal term accumulating information on all $-\infty < y < \infty$, in contrast to the local character of the partial derivatives. For general Lévy processes, the three terms under the integral can not be separated, otherwise the integral may fail to converge. It can be separated in the case of Merton's jump-diffusion model, because this process is of finite activity, $\lambda = \nu(\mathbb{R}) < \infty$.

In what follows, we discuss Merton's jump-diffusion process, with lognormal distribution for $q = \mathrm{e}^y$. The integral in (7.33) can be split into three terms with three integrals

$$\int_{\mathbb{R}} V(S\mathrm{e}^y, t)\nu(\mathrm{d}y) - V(S,t)\int_{\mathbb{R}} \nu(\mathrm{d}y) - S\frac{\partial V(S,t)}{\partial S}\int_{\mathbb{R}} (\mathrm{e}^y - 1)\nu(\mathrm{d}y) .$$

In view of $\nu(\mathrm{d}y) = \lambda f(y)\mathrm{d}y$, factors λ show up. f is the normal density, and the integrals become expectations. Then the first integral can be written $\lambda \mathsf{E}(V(S\mathrm{e}^y, t))$, and the second integral is λ. The third integral $\mathsf{E}(\mathrm{e}^y - 1)$ does not depend on V and

can be calculated beforehand since the distribution for $q = \mathrm{e}^y$ is stipulated.[5] The lognormal density for q is

$$f_q(x) = \frac{1}{\sqrt{2\pi}\,\sigma_J \cdot x} \exp\left\{ -\frac{(\log x - \mu_J)^2}{2\sigma_J^2} \right\} \mathbf{1}_{\{x>0\}}$$

and we recover the constant of Example 7.7:

$$\begin{aligned} c :&= \int_0^\infty (x-1) f_q(x)\,\mathrm{d}x \\ &= \int_{-\infty}^\infty (\mathrm{e}^y - 1) f(y)\,\mathrm{d}y = \exp\left[\mu_J + \frac{1}{2}\sigma_J^2\right] - 1\,. \end{aligned}$$

With the precalculated number c, the resulting Eq. (7.33) can be ordered into

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (r - \lambda c) S \frac{\partial V}{\partial S} - (\lambda + r) V + \lambda \mathsf{E}(V(qS, t)) = 0\,. \tag{7.34}$$

The last term is an integral taken over the unknown solution function $V(S, t)$. So the resulting equation is a PIDE, a special case of (7.33). Note that the product λc is the drift compensation in Example 7.7. The standard Black–Scholes PDE (7.1) is included for $\lambda = 0$. A simplified derivation of (7.34) can be found in Appendix A.4. For further discussions, see for example [84, 270, 365, 375].

7.3.3 Transformation of the PIDE

We approach the PIDE (7.34) with the transformation

$$\tau := T - t\,, \quad x := \log S\,, \quad u(x, \tau) := V(\mathrm{e}^x, T - \tau)\,, \tag{7.35}$$

which appears moderate as compared to (4.3). Substituting accordingly

$$u_x = \frac{\partial V}{\partial S} S\,, \quad u_{xx} = u_x + S^2 \frac{\partial^2 V}{\partial S^2}$$

into (7.34) leads to

$$-u_\tau + \tfrac{1}{2}\sigma^2 (u_{xx} - u_x) + (r - \lambda c) u_x - (\lambda + r) u + \lambda \mathsf{E}(V(q\mathrm{e}^x, T - \tau)) = 0\,,$$

[5] The parameters are not the same as those in (1.64).

which is organized into

$$u_\tau - \tfrac{1}{2}\sigma^2 u_{xx} - (r - \lambda c - \tfrac{1}{2}\sigma^2)u_x + (\lambda + \tau)u - \lambda \mathsf{E}(V(q\mathrm{e}^x, T - \tau)) = 0\,.$$

After the above transformation $S = \mathrm{e}^x$ we next transform the jump-size variable $q = \mathrm{e}^y$. Ignoring the factor λ, the integral term changes to

$$\begin{aligned} &\mathsf{E}(V(q\mathrm{e}^x, T - \tau)) = \mathsf{E}(V(\mathrm{e}^{x+y}, T - \tau)) = \mathsf{E}(u(x + y, \tau)) \\ &= \int_{\mathbb{R}} u(x + y, \tau) f(y)\,\mathrm{d}y = \int_{\mathbb{R}} u(z, \tau) f(z - x)\,\mathrm{d}z\,, \end{aligned} \tag{7.36}$$

where we have applied the substitution $z := x + y$. The function f for Merton's jump-diffusion model is the density of $y = \log q \sim \mathcal{N}(\mu_\mathrm{J}, \sigma_\mathrm{J}^2)$. In summary, the PIDE of Merton's jump-diffusion model is

Problem 7.9 (Merton's Jump-Diffusion PIDE)

$$\begin{aligned} &u_\tau - \tfrac{1}{2}\sigma^2 u_{xx} - (r - \lambda c - \tfrac{1}{2}\sigma^2)u_x + (\lambda + r)u \\ &\quad - \lambda \int_{\mathbb{R}} u(z, \tau) f(z - x)\,\mathrm{d}z = 0\,, \\ &\quad \textit{with}\ \ f(y) = \frac{1}{\sqrt{2\pi}\sigma_\mathrm{J}} \exp\left[-\frac{(y - \mu_\mathrm{J})^2}{2\sigma_\mathrm{J}^2}\right] \\ &\quad \textit{and}\ \ \ c = \exp[\mu_\mathrm{J} + \tfrac{1}{2}\sigma_\mathrm{J}^2] - 1\,. \end{aligned} \tag{7.37}$$

This is the problem to be solved numerically.

7.3.4 Numerical Approximation

For an approximation of the integral (7.36) we truncate the domain to a finite interval $x_{\min} \le x \le x_{\max}$. In view of the meaning of the integral, this truncation amounts to disregard large jumps. This might be seen as a weakness of the approach, but jumps that large are highly improbable. The simplest discretization approach is to use an equidistant x-grid with

$$\Delta x := \frac{x_{\max} - x_{\min}}{m}\,, \quad x_i := x_{\min} + i\Delta x\,, \quad i = 0, \ldots, m\,,$$

for a suitable integer m. As in Chap. 4, the time-stepping nodes are τ_ν, and the approximations of $u(x_i, \tau_\nu)$ are denoted by $w_{i,\nu}$. The integral in (7.37) is evaluated at each node $(x, \tau) = (x_i, \tau_\nu)$. That is, for each i, ν, the numbers

$$\int_{\mathbb{R}} u(z, \tau_\nu) f(z - x_i)\,\mathrm{d}z \approx \int_{x_{\min}}^{x_{\max}} u(z, \tau_\nu) f(z - x_i)\,\mathrm{d}z$$

are to be approximated. Applying the composite trapezoidal sum (C.2) with

$$f_{i,l} := f(x_l - x_i) = f((l-i)\Delta x)\,,$$

the approximation of the integral for each i, ν is

$$\Delta x\left[\frac{w_{0,\nu} f_{i,0}}{2} + \sum_{l=1}^{m-1} w_{l,\nu} f_{i,l} + \frac{w_{m,\nu} f_{i,m}}{2}\right]. \tag{7.38}$$

The numbers $f_{i,l}$ are elements of a Toeplitz matrix.[6] That is, the entries take only $2m+1$ different numbers. Due to the exponential structure of f, the elements in the northeast and southwest corners of the $f_{i,l}$-matrix go to zero. In this sense, this Toeplitz matrix has a "banded" structure. In summary, for each i, ν the integral is approximated by a scalar product of the row vector

$$\Delta x\left(\frac{f_{i,0}}{2}, f_{i,1}, \ldots, f_{i,m-1}, \frac{f_{i,m}}{2}\right)$$

times the vector $w^{(\nu)}$. In (7.38) the first term $w_{0,\nu}$ and the last term $w_{m,\nu}$ (where boundary conditions enter) must be treated separately in case we deal with the short vector $(w_1, \ldots, w_{m-1})$ as in Sect. 4.2.3. Now assemble all the rows into an $(m+1)^2$-matrix C. Then for all i within time level ν, the integrals are represented by the product

$$Cw^{(\nu)}\,.$$

Neglecting the fact that many of its elements are close to zero, the matrix C is dense, which reflects the nonlocal character of the integral. This is in contrast to the local character of standard finite differences with its tridiagonal matrices. The transformation (7.35) is different from (4.3), but tridiagonal matrices can be derived from (7.37) in a similar way as done in Chap. 4. The dense matrix C adds to the tridiagonal matrices, which makes the solution of linear systems with full matrices in each time step $\nu \to \nu+1$ more expensive. In an attempt to save costs, *splitting* has been suggested. This means to evaluate the integral at the previous line (ν). In this way, the multiplication Cw only shows up in the right-hand side of the known terms. The tridiagonality of the left-hand side matrices is maintained, and the method still converges. Up to boundary conditions, this splitting can be represented by an Euler-type implicit scheme

$$\frac{w^{(\nu+1)} - w^{(\nu)}}{\Delta\tau} = Gw^{(\nu+1)} + \lambda Cw^{(\nu)}\,, \tag{7.39}$$

[6] The entries of a Toeplitz matrix are constant along each diagonal.

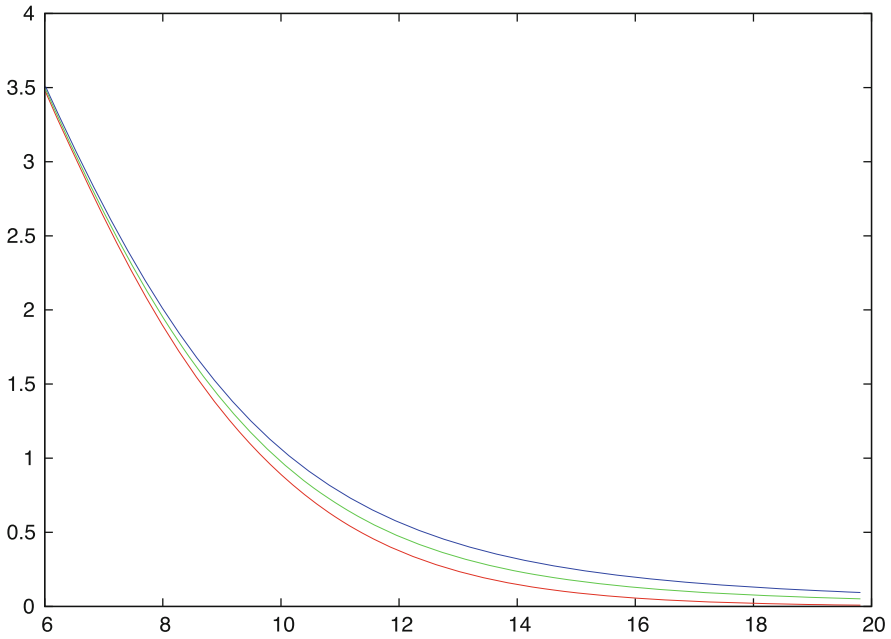

Fig. 7.4 $V(S,0)$ of a European put option, solution of Problem 7.9; parameters as in Example 1.21: $K = 10$, $r = 0.06$, $\sigma = 0.3$, $T = 1$, with Merton's jump diffusion, $\mu_J = -0.3$, $\sigma_J = 0.4$, and three values of jump intensity λ: 0 (*lower curve* in *red*, no jump), 0.1 (*green curve*), and 0.2 (*top curve*, in *blue*); $x_{\min} = -3$, $x_{\max} = \log(K) + 1.6 = 3.9$. The chosen value of $\mu_J = -0.3$ corresponds to $q = \exp(\mu_J) = 0.74$, or a 26% fall in the asset price

where the matrix G represents the local information of the differentials. Neither G nor C are symmetric. We leave it to the reader to set up the system of equations ($\longrightarrow$ Exercise 7.7).[7] The matrices G and C are used for the analysis, no matrix is needed for the algorithm. For an illustration how a larger intensity λ increases the value of an option see Fig. 7.4.

Since the splitting can deteriorate the accuracy, a fixed point iteration has been suggested [105]. The integral term $\mathsf{E}(V)$ with its truncation and discretization challenges the control of the involved errors. For example, [85] gives an estimate of the error induced by truncating the integral, as well as a convergence proof for finite differences applied to general Lévy models. Codes for American options based on a penalty formulation or on an LCP formulation can be easily modified and extended by an integral term. The techniques of Chap. 4 or Chap. 5 can be applied. Application of FFT increases the efficiency [105]. Typically, each Lévy process calls

[7] The number of arithmetic operations can be cut down by neglecting elements close to zero. To this end, in (7.39) simply replace the matrix C by a banded matrix $\overline{C}$, whose elements c_{il} are those of C except outside a band defined by $-B_L \le l - i \le B_R$ for suitably chosen positive integers $B_L, B_R < m$, where the elements are set to zero.

for a separate algorithm. A Monte Carlo approach is [272]. For Merton's model and European options, an analytic solution is given [270], which allows to test corresponding algorithms.

7.4 Application of the Fourier Transform

The Fourier transform $\mathcal{F}$ of a real function f is defined by[8]

$$\mathcal{F}[f(u)] := \int_{y=-\infty}^{\infty} \mathrm{e}^{\mathrm{i}uy} f(y)\,\mathrm{d}y\,. \tag{7.40}$$

This requires integrability of f. The inverse Fourier transformation is

$$\mathcal{F}^{-1}[g(x)] = \frac{1}{2\pi}\int_{u=-\infty}^{\infty} \mathrm{e}^{-\mathrm{i}xu} g(u)\,\mathrm{d}u\,. \tag{7.41}$$

A sufficiently well-behaved f is recovered by the inversion,

$$f = \mathcal{F}^{-1}\mathcal{F}f\,.$$

We perform this process of transform and inverse transform for a function $c(k)$ to be defined below. The application of the Fourier transform in our context and the outline of three steps of the subsequent analysis is symbolized as follows:

$$\begin{array}{ccc} & \overset{(1)}{\longrightarrow} & \\ c(k) & \circ \quad \bullet & g(u) = \text{integral} \\ & & \downarrow (2) \\ c(k) & \circ \quad \bullet & g(u) = \text{formula} \\ & \underset{(3)}{\longleftarrow} & \end{array}$$

Step (1) is the forward Fourier transform (7.40) of a function $c(k)$. The result is an integral expression $g(u)$. In our context this integral can be solved analytically (step (2)), which produces a formula for $g(u)$. The inverse transformation (7.41) in step (3) is approximated numerically by the Fast Fourier Transformation (FFT), based on (C.7). The detour (1)–(3) is worth the effort, because the FFT calculation of $c(k)$ is faster to evaluate than the original $c(k)$.

[8] There are different conventions for the Fourier transform; for background, see special literature, for example [371]. To get used to it try Exercise 7.8.

Recall the characteristic function (7.28) ϕ of a Lévy process X_t. These functions are the Fourier transform of the density function of X,

$$\phi_{X_t}(u) := \mathsf{E}(\exp(\mathrm{i}uX_t)) = \int_{-\infty}^{\infty} \mathrm{e}^{\mathrm{i}uX} f_{\text{density}X}\,\mathrm{d}x = \mathcal{F}[f_{\text{density}X}]\,. \tag{7.42}$$

The characteristic functions ϕ of many processes X are known and available as analytical expressions, for example, in [84, 235, 332].

In the following, we investigate a European call with vanilla payoff $\Psi(S) = (S-K)^+$ with an arbitrary underlying Lévy process S_t. The integral representation of the call's value under the risk-neutral measure Q is

$$\begin{aligned} V(S_t,t;\,K) &= \mathrm{e}^{-r(T-t)}\mathsf{E}_{\mathsf{Q}}[\Psi(S_T)\,|\,S_t] \\ &= \mathrm{e}^{-r(T-t)}\int_{S_T=K}^{\infty}(S_T-K)f_{\text{density}}(S_T)\,\mathrm{d}S_T\,, \end{aligned}$$

where f is the density of S_T of the Lévy process starting at t with the value S_t. Transform

$$S_T = \mathrm{e}^s,\ K = \mathrm{e}^k,\ \mathrm{d}S_T = \mathrm{e}^s\mathrm{d}s\,; \tag{7.43}$$

note that $k \in \mathbb{R}$. Then

$$V(S_t,t;\,K) = \mathrm{e}^{-r(T-t)}\int_k^{\infty}(\mathrm{e}^s-\mathrm{e}^k)\hat{f}(s)\,\mathrm{d}s\,,$$

where $\hat{f}(s) = \mathrm{e}^s f(\mathrm{e}^s)$ is the density of $\log S$, similar as in Sect. 1.8.2. Following [68], in order to make the function integrable, we scale the integral with a factor $\exp(\alpha k)$ (a constant):

$$c(k) := \mathrm{e}^{\alpha k}\mathrm{e}^{-r(T-t)}\int_k^{\infty}(\mathrm{e}^s-\mathrm{e}^k)\hat{f}(s)\,\mathrm{d}s = \mathrm{e}^{\alpha k}V(S_t,t;\,K) \tag{7.44}$$

and denote $\mathcal{F}[c(u)]$ its Fourier transform. We leave the choice of the scaling parameter α open until later.

As outlined above, when $\mathcal{F}[c]$ is calculated, then the call's value $V(S,t)$ is recovered from the inverse Fourier transformation,

$$V(S_t,t;\mathrm{e}^k) = \left(\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{e}^{-\mathrm{i}ux}\mathcal{F}[c(u)]\,\mathrm{d}u\right)\cdot\mathrm{e}^{-\alpha k}\,,$$

which can be approximated efficiently by the Fast Fourier Transform (FFT). This outlines the program of the three steps (1), (2), (3), and now we turn to its realization.

The Fourier transform of $c(k)$ is

$$\begin{aligned}\mathcal{F}[c(u)] &= \int_{k=-\infty}^{\infty} \mathrm{e}^{\mathrm{i}uk} c(k)\,\mathrm{d}k \\ &= \int_{-\infty}^{\infty} \mathrm{e}^{\mathrm{i}uk}\mathrm{e}^{\alpha k}\mathrm{e}^{-r(T-t)} \int_{s=k}^{\infty} (\mathrm{e}^{s} - \mathrm{e}^{k})\hat{f}(s)\,\mathrm{d}s\,\mathrm{d}k \\ &= \mathrm{e}^{-r(T-t)} \int_{k=-\infty}^{\infty} \int_{s=k}^{\infty} \mathrm{e}^{(\mathrm{i}u+\alpha)k}(\mathrm{e}^{s} - \mathrm{e}^{k})\hat{f}(s)\,\mathrm{d}s\,\mathrm{d}k \\ &= \mathrm{e}^{-r(T-t)} \int_{s=-\infty}^{\infty} \int_{k=-\infty}^{s} \mathrm{e}^{(\mathrm{i}u+\alpha)k}(\mathrm{e}^{s} - \mathrm{e}^{k})\hat{f}(s)\,\mathrm{d}k\,\mathrm{d}s\,,\end{aligned}$$

where the last equation holds since

$$\{\, k \le s < \infty \mid -\infty < k < \infty \,\} = \{\, -\infty < k \le s \mid -\infty < s < \infty \,\}\,.$$

This leads to

$$\begin{aligned}\mathcal{F}[c(u)] &= \mathrm{e}^{-r(T-t)} \int_{-\infty}^{\infty} \hat{f}(s) \int_{-\infty}^{s} [\mathrm{e}^{(\mathrm{i}u+\alpha)k+s} - \mathrm{e}^{(\mathrm{i}u+\alpha+1)k}]\,\mathrm{d}k\,\mathrm{d}s \\ &= \mathrm{e}^{-r(T-t)} \int_{-\infty}^{\infty} \hat{f}(s) \left[\frac{\mathrm{e}^{s}\mathrm{e}^{(\mathrm{i}u+\alpha)k}}{\mathrm{i}u+\alpha} - \frac{\mathrm{e}^{(\mathrm{i}u+\alpha+1)k}}{\mathrm{i}u+\alpha+1} \right]_{k=-\infty}^{s} \mathrm{d}s\,.\end{aligned} \tag{7.45}$$

To have the integral exist, we require the factor $\mathrm{e}^{\alpha k}$ to vanish for $k \to -\infty$, which leads to choose $\alpha > 0$. That is, the factor $\exp(\alpha k)$ amounts to a damping of the integral. The bracketed term in (7.45) is

$$\frac{(\mathrm{i}u+\alpha+1)\mathrm{e}^{s(\mathrm{i}u+\alpha+1)} - (\mathrm{i}u+\alpha)\mathrm{e}^{s(\mathrm{i}u+\alpha+1)}}{\mathrm{i}u(2\alpha+1) + \alpha(\alpha+1) - u^2}\,,$$

and we come up with

$$\mathcal{F}[c(u)] = \frac{\mathrm{e}^{-r(T-t)}}{\mathrm{i}u(2\alpha+1) + \alpha(\alpha+1) - u^2} \int_{-\infty}^{\infty} \hat{f}(s)\mathrm{e}^{\mathrm{i}s(u-(\alpha+1)\mathrm{i})}\,\mathrm{d}s\,.$$

We denote the integral therein $\phi(u-(\alpha+1)\mathrm{i})$, because it is the characteristic function of the density $\hat{f}$. For ϕ an analytic expression is known. Hence

$$\mathcal{F}[c(u)] = \frac{\mathrm{e}^{-r(T-t)}\,\phi(u-(\alpha+1)\mathrm{i})}{\alpha^2+\alpha-u^2+\mathrm{i}u(2\alpha+1)} =: g(u) \tag{7.46}$$

can be considered to be a known function g, and step (2) is completed. For the final choice of the parameter $\alpha > 0$ further request $g(u) = \mathcal{F}[c(u)]$ to be integrable as well. Since the integration is along real values of u one has to take care that the

denominator has only imaginary roots in u. The choice of α is discussed in the literature [68, 235]. Usually $\alpha = 3$ works well.

The inverse Fourier transformation evaluates

$$\mathrm{e}^{-\alpha k}\frac{1}{2\pi}\int_{-\infty}^{\infty}\mathrm{e}^{-\mathrm{i}ku}g(u)\,\mathrm{d}u\,.$$

The integral is real, and hence its integrand is real too. Think of g from (7.46) being split into real part and imaginary part, $g(u) = g_1(u) + \mathrm{i}g_2(u)$. Then $\mathrm{i}(\cos(ku)g_2(u) - \sin(ku)g_1(u)) = 0$, and we conclude that $g_1(u)$ is an even function, and $g_2(u)$ is an odd function. Hence the integrand

$$\cos(ku)g_1(u) + \sin(ku)g_2(u)$$

is even, and the value of the call is

$$V(S_t, t; \mathrm{e}^k) = \frac{\mathrm{e}^{-\alpha k}}{\pi}\int_0^{\infty}\mathrm{e}^{-\mathrm{i}ku}g(u)\,\mathrm{d}u\,. \tag{7.47}$$

Next, the semi-infinite integration interval is truncated to a finite length A. Thereby, for most Lévy models the truncation error can be made arbitrarily small because the characteristic function ϕ decays exponentially fast at infinity.[9] With the restriction to the integration interval $0 \le u \le A$ and $M-1$ subintervals with equal length Δu, the discrete grid points are

$$u_j := j\Delta u = j\frac{A}{M-1}\,, \quad j = 0, \ldots, M-1\,.$$

Choosing the trapezoidal sum (C.2) for the quadrature, the approximation is

$$\int_0^{\infty}\mathrm{e}^{-\mathrm{i}ku}g(u)\,\mathrm{d}u \;\approx\; \frac{A}{M-1}\sum_{j=0}^{M-1}\beta_j\, g(u_j)\,\mathrm{e}^{-\mathrm{i}ku_j} \tag{7.48}$$

with weights $\beta_0 = \beta_{M-1} = \frac{1}{2}$ and $\beta_j = 1$ for $1 \le j \le M-2$. The trapezoidal sum goes along with a sampling error of the order $O(\Delta u^2)$.

So far, the log-strike $k = \log K$ is not specified. The aim is to exploit the potential of FFT, which calculates sums of the type

$$\sum_{j=0}^{M-1} a_j\,\mathrm{e}^{-\mathrm{i}\nu j\frac{2\pi}{M}} \tag{7.49}$$

[9] This does not hold for the VG process, see [84, 235].

for complex numbers $a_0, \ldots, a_{M-1}$, one sum for each ν. This amounts to calculate a *vector* of M such sums, for $\nu = 0, \ldots, M-1$. Applying FFT we gain the possibility to calculate the above for M strikes simultaneously. Let us calculate the call values for the log-strike values

$$k_\nu := -b + \Delta k \cdot \nu\,, \quad \nu = 0, \ldots, M-1\,, \tag{7.50}$$

for suitable values of b and Δk, which define the k-range and the strike spacing of interest. Substituting these values k_ν into the above sum (7.48) produces

$$\frac{A}{M-1} \sum_{j=0}^{M-1} \beta_j\, g(u_j) \exp\left[-\mathrm{i}(-b + \Delta k\, \nu) j \frac{A}{M-1}\right].$$

The argument of the exponential function is

$$\mathrm{i}bj\frac{A}{M-1} - \mathrm{i}\nu j \Delta k \frac{A}{M-1}\,.$$

To apply FFT aiming at (7.49), steps Δk and $\Delta u = \frac{A}{M-1}$ must be chosen such that

$$\Delta k \frac{A}{M-1} = \Delta k\, \Delta u = \frac{2\pi}{M}\,. \tag{7.51}$$

Then the sum in (7.48) is

$$\frac{A}{M-1} \sum_{j=0}^{M-1} \beta_j g(u_j) \exp\left[\mathrm{i}bj\frac{A}{M-1}\right] \mathrm{e}^{-\mathrm{i}\nu j \frac{2\pi}{M}}\,,$$

which is the standard FFT applied to (7.49) for the complex numbers

$$a_j := A\beta_j g(u_j) \exp\left[\mathrm{i}bj\frac{A}{M-1}\right], \quad i = 0, \ldots, M-1\,. \tag{7.52}$$

This completes the calculation of a bunch of European call values: The integral in (7.47) is approximated by the FFT sum (7.49) with coefficients (7.52). For the highly efficient calculation of the FFT sums (7.49) consult standard literature on numerical analysis (such as [306]), and related software packages.

The above method amounts to a fast algorithm in case option prices are to be calculated on a grid of many strikes, all options with the same maturity T. The log-strike grid of the values k_ν is defined by (7.50) with the parameters b and Δk, which in turn are based on A, M. By (7.51),

$$\Delta k = \frac{2\pi}{A}\frac{M-1}{M}\,.$$

And to cover log strikes in the at-the-moment range around $k = 0$, one aims at

$$b = \frac{(M-1)\Delta k}{2}\,.$$

Efficiency of FFT is maximal for M a power of 2. The Eq. (7.51) is a limitation that requests a careful design of parameters M and A.

In this section, we have explained the basic FFT approach of Carr and Madan [68]. The Fast Fourier Transform can be applied also for early-exercise options [248]. A novel transform is based on Fourier-cosine expansions [125], which is also applied to barrier options [126]. The resulting algorithms converge exponentially fast. In summary, FFT-based methods have shown a rich potential, in particular for option pricing under Lévy models.

7.5 Notes and Comments

On Sect. 7.1

For a critical account of Leland's approach see [380]. The nonlinear version (7.6)–(7.8) is due to [187]. A piecewise linear treatment is suggested in [77]. The paper [18] discusses Eq. (7.7), suggesting a modification for the case $\gamma \geq 1$, where $\hat{\sigma}^2$ would be negative for $\Gamma < 0$. For bounds on V in case of "misspecified" volatility, see [118]. For related work, consult also [116, 156, 159].

Apart from the one-factor case, ranges for parameters play a role also in multiasset cases. For example, consider two assets with prices S_1, S_2, and assume a correlation in the range $-1 \leq \rho_{\min} \leq \rho \leq \rho_{\max} \leq 1$. In the Black–Scholes equation (6.5), the term

$$\rho\sigma_1\sigma_2 S_1 S_2 \frac{\partial^2 V}{\partial S_1 \partial S_2}$$

occurs. Depending on the sign of the cross derivative $\frac{\partial^2 V}{\partial S_1 \partial S_2}$, ρ is chosen either as $\rho_{\min}$ or $\rho_{\max}$ in order to characterize a "worst-case," see [362].

To complete the introduction into more general models we outline the Dupire equation in Appendix A.6.

On Sect. 7.2

For reference and examples consult [134, 176, 177]. The assumption of a constant c_+ in Theorem 7.2 is not always satisfied easily. For example, in the Barles and Soner model of Sect. 7.1.2 and a payoff with jump discontinuity (as digital option), $c_+ = c_+(\Delta x) = O(\Delta x^2)$, which affects the assumptions of Theorem 7.2, and has

strong implications on stability. Apart from nonsmooth payoffs, also the PDE itself typically is not smooth. For American options, the penalty term in (7.16) causes a lack of smoothness. Also the volatility function $\tilde{\sigma}$ may be nonsmooth. This happens, for example, in Leland's model when V_{SS} changes sign. Newton's method then works with a generalized derivative. The higher the degree of "non-smoothness," the worse the convergence rate of CN. The BDF method (7.26)/(7.27) is highly recommended. An a priori check of convergence criteria is advisable.

On Sect. 7.3

The definition of Lévy processes includes stochastic continuity. A table of Lévy densities f_L is found in [332, p. 154]. The Lévy-Khinchin representation (7.29) is a scalar setting; [69] develops analytic expressions for the characteristic function of time-changed Lévy process in a general vector setting. In this framework, Heston's stochastic-volatility model can be represented as time-changed Brownian motion.

For time-changed Lévy processes, consult [11, 67, 69, 84]. Time-changed Lévy processes have been successfully applied to match empirical data. For processes with density function (Merton, VG, NIG), Algorithm 1.18 can be applied [309]. Lévy-process models have been extended by incorporating stochastic volatilities [67, 212]. A subordinator $\tau(t)$ can be constructed as integral of a square-root process.

Pham [299] investigates properties of American options. Heston presents the characteristic function for his model in [178]. His model extended by jump diffusion [30] can be cast into the above framework: In this case a two-dimensional PDE is considered. For computational approaches see [6–8, 55, 85, 104, 105, 263].

On Sect. 7.4

Choosing the weights w_j of Simpson's sums instead of trapezoidal sums, the integrations get more accurate. An application to VG is found in [68]. Modifications and extensions of the above basic approach are described and reviewed in [235]. For references on transform methods in option pricing, see [126].

7.6 Exercises

7.1 Let ΔW be the increment of a Wiener process, see Sect. 1.6.1. Show

$$\mathsf{E}(|\Delta W|) = \sqrt{\Delta t}\,\sqrt{\frac{2}{\pi}}\,.$$

7.2 (Barles–Soner Model)
The differential equation of Barles and Soner is

$$\frac{\mathrm{d}f(x)}{\mathrm{d}x} = \frac{f(x)+1}{2\sqrt{xf(x)}-x} \quad \text{with } f(0) = 0\,.$$

(a) By numerical computations, analyze the solution for $-2 \le x \le 2$.
(b) Construct an approximating function $\hat{f}(x)$ in a piecewise fashion.

7.3 (Payoffs of Spreads)
We consider portfolios of two or more options of the same type with the same underlying stock. K_1, K_2, K are strikes with $K_1 < K_2$.

(a) A *butterfly spread* is a portfolio with

- one long call with strike K_1,
- one long call with strike K_2,
- two short calls with strike $K = \frac{K_2-K_1}{2}$.

The payoff is

$$\Psi(S) = \begin{cases} 0 & \text{for } S \le K_1 \\ S-K_1 & \text{for } K_1 < S \le K \\ K_2 - S & \text{for } K < S \le K_2 \\ 0 & \text{for } K_2 \le S\,. \end{cases}$$

(b) A *bull spread* is a portfolio with

- one long call with strike K_1,
- one short call with strike K_2,

The payoff is

$$\Psi(S) = \begin{cases} 0 & \text{for } S \le K_1 \\ S-K_1 & K_1 < S \le K_2 \\ K_2 - K_1 & K_2 < S\,. \end{cases}$$

For both spreads (a) and (b) explain and sketch the payoff. Apply the transformation (7.17) (Exercise 7.4) to derive the transformed payoff $u^*(x)$. For (b), apply the transformation with K_2.

7.4 (Transformation of Nonlinear Black–Scholes Models)
According Sect. 7.2, consider the following nonlinear PDE

$$V_t + \frac{1}{2}\sigma^2(t,S,V_{SS})S^2V_{SS} + (r-\delta)SV_S - rV + \hat{p}\max(\Psi - V, 0) = 0\,,$$

where $\sigma^2(t, S, V_{SS})$ depends on the particular model; r is the risk-free interest rate and δ is the continuous dividend yield. Apply the transformation (7.17)

$$x = \log(S/K),\ \tau = \sigma_0^2(T-t)/2,\ u(x,\tau) = \mathrm{e}^{-x}V(S,t)/K,$$

with $K > 0$ and a model-dependent parameter σ_0, and derive a PDE for u.

7.5 (Convergence of the Fully Implicit Method)
Two out of the three criteria for monotony in Theorem 7.2 are (i) and (ii). For

(a) Leland's model of transaction costs, with parameter γ, and
(b) the model of uncertain volatility with $\sigma_{\min} \le \sigma \le \sigma_{\max}$,

show that (i) and (ii) are satisfied. What are the constants c_+? For (b), σ^- of (7.12) suffices.

7.6 For a Lévy process X_t adapted to a filtration $\mathcal{F}_t$ show

$$\mathsf{E}(\mathrm{e}^{X_t} \mid \mathcal{F}_s) - \mathrm{e}^{X_s} = \mathsf{E}(\mathrm{e}^{X_{t-s}}) - \mathrm{e}^{X_0}\,.$$

7.7 (Project: Implementing a PIDE)
Set up a computer program to solve Merton's jump diffusion (7.37) numerically. To this end, concentrate on European-style vanilla options. Set up boundary conditions using (4.27), and apply a BDF implicit scheme. Think of how to choose $x_{\min}$, $x_{\max}$ in relation to the strike K.
Hint: For testing the core part of the program, set the jump intensity $\lambda = 0$ and compare to the Black–Scholes value.

7.8 (Fourier Transform)
Consider the Fourier transform

$$\mathcal{F}[f(u)] := \int_{-\infty}^{\infty} \mathrm{e}^{\mathrm{i}uy} f(y)\,\mathrm{d}y\,.$$

For the example $f(y) := \mathrm{e}^{-a|y|}$ and complex a show that

$$\int_{-A}^{A} \mathrm{e}^{\mathrm{i}uy} f(y)\,\mathrm{d}y$$

converges for $A \to \infty$ and $\mathrm{Re}(a) > 0$.

Appendix A
Financial Derivatives

A.1 Investment and Risk

Basic markets in which money is invested trade in particular with

- equities (stocks),
- bonds, and
- commodities.

Front pages of *The Financial Times* or *The Wall Street Journal* open with charts informing about the trading in these key markets. Such charts summarize a myriad of buys and sales, and of individual gains and losses. The assets bought in the markets are held in the portfolios of investors.

An easy way to buy or sell an asset is a spot contract, which is an agreement on the price, assuming delivery on the same date. Typical examples are furnished by the trading of stocks on an exchange, where the spot price is paid the same day. On the spot markets, gain or loss, or risks are clearly visible. The spot contracts are contrasted with those contracts that agree today ($t = 0$) to sell or buy an asset for a certain price at a certain *future time* ($t = T$). Historically, the first objects traded in this way have been commodities, such as agricultural products, metals, or oil. For example, a farmer may wish to sell in advance the crop expected for the coming season. Such trading has been extended to stocks, currencies and other financial instruments. Today there is a virtually unlimited variety of contracts on objects and their future state, from credit risks to weather prediction.

The future price of the underlying asset is usually unknown, it may move up or down in an unexpected way. For example, scarcity of a product will result in higher prices. Or the prices of stocks may decline sharply. But the contract must fix a price today, for an exchange of asset and payment that will happen in weeks or months. At maturity, the spot price usually differs from the agreed price of the contract. The difference between spot price and contract price can be significant.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

Hence contracts into the future are risky. Financial risk of assets is defined as the degree of uncertainty of their return.

No investment is really free of risks. But some bonds can come close to the idealization of being riskless. If the issuer of a bond has top ratings, then the return of a bond at maturity can be considered safe, and its value is known today with certainty. Such a bond is regarded as "riskless asset." The rate earned on a riskless asset is the *risk-free interest rate*. To avoid the complication of re-investing coupons, *zero-coupon bonds* are considered. The interest rate, denoted r, depends on the time to maturity T. The interest rate r is the continuously compounded interest which makes an initial investment S_0 grow to $S_0 e^{rT}$. We assume the interest rate r to be nonnegative. Often $r > 0$ will be taken constant throughout the time period $0 \leq t \leq T$. A candidate for r is the LIBOR.[1] Examples of bonds in real bond markets that come close to our idealized risk-free bond are issued by governments of AAA rated countries. See [191] for further introduction, and consult for instance *The Wall Street Journal* for market diaries.

All other assets are risky, with equities being the most prominent examples. *Hedging* is possible to protect against financial loss. Many hedging instruments have been developed. Since these financial instruments depend on the particular asset that is to be hedged, they are called *derivatives*. Main types of derivatives are *futures, forwards, options,* and *swaps*. They are explained below in some more detail. Tailoring and pricing derivatives is the core of *financial engineering*. Hedging with derivatives is the way to bound financial risks and to protect investments.

A.2 Financial Derivatives

Derivatives are instruments to assist and regulate agreements on transactions of the future. Derivatives can be traded on specialized exchanges.

Futures and **forwards** are agreements between two parties to buy or sell an asset at a certain time in the future for a certain delivery price. Both parties make a binding commitment, there is nothing to choose at a later time. For forwards no premiums are required and no money changes hands until maturity. A basic difference between futures and forwards is that futures contracts are traded on exchanges and are more formalized, whereas forwards are traded in the over-the-counter market (OTC). Also the OTC market usually involves financial institutions. Large exchanges on which futures contracts are traded are the Chicago Board of Trade (CBOT), the Chicago Mercantile Exchange (CME), and the Eurex.

Options are *rights* to buy or sell underlying assets for an *exercise price (strike)*, which is fixed by the terms of the option contract. That is, the purchaser of the option is *not obligated* to buy or sell the asset. This decision will be based on the payoff, which is contingent on the underlying asset's behavior. The buying or selling

[1] London Interbank Offered Rate.

of the underlying asset by exercising the option at a future date ($t = T$) must be distinguished from the purchase of the option (at $t = 0$, say), for which a premium is paid. After the Chicago Board of Options Exchange (CBOE) opened in 1973, the volume of the trading with options has grown dramatically.

Swaps are contracts regulating an exchange of cash flows at different future times. A common type of swap is the *interest-rate swap*, in which two parties exchange interest payments periodically, typically fixed-rate payments for floating-rate payments. Counterparty A agrees to pay to counterparty B a fixed interest rate on some notional principal, and in return party B agrees to pay party A interest at a floating rate on the same notional principal. The principal itself is not exchanged. Each of the parties borrows the money at his market. The interest payment is received from the counterparty and paid to the lending bank. Since the interest payments are in the same currency, the counterparties only exchange the interest differences. The *swap rate* is the fixed-interest rate fixed such that the deal (initially) has no value to either party ("par swap"). For a *currency swap*, the two parties exchange cash flows in different currencies.

An important application of derivatives is **hedging**. Hedging means to eliminate or limit risks. For example, consider an investor who owns shares and wants protection against a possible decline of the price below a value K in the next three months. The investor could buy put options on this stock with strike K and with a maturity that matches his three months time horizon. Since the investor can exercise his puts when the share price falls below K, it is guaranteed that the stock can be sold at least for the price K during the life time of the option. With this strategy the value of the stock is protected. The premium paid when purchasing the put option plays the role of an insurance premium. Hedging is intrinsic for calls. The writer of a call must hedge his position to avoid being hit by rising asset prices. Generally speaking, options and other derivatives facilitate the transfer of financial risks.

What kind of principle is so powerful to serve as basis for a fair valuation of derivatives? The concept is **arbitrage**, or rather the assumption that arbitrage is not possible in an idealized market. Arbitrage means the existence of a portfolio, which requires no investment initially, and which with guarantee makes no loss but very likely a gain at maturity. Or shorter: arbitrage is a self-financing trading strategy with zero initial value and positive terminal value.

If an arbitrage profit becomes known, arbitrageurs will take advantage and try to lock in.[2] This makes the arbitrage profits shrink. In an idealized market, information spreads rapidly and arbitrage opportunities become apparent. So arbitrage cannot last for long. Hence, in efficient markets at most very small arbitrage opportunities are observed in practice. For the modeling of financial markets this leads to postulate the **no-arbitrage principle**: One assumes an idealized market such that arbitrage is ruled out. Arguments based on the no-arbitrage principle resemble indirect proofs in mathematics: Suppose a certain financial situation. If this assumed scenario enables

[2]This assumes that investors prefer more to less, the basis for a rational pricing theory [269].

constructing an arbitrage opportunity, then there is a conflict to the no-arbitrage principle. Consequently, the assumed scenario is considered impossible.

For valuing derivatives one compares the return of the risky financial investment with the return of an investment that is free of risk. For the comparison, one calculates the gain the same initial capital would yield when invested in riskless bonds. To compare properly, one chooses a bond with time horizon T matching the terms of the derivative that is to be priced. Then, by the no-arbitrage principle, the risky investment should have the same price as the equivalent risk-free strategy. The construction and choice of derivatives to optimize portfolios and protect against extreme price movements is the essence of financial engineering.

The pricing of options is an ambitious task and requires sophisticated algorithms. Since this book is devoted to computational tools, mainly concentrating on options, the features of options are part of the text (Sect. 1.1 for standard options, and Sect. 6.1 for exotic options). This text will not enter further the discussion of forwards, futures, and swaps, with one exception: We choose the forward as an example (below) to illustrate the concept of arbitrage. For a detailed discussion of futures, forwards and swaps we refer to the literature, for instance to [31, 191, 251, 282, 339, 375].

A.3 Forwards and the No-Arbitrage Principle

As stated above, a forward is a contract between two parties to buy or sell an asset to be delivered at a certain time T in the future for a certain delivery price F. The time the parties agree on the forward contract (fixing T and F) is set to $t_0 = 0$. Since no premiums and no money change hands until maturity, the initial value of a forward is zero.

The party with the *long position* agrees to buy the underlying asset; the other party assumes the *short position* and agrees to sell the asset.

For the subsequent explanations S_t denotes the price of the asset at time t in the time interval $0 \le t \le T$. To fix ideas, we assume just one interest rate r for both borrowing or lending risk-free money over the time period $0 \le t \le T$. By the definition of the forward, at time of maturity T the party with the long position pays F to get the asset, which is then worth S_T.

Arbitrage Arguments
As will be shown next, the no-arbitrage principle enforces the *forward price* to be

$$F = S_0 \, \mathrm{e}^{rT} . \tag{A.1}$$

Thereby it is assumed that the asset does not produce any income (dividends) and does not cost anything until $t = T$.

Let us see how the no-arbitrage principle is invoked. We ask what the fair price F of a forward is at time $t = 0$, when the terms of a forward are settled. Then the spot price of the asset is S_0.

Assume first $F > S_0 e^{rT}$. Then an arbitrage strategy exists as follows: At $t = 0$ borrow S_0 at the interest rate r, buy the asset, and enter into a forward contract to sell the asset for the price F at $t = T$. When the time instant T has arrived, the arbitrageur completes the strategy by selling the asset $(+F)$ and by repaying the loan $(-S_0 e^{rT})$. The result is a riskless profit of $F - S_0 e^{rT} > 0$. This contradicts the no-arbitrage principle, so $F - S_0 e^{rT} \leq 0$ must hold.
Suppose next the complementary situation $F < S_0 e^{rT}$. In this case an investor who owns the asset[3] would sell it, invest the proceeds at interest rate r for the time period T, and enter a forward contract to buy the asset at $t = T$. In the end there would be a riskless profit of $S_0 e^{rT} - F > 0$. The conflict with the no-arbitrage principle implies $S_0 e^{rT} - F \leq 0$.
Combining the two inequalities $\leq$ and $\geq$ proves the equality. $[S_0 e^{r_1 T} \leq F \leq S_0 e^{r_2 T}$ in case of different rates $0 \leq r_1 \leq r_2$ for lending or borrowing]

One of the many applications of forwards is to hedge risks caused by foreign exchange.

Example (Hedging Against Exchange Rate Moves)
A U.S. corporation will receive one million euro in 3 months (on December 25), and wants to hedge against exchange rate moves. The corporation contacts a bank ("today" on September 25) to ask for the forward foreign exchange quotes. The 3-month forward exchange rate is that \$1.1428 will buy one euro, says the bank.[4] Why this? For completeness, on that day the spot rate is \$1.1457. If the corporation and the bank enter into the corresponding forward contract on September 25, the corporation is obligated to sell one million euro to the bank for \$1,142,800 on December 25. The bank then has a long forward contract on euro, and the corporation is in the short position.
Let us summarize the terms of the forward:

asset: one million euro
asset price S_t: the value of the asset in US \$ $(S_0 = \$1,145,700)$
maturity T= 1/4 (three months)
delivery price F: \$1,142,800 (forward price)

To understand the forward price in the above example, we need to generalize the basic forward price $S_0 e^{rT}$ to a situation where the asset produces income. In the foreign-exchange example, the asset earns the foreign interest rate, which we denote δ. To agree on a forward contract, $F e^{-rT} = S_0 e^{-\delta T}$, so

$$F = S_0 e^{(r-\delta)T}. \tag{A.2}$$

[3] Otherwise: *short sale*, selling a security the seller does not own.
[4] September 25, 2003.

(See [191].) On the date of the example the 3-month interest rate in the U.S. was $r = 1\%$, and in the euro world $\delta = 2\%$. So

$$S_0 \mathrm{e}^{(r-\delta)T} = 1145700 \mathrm{e}^{-0.01\frac{1}{4}} = 1142800$$

which explains the 3-month forward exchange rate of the example.

A.4 The Black–Scholes Equation

The Classic Equation
This appendix applies Itô's lemma to derive the Black–Scholes equation out of Assumptions 1.2. The basic assumption of a geometric Brownian motion of the stock price amounts to

$$\mathrm{d}S_t = \mu S_t \,\mathrm{d}t + \sigma S_t \,\mathrm{d}W_t \tag{A.3}$$

with constant μ and σ. Consider a portfolio consisting at time t of α_t shares of the asset with value S_t, and of β_t shares of the bond with value B_t. The bond is assumed riskless with

$$\mathrm{d}B_t = rB_t \,\mathrm{d}t\,. \tag{A.4}$$

At time t the wealth process of the portfolio is

$$\Pi_t := \alpha_t S_t + \beta_t B_t\,. \tag{A.5}$$

The portfolio is supposed to hedge a European option with value V_t, and payoff V_T at maturity T. So we aim at constructing α_t and β_t such that the portfolio *replicates* the payoff,

$$\Pi_T = V_T = \text{ payoff}\,. \tag{A.6}$$

The European option cannot be traded before maturity; neither any investment is required in $0 < t < T$ for holding the option nor is there any payout stream. To compare the values of V_t and Π_t, and to apply arbitrage arguments, the portfolio should have an equivalent property. Suppose the portfolio is "closed" for $0 < t < T$ in the sense that no money is injected into or removed from the portfolio. This amounts to the *self-financing property*

$$\mathrm{d}\Pi_t = \alpha_t \,\mathrm{d}S_t + \beta_t \,\mathrm{d}B_t\,. \tag{A.7}$$

That is, changes in the value of Π_t are due only to changes in the prices S or B. Equation (A.7) is equivalent to $S\,\mathrm{d}\alpha_t + B\,\mathrm{d}\beta_t = 0$, indicating that the quantities of stocks and bonds are continuously rebalanced—certainly an idealization.

Now the no-arbitrage principle is invoked. Replication (A.6) and self-financing (A.7) imply

$$\Pi_t = V_t \quad \text{for all } t \text{ in } 0 \le t \le T\,, \tag{A.8}$$

because both investments have the same payout stream. So the replicating and self-financing portfolio is equivalent to the risky option. The portfolio duplicates the risk of the option. How this fixes dynamically the quantities α_t and β_t of stocks and bonds is described next.

Assuming a sufficiently smooth value function $\Pi_t = V(S,t)$, we infer from Itô's lemma (Sect. 1.8)

$$\mathrm{d}\Pi = \left(\mu S\frac{\partial V}{\partial S} + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}\right)\mathrm{d}t + \sigma S\frac{\partial V}{\partial S}\,\mathrm{d}W\,. \tag{A.9}$$

On the other hand, substitute (A.3) and (A.4) into (A.7) and obtain another version of $\mathrm{d}\Pi$, namely,

$$\mathrm{d}\Pi = (\alpha\mu S + \beta r B)\,\mathrm{d}t + \alpha\sigma S\,\mathrm{d}W\,. \tag{A.10}$$

Because of uniqueness, the coefficients of both versions must match. Comparing the $\mathrm{d}W$ coefficients for $\sigma \neq 0$ leads to the hedging strategy

$$\alpha_t = \frac{\partial V(S_t,t)}{\partial S}\,. \tag{A.11}$$

Matching the $\mathrm{d}t$ coefficients gives a relation for β, in which the stochastic $\alpha\mu S$ terms drop out. The βB term is replaced via (A.5) and (A.8), which amounts to

$$S\frac{\partial V}{\partial S} + \beta B = V\,.$$

This results in the renowned Black–Scholes equation (1.5),

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0\,. \tag{A.12}$$

The terminal condition is given by (A.6).

Choosing in (A.11) the *delta hedge* $\Delta(S,t) := \alpha = \frac{\partial V}{\partial S}$ provides a dynamic strategy to eliminate the risk that lies in stochastic fluctuations and in the unknown drift μ of the underlying asset. The corresponding number of units of the underlying asset makes the portfolio (A.5) riskless. Hence the **delta** $\Delta = \frac{\partial V}{\partial S}$ plays a crucial role for a perfect hedging of portfolios. Of course, this delta hedging works under the stringent assumption that the market is correctly described by the model defined by Assumptions 1.2. But note that a continuous rebalancing of the portfolio is not realistic in practice. Real markets are incomplete and perfect hedges do not exist.

Delta hedging only neutralizes the prime risk of direct exposure to the underlying. Other risks, such as volatility risk and model risk, remain.

Having a model at hand as the Black–Scholes equation, it can be used inversely to calculate a probability distribution that matches underlying market prices (⟶ Exercise 1.19). This calibrates the model's crucial market parameter σ. Then, in turn, the hedging variable is calculated from the model. Symbolically, this application of the methods can be summarized by

$$V^{\mathrm{mar}} \longrightarrow \sigma \longrightarrow \Delta \,.$$

The methods of option valuation are intrinsic to this process. (In reality, hedging must be done in discrete time.)

In the above sense of eliminating risk, the modeling of V is risk neutral. Note that in the derivation of the Black–Scholes equation the standard understanding of constant coefficients μ, σ, r was actually not used. In fact the Black–Scholes equation holds also for time-varying deterministic functions $\mu(t), \sigma(t), r(t)$ (⟶ Exercise 1.25). For reference see, for example, [31, 110, 193, 345]. As will be shown below, there is a simple analytic formula for Δ in case of European options in the Black–Scholes model.

The Solution and the Greeks

The Black–Scholes equation has a closed-form solution. For a European call with vanilla payoff and continuous dividend yield δ as in (4.1) (in Sect. 4.1) the formulas are

$$d_1 := \frac{\log \frac{S}{K} + \left(r - \delta + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}\,, \tag{A.13}$$

$$d_2 := d_1 - \sigma\sqrt{T-t} = \frac{\log \frac{S}{K} + \left(r - \delta - \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}}\,, \tag{A.14}$$

$$V_{\mathrm{C}}(S,t) = S\mathrm{e}^{-\delta(T-t)}F(d_1) - K\mathrm{e}^{-r(T-t)}F(d_2)\,. \tag{A.15}$$

Here F denotes the standard normal cumulative distribution (with density f, compare Exercise 1.5 or Appendix E.2). The value $V_{\mathrm{P}}(S,t)$ of a put is obtained by applying the put-call parity on (A.15), see Exercise 1.1. For a continuous dividend yield δ as in (4.1) the put-call parity of European options is

$$V_{\mathrm{P}} = V_{\mathrm{C}} - S\mathrm{e}^{-\delta(T-t)} + K\mathrm{e}^{-r(T-t)} \tag{A.16}$$

from which

$$V_{\mathrm{P}}(S,t) = -S\mathrm{e}^{-\delta(T-t)}F(-d_1) + K\mathrm{e}^{-r(T-t)}F(-d_2) \tag{A.17}$$

follows. For options deep out of the money, when $V \approx 0$, an evaluation of the Black–Scholes formula suffers from cancellation. But for out-of-the-money options the Black–Scholes model is not recommended anyhow.

The Black-Scholes formulas (A.15) and (A.17) can be applied to European options also for discrete dividend payments. To this end, the stock price is reduced by the present value of all dividends during the life of the option [191, 282]. For example, assume one dividend is paid with known ex-dividend date t_D ($0 < t_D < T$) and known amount D. Then evaluate the Black–Scholes formula at $(\tilde{S}, t)$ with

$$\tilde{S} := S - D\mathrm{e}^{-r(t_D - t)}$$

instead of S, and with $\delta = 0$.

For nonconstant but known deterministic coefficient functions $\sigma(t)$, $r(t)$, $\delta(t)$, the closed-form solution is modified by introducing integral mean values [234, 291, 375, 378]. For example, replace the term $r(T-t)$ by the more general term $\int_t^T r(s)\,\mathrm{d}s$, and replace

$$\sigma\sqrt{T-t} \quad \longrightarrow \quad \left(\int_t^T \sigma^2(s)\,\mathrm{d}s\right)^{1/2}.$$

Differentiating the Black–Scholes formula gives delta, $\Delta = \frac{\partial V}{\partial S}$, as

$$\begin{aligned} \Delta &= \mathrm{e}^{-\delta(T-t)}\,F(d_1) && \text{for a European call,} \\ \Delta &= \mathrm{e}^{-\delta(T-t)}\,(F(d_1)-1) && \text{for a European put.} \end{aligned} \tag{A.18}$$

The delta Δ of (A.11) is the most prominent example of the "greeks." Also other derivatives of V are denoted by greek sounding names:

$$\text{gamma} = \frac{\partial^2 V}{\partial S^2}, \quad \text{theta} = \frac{\partial V}{\partial t}, \quad \text{vega} = \frac{\partial V}{\partial \sigma}, \quad \text{rho} = \frac{\partial V}{\partial r}.$$

As pointed out by [375], vega and rho, the derivatives with respect to parameters must be handled with care. In case of the Black–Scholes model, analytic expressions can be obtained by differentiating (A.15)/(A.17). For example,

$$\text{gamma} = \mathrm{e}^{-\delta(T-t)}\frac{f(d_1)}{\sigma S\sqrt{T-t}},$$

both for European put and call. For other greeks see, for instance, [172]. The essential parts of a derivation of the Black–Scholes formula (A.15) or (A.17) can be collected from this book; see for instance Exercise 1.10 or Exercise 3.12.

Hedging a Portfolio in Case of a Jump-Diffusion Process
Next consider a jump-diffusion process as described in Sect. 1.9, summarized by Eq. (1.74). The portfolio is the same as above, see (A.5), and we invoke the same assumptions such as replication and self-financing. Itô's lemma is applied in a piecewise fashion on the time intervals between jumps. Accordingly (A.9) is modified by adding the jumps in V with jumps sizes

$$\Delta V := V(S_{\tau+}, \tau) - V(S_{\tau-}, \tau)$$

for all jump instances τ_j. Consequently the term $\Delta V \, \mathrm{d}J$ is added to (A.9). On the other hand, (1.74) leads to add the term $\alpha(q-1)S \, \mathrm{d}J$ to (A.10). Comparing coefficients of the $\mathrm{d}W$ terms in both expressions of Π suggests the hedging strategy (A.11), namely, $\alpha = \frac{\partial V}{\partial S}$, and allows to shorten both versions of Π by subtracting equal terms. This is a piecewise argumentation for hedging the diffusion, not the jumps [375]. Let us denote the resulting values of the reduced portfolios by $\tilde{\Pi}$. Then (A.9) leads to

$$\mathrm{d}\tilde{\Pi} = \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} \right) \mathrm{d}t + (V(qS,t) - V(S,t)) \, \mathrm{d}J$$

and (A.10) becomes

$$\mathrm{d}\tilde{\Pi} = \left(rV - rS\frac{\partial V}{\partial S} \right) \mathrm{d}t + \frac{\partial V}{\partial S}(q-1)S \, \mathrm{d}J$$

(The reader may check.)

Different from the analysis leading to the Black–Scholes equation, $\mathrm{d}\tilde{\Pi}$ is not deterministic and it does not make sense to equate both versions. The risk can not be perfectly hedged away to zero in the case of jump-diffusion processes. That is, the market is not complete, and the equivalent martingale measure is not unique. Following [270], we apply the expectation operator over the random variable q to both versions of $\tilde{\Pi}$. Denote this expectation E, with

$$\mathsf{E}(X) = \int_{-\infty}^{\infty} x f_q(x) \, \mathrm{d}x \tag{A.19}$$

in case q_t has a density f_q that obeys $q > 0$. The expectations of both versions of $\mathsf{E}(\tilde{\Pi})$ can be equated. The result is

$$\begin{aligned} 0 = &\left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV \right) \mathrm{d}t \\ &+ \mathsf{E}\left([V(qS,t) - V(S,t) - (q-1)S\frac{\partial V}{\partial S}] \, \mathrm{d}J \right) . \end{aligned}$$

Since all stochastic terms are assumed independent, the second part of the equation is

$$\mathsf{E}[\ldots]\,\mathsf{E}(\mathrm{d}J)\,.$$

Using from (1.72)

$$\mathsf{E}(\mathrm{d}J) = \lambda\,\mathrm{d}t$$

and the abbreviation

$$c := \mathsf{E}(q-1)$$

this second part of the equation becomes

$$\{\mathsf{E}(V(qS,t)) - V(S,t) - cS\frac{\partial V}{\partial S}\}\,\lambda\,\mathrm{d}t\,.$$

The integral $c = \mathsf{E}(q-1)$ does not depend on V. This number c can be calculated via (A.19) as soon as a distribution for q is stipulated. For instance, one may assume a lognormal distribution, with relevant parameters fitted from marked data.[5] With the precalculated number c, the resulting differential equation can be ordered into

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + (r-\lambda c)S\frac{\partial V}{\partial S} - (\lambda + r)V + \lambda\mathsf{E}(V(qS,t)) = 0\,. \qquad \text{(A.20)}$$

Note that the last term is an integral taken over the unknown solution function $V(S,t)$. So the resulting equation is a partial integro-differential equation (PIDE). See Sect. 7.3 for a numerical solution.

A.5 Early-Exercise Curve

This appendix briefly discusses properties of the early-exercise curve S_f of standard American put and call options described by the Black–Scholes model, compare Sect. 4.5.1. Note that this excludes discrete dividend payments. Then the following holds for the

Put

(1) $S_\mathrm{f}(t)$ is continuously differentiable for $0 \le t < T$.
(2) $S_\mathrm{f}(t)$ is nondecreasing.

[5] The parameters are not the same as those in (1.64).

(3) A lower bound is

$$S_{\mathrm{f}}(t) > \tfrac{\lambda_2}{\lambda_2 - 1} K \,, \text{ where}$$
$$\lambda_2 = \tfrac{1}{\sigma^2} \left\{ -\left(r - \delta - \tfrac{\sigma^2}{2}\right) - \sqrt{\left(r - \delta - \tfrac{\sigma^2}{2}\right)^2 + 2\sigma^2 r} \right\} . \tag{A.21}$$

(4) An upper bound for $t < T$ is given by (4.32),

$$S_{\mathrm{f}}(t) < S_{\mathrm{f}}(T) := \lim_{\substack{t \to T \\ t < T}} S_{\mathrm{f}}(t) = \min\left(K, \tfrac{r}{\delta} K\right) = \begin{cases} K & \text{for } 0 \le \delta \le r \,, \\ \frac{r}{\delta} K & \text{for } r < \delta \,. \end{cases}$$

For proofs of (1) see [234, 282]. For the smoothness of the value function $V(S, t)$ on the continuation region, see [282]. Monotonicity of $V(S, t)$ with respect to time implies (2), as shown for instance in [234].

The monotonicity of S_{f} leads to conclude that a lower bound is obtained by $T \to \infty$. This limiting case is the perpetual option, compare Exercise 4.7. Specifically for $\delta = 0$, λ_2 simplifies, and the lower bound is $K \frac{q}{1+q}$, where $q := \frac{2r}{\sigma^2}$. For an illustration of a long horizon $T = 40$ see Fig. A.1. Simple calculus shows that λ_2 is the same as the λ_2 in Exercise 4.7.

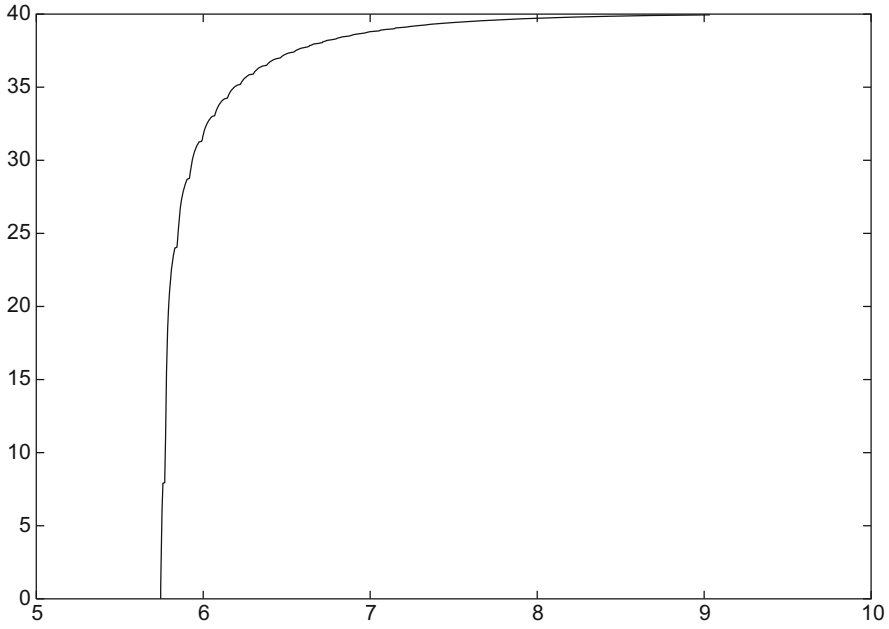

Fig. A.1 Approximation of the early-exercise curve of an American put with $K = 10$, $T = 40$, $r = 0.06$, $\sigma = 0.3$, $\delta = 0$, which leads to $\lambda_2 = -\frac{4}{3}$ and a lower bound of $\frac{4}{7}K$ (output of a finite-difference calculation, not smoothed)

Here we give a proof of property (4). First note $S_\mathrm{f} \le K$. The range $S_\mathrm{f} < S < K$ is part of the continuation region, where the Black–Scholes equation holds. Observe that

$$\frac{\partial V(S,T)}{\partial t} \le 0$$

because otherwise for t close to T a contradiction to $V \ge$ payoff results. How does this sign comply with the relation between r and δ ? For $t = T$ the value V_P^Am equals the payoff, $V_\mathrm{P}^\mathrm{Am}(S,T) = K - S$ for $S < K$. Substitute this into the Black–Scholes equation gives[6]

$$\frac{\partial V}{\partial t} + 0 - (r-\delta)S - rV = 0\,,$$

or

$$\frac{\partial V(S,T)}{\partial t} = rK - \delta S\,.$$

Hence, for $t = T$ and $S_\mathrm{f} < S < K$,

$$rK - \delta S \le 0$$

must hold.
First discuss the case $\delta > r$, or $\frac{r}{\delta}K < K$. Then either $S_\mathrm{f}(T) = \frac{r}{\delta}K$ (the assertion), or there is one of the two open intervals (i) $S_\mathrm{f}(T) < \frac{r}{\delta}K$, (ii) $\frac{r}{\delta}K < S_\mathrm{f}(T)$:

(i) There is S such that $S_\mathrm{f}(T) < S < \frac{r}{\delta}K$. Then

$$\frac{\partial V(S,T)}{\partial t} = rK - \delta S > 0\,,$$

which contradicts $\frac{\partial V(S,T)}{\partial t} \le 0$.

(ii) There is S such that $\frac{r}{\delta}K < S < S_\mathrm{f}(T)$. For each such S there is a small $\mathrm{d}t$ such that $(S, T - \mathrm{d}t)$ is in the stopping region (see Fig. 4.6). From $rK < \delta S$ and $rK\mathrm{d}t < \delta S\mathrm{d}t$ conclude

$$K(\mathrm{e}^{r\mathrm{d}t} - 1) < S(\mathrm{e}^{\delta \mathrm{d}t} - 1)\,.$$

That is, dividend over the period $\mathrm{d}t$ earns more than interest on K, and early exercise is not optimal. This contradicts the meaning of $S < S_\mathrm{f}(T)$.

[6] Recall the context: V means V_P^Am.

Finally we discuss the case $\delta \le r$. By the definition of S_f, $S_f(T) > K$ cannot happen. Assume $S_f(T) < K$. Then for $S_f(T) < S < K$ and $t = T$

$$\underbrace{\frac{dV}{dt}}_{\le 0} = \underbrace{rK - \delta S}_{>0}$$

leads to a contradiction. So $S_f(T) = K$ for $\delta \le r$. Both assertions are summarized to

$$\lim_{\substack{t \to T \\ t<T}} S_f(t) = \min\left(K, \frac{r}{\delta}K\right) .$$

We conclude with listing the properties of an American **Call** with $\delta > 0$:

(1) $S_f(t)$ is continuously differentiable for $0 \le t < T$.
(2) $S_f(t)$ is nonincreasing.
(3) An upper bound is

$$\begin{aligned} &S_f(t) < \tfrac{\lambda_1}{\lambda_1 - 1}K \text{ , where} \\ &\lambda_1 = \tfrac{1}{\sigma^2}\left\{ -\left(r - \delta - \tfrac{\sigma^2}{2}\right) + \sqrt{\left(r - \delta - \tfrac{\sigma^2}{2}\right)^2 + 2\sigma^2 r}\right\} . \end{aligned} \tag{A.22}$$

(4) A lower bound for $t < T$ is given by (4.33),

$$S_f(t) > \max\left(K, \frac{r}{\delta}K\right) .$$

Derivations are analogous as in the case of the American put. We note from properties (4) two extreme cases for $t \to T$:

$$\begin{aligned} \text{put} &: r \to 0 \Rightarrow \quad S_f \to 0 \\ \text{call} &: \delta \to 0 \Rightarrow S_f \to \infty . \end{aligned}$$

The second assertion is another clue that for a call early exercise will never be optimal when no dividends are paid ($\delta = 0$). Likewise, an American put is identical to the European counterpart in case $r = 0$.

By the way, the **symmetry** of the above properties is reflected by

$$\begin{aligned} &S_{f,\,\text{call}}(t; r, \delta)\, S_{f,\,\text{put}}(t; \delta, r) = K^2 , \\ &V_C^{Am}(S, T - t; K, r, \delta) = V_P^{Am}(K, T - t; S, \delta, r) . \end{aligned} \tag{A.23}$$

This put-call symmetry is derived in [101, 267]. Note that the put-call symmetry is derived under the assumptions of the Black–Scholes model, whereas the put-call

parity for European options is independent of the underlying model. For discrete dividend payments, S_f needs not be continuous [273, 368].

A.6 Equations with Volatility Function

An extension of the Black–Scholes equation allows for variable coefficients,

$$\frac{\partial V}{\partial t} + \frac{\sigma(S,t)^2}{2} S^2 \frac{\partial^2 V}{\partial S^2} + (r(S,t) - \delta(S,t)) S \frac{\partial V}{\partial S} - r(S,t) V = 0\,, \tag{A.24}$$

see, for example, [2, 10, 28]. This assumes r, δ, σ to be deterministic functions. For the special case of constant coefficients, the transformation (4.3) leads to the (backward) heat equation (4.2), see also Exercise 1.4. For variable coefficients this transformation can not be applied.

Variable Volatility
In many applications, r and δ can be assumed constant, and only σ is taken as function $\sigma(S,t)$, for example, in local volatility problems. In such a situation, the transformation of the independent variables

$$\begin{aligned} x &:= \log(S/K) - (r-\delta)t\,, \\ \hat{V}(x,t) &:= V(S,t)\,, \quad \hat{\sigma}(x,t) := \sigma(S,t) \end{aligned} \tag{A.25}$$

leads to

$$\frac{\partial \hat{V}}{\partial t} + \frac{1}{2}\hat{\sigma}^2 \left(\frac{\partial^2 \hat{V}}{\partial x^2} - \frac{\partial \hat{V}}{\partial x} \right) - r\hat{V} = 0\,.$$

(The reader is encouraged to show this as an exercise.) This version still has a convection term $\frac{\partial \hat{V}}{\partial x}$, which may be the source of dispersion. With a further transformation, the scaling $\hat{V}(x,t) \leftrightarrow y(x,t)$ via

$$\hat{V}(x,t) = K \exp\left(\frac{x}{2} + rt\right) y(x,t)\,, \tag{A.26}$$

which is an important ingredient of Exercise 1.4, we arrive at

$$\frac{\partial y}{\partial t} + \frac{1}{2}\hat{\sigma}^2(x,t) \left(\frac{\partial^2 y}{\partial x^2} - \frac{1}{4} y \right) = 0\,.$$

Consult [197] for these transformations, the lack of dispersion of related numerical schemes, and for the higher-dimensional case. Of course, for the backward situation of the Black–Scholes scenario, in addition the time is reversed by $\tau := T - t$ in

order to obtain the well-posed problem

$$\frac{\partial y}{\partial \tau} - \frac{1}{2}\hat{\sigma}^2(x,\tau)\left(\frac{\partial^2 y}{\partial x^2} - \frac{1}{4}y\right) = 0\,. \tag{A.27}$$

Dupire's Equation

In practice, an important question is how to choose the local volatility function $\sigma(S,t)$ such that the corresponding model (A.27) yields results consistent with the market. In particular, one attempts to match the *volatility smile*, which amounts to a somewhat convex shape of the values of implied volatility over the strike K.

Recall that the value function depends on

$$V(S,t;\,K,T;\,r,\sigma,\delta).$$

For Black and Scholes, K and T are fixed, and $V(S,t)$ is calculated for independent variables S,t. Dupire [112] switches the role of these variables: He keeps S,t fixed and calculates $V(.,.;\,K,T)$ for independent variables K,T.

Dupire's local volatility model is built as follows: For a general diffusion process $\mathrm{d}S = a(S,t)\,\mathrm{d}t + b(S,t)\,\mathrm{d}W$, consider a European call with the integral representation

$$\begin{aligned} V(S_0,t_0;\,K,T) &= \mathrm{e}^{-r(T-t_0)}\int_{-\infty}^{\infty}(S_T-K)^+\,p\,(S_T,T;S_0,t_0)\,\mathrm{d}S_T \\ &= \mathrm{e}^{-r(T-t_0)}\int_{K}^{\infty}(S_T-K)\,p(S_T,T;S_0,t_0)\,\mathrm{d}S_T\,. \end{aligned} \tag{A.28}$$

Here $p(S_T,T;S_0,t_0)$ is the probability density of a transition *forward* from (S_0,t_0) to (S_T,T). A special case is (1.64)/(1.66), where f_{GBM} characterizes the transition with respect to GBM with $a = rS$, $b = \sigma S$. For general $a(S,t)$, $b(S,t)$, the transition probability p solves a partial differential equation, namely, the famous **Fokker–Planck Equation**

$$\frac{\partial p}{\partial T} - \frac{1}{2}\frac{\partial^2}{\partial S_T^2}[b(S_T,T)^2\cdot p(S_T,T;S_0,t_0)] + \frac{\partial}{\partial S_T}[a(S_T,T)\cdot p(S_T,T;S_0,t_0)] = 0 \tag{A.29}$$

with initial conditions for $T = t_0$:

$$p(S_T,T=t_0;S_0,t_0) = \delta(S_T-S_0) = \text{ Dirac's delta function.}$$

To deduce an equation for V depending on K,T, the partial derivatives

$$\frac{\partial V}{\partial T},\ \frac{\partial V}{\partial K},\ \frac{\partial^2 V}{\partial K^2}$$

of (A.28) are calculated, which yields expressions with partial derivatives of p. The Fokker–Planck equation (A.29) substitutes $\frac{\partial p}{\partial T}$. Specifically, for a Black–Scholes type process with

$$\begin{aligned} a(S,t) &= (r-\delta)S \\ b(S,t) &= \sigma(S,t)S \end{aligned} \tag{A.30}$$

(δ again the dividend rate), one arrives at

$$\frac{\partial V}{\partial T} = \frac{1}{2}\sigma(K,T)^2K^2\frac{\partial^2 V}{\partial K^2} - (r-\delta)K\frac{\partial V}{\partial K} - \delta V\,. \tag{A.31}$$

This is the Dupire PDE. Compare it with the Black–Scholes equation, and notice the different sign of the diffusion term (the second-order derivative) of the Dupire equation, which reflects its forward character. The (K,T)-domain for Dupire is $T \geq t$, $K > 0$, and $V(S,t;K,T=t) = (S-K)^+$ for a call is an *initial* condition. Formally this is (1.3), but here K is the independent variable and S is the constant. If a model for the local volatility function σ is postulated, then European options of *all* strikes K and maturities T can be calculated in a single "sweep" by solving the forward equation (A.31). Transformations analogous to (A.25), (A.26) again lead to (A.27), with τ replaced by T.

Also the inverse problem is of interest. One can show that the numerator and the denominator of the radicand below in (A.32) are nonnegative. Hence the Dupire equation can be solved for $\sigma(K,T)$,

$$\sigma(K,T) = \sqrt{2\,\frac{\frac{\partial V}{\partial T} + (r-\delta)K\frac{\partial V}{\partial K} + \delta V}{K^2\frac{\partial^2 V}{\partial K^2}}}\,. \tag{A.32}$$

Upon calibrating the formula (A.32), one must regard its sensitivity to noise in the data, in particular, for small denominators. For example, using the moving least squares algorithm of [154], the derivatives

$$a_1 := \frac{\partial V}{\partial T},\ a_2 := \frac{\partial V}{\partial K},\ a_3 := \frac{\partial^2 V}{\partial K^2}$$

can be extracted from market data, as well as $a_0 := V$, all depending on $(S,t;K,T)$. This gives an approximation

$$\bar{\sigma}(K,T) = \sqrt{2(a_1 + (r-\delta)Ka_2 + \delta a_0)/(a_3K^2)}$$

of the volatility function (A.32). After the approximation $\bar{\sigma}$ is calibrated based on vanilla data, it can be used to price nonvanilla instruments. There are further approximations for $\sigma(K,T)$, consult [102, 128, 375]. A reference on the Fokker–Planck equation is [318].

Appendix B
Stochastic Tools

B.1 Essentials of Stochastics

This appendix lists some basic instruments and notations of probability theory and statistics. For further foundations we refer to the literature, for example, [37, 127, 131, 199, 274, 340].

Let Ω be a *sample space*. In our context Ω is mostly uncountable, for example, $\Omega = \mathbb{R}$. A subset of Ω is an *event* and an element $\omega \in \Omega$ is a sample point. The sample space Ω represents all possible scenarios. Classes of subsets of Ω must satisfy certain requirements to be useful for probability. One assumes that such a class $\mathcal{F}$ of events is a *σ-algebra* or a *σ-field*.[1] That is, $\Omega \in \mathcal{F}$, and $\mathcal{F}$ is closed under the formation of complements and countable unions. In our finance scenario, $\mathcal{F}$ represents the space of events that are observable in a market. If t denotes time, all information available until t can be regarded as a σ-algebra $\mathcal{F}_t$. Then it is natural to assume a *filtration*—that is, $\mathcal{F}_t \subseteq \mathcal{F}_s$ for $t < s$.

The sets in $\mathcal{F}$ are also called *measurable sets*. A measure on these sets is the probability measure P, a real-valued function taking values in the interval $[0, 1]$, with the three axioms

$$\mathsf{P}(A) \geq 0 \ \text{ for all events } A \in \mathcal{F}\,, \qquad \mathsf{P}(\Omega) = 1\,,$$
$$\mathsf{P}\left(\bigcup_{i=1}^{\infty} A_i\right) = \sum_{i=1}^{\infty} \mathsf{P}(A_i) \ \text{ for any sequence of disjoint } A_i \in \mathcal{F}\,.$$

The triplet $(\Omega, \mathcal{F}, \mathsf{P})$ is called a *probability space*. An assertion is said to hold *almost everywhere* (P–a.e.) if it is wrong with probability 0.

[1] This notation with σ is not related with volatility.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

A real-valued function X on Ω is called **random variable** if the sets

$$\{X \le x\} := \{\omega \in \Omega \mid X(\omega) \le x\} = X^{-1}((-\infty, x])$$

are measurable for all $x \in \mathbb{R}$. That is, $\{X \le x\} \in \mathcal{F}$. This book does not explicitly indicate the dependence on the sample space Ω. We write X instead of $X(\omega)$, or X_t or $X(t)$ instead of $X_t(\omega)$ when the random variable depends on a parameter t.

For $x \in \mathbb{R}$ a **distribution function** $F(x)$ of X is defined by the probability P that $X \le x$,

$$F(x) := \mathsf{P}(X \le x)\,. \tag{B.1}$$

Distributions are nondecreasing, right-continuous, and satisfy the limits

$$\lim_{x \to -\infty} F(x) = 0 \quad \text{and} \quad \lim_{x \to +\infty} F(x) = 1\,.$$

Every absolutely continuous distribution F has a derivative almost everywhere, which is called **density function**. For all $x \in \mathbb{R}$ a density function f has the properties $f(x) \ge 0$ and

$$F(x) = \int_{-\infty}^{x} f(t)\,\mathrm{d}t\,. \tag{B.2}$$

To stress the dependence on X, the distribution is also written F_X and the density f_X. If X has a density f then the kth *moment* is defined as

$$\mathsf{E}(X^k) := \int_{-\infty}^{\infty} x^k f(x)\,\mathrm{d}x = \int_{-\infty}^{\infty} x^k\,\mathrm{d}F(x)\,, \tag{B.3}$$

provided the integrals exist. The most important moment of a distribution is the **expected value** or **mean**

$$\mu := \mathsf{E}(X) := \int_{-\infty}^{\infty} x f(x)\,\mathrm{d}x\,. \tag{B.4}$$

The **variance** is defined as the second central moment

$$\sigma^2 := \mathsf{Var}(X) := \mathsf{E}((X-\mu)^2) \;=\; \int_{-\infty}^{\infty} (x-\mu)^2 f(x)\,\mathrm{d}x\,. \tag{B.5}$$

A consequence is

$$\sigma^2 = \mathsf{E}(X^2) - \mu^2\,.$$

The expectation depends on the underlying probability measure P, which is sometimes emphasized by writing E_{P}. Here and in the sequel we assume that the integrals exist. The square root $\sigma = \sqrt{\mathsf{Var}(X)}$ is the *standard deviation* of X. For $\alpha, \beta \in \mathbb{R}$ and two random variables X, Y on the same probability space, expectation and variance satisfy

$$\begin{aligned} \mathsf{E}(\alpha X + \beta Y) &= \alpha \mathsf{E}(X) + \beta \mathsf{E}(Y)\,, \\ \mathsf{Var}(\alpha X + \beta) &= \mathsf{Var}(\alpha X) = \alpha^2 \mathsf{Var}(X)\,. \end{aligned} \tag{B.6}$$

The *covariance* of two random variables X and Y is

$$\mathsf{Cov}(X, Y) := \mathsf{E}\left((X - \mathsf{E}(X))(Y - \mathsf{E}(Y))\right) = \mathsf{E}(XY) - \mathsf{E}(X)\mathsf{E}(Y)\,,$$

from which

$$\mathsf{Var}(X \pm Y) = \mathsf{Var}(X) + \mathsf{Var}(Y) \pm 2\mathsf{Cov}(X, Y) \tag{B.7}$$

follows. More general, the covariance between the components of a *vector* X is the matrix

$$\mathsf{Cov}(X) = \mathsf{E}[(X - \mathsf{E}(X))(X - \mathsf{E}(X))^{tr}] = \mathsf{E}(XX^{tr}) - \mathsf{E}(X)\mathsf{E}(X)^{tr}\,, \tag{B.8}$$

where the expectation E is applied to each component. The diagonal carries the variances of the components of X. Back to the scalar world: Two random variables X and Y are called *independent* if

$$\mathsf{P}(X \le x, Y \le y) = \mathsf{P}(X \le x)\,\mathsf{P}(Y \le y)\,.$$

Independent variables are uncorrelated. For independent random variables X and Y the equations

$$\begin{aligned} \mathsf{E}(XY) &= \mathsf{E}(X)\,\mathsf{E}(Y)\,, \\ \mathsf{Var}(X + Y) &= \mathsf{Var}(X) + \mathsf{Var}(Y) \end{aligned}$$

are valid; analogous assertions hold for more than two independent random variables. For convex functions ϕ, Jensen's inequality holds:

$$\phi(\mathsf{E}(X)) \le \mathsf{E}(\phi(X))\,.$$

Normal Distribution (Gaussian Distribution) The density of the normal distribution is

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)\,. \tag{B.9}$$

$X \sim \mathcal{N}(\mu, \sigma^2)$ means: X is normally distributed with expectation μ and variance σ^2. An implication is $Z = \frac{X-\mu}{\sigma} \sim \mathcal{N}(0,1)$, which is the *standard* normal distribution, or $X = \sigma Z + \mu \sim \mathcal{N}(\mu, \sigma^2)$. The values of the corresponding distribution function $F(x)$ can be approximated by analytic expressions (⟶ Appendix E.2) or numerically (⟶ Exercise 1.5). For multidimensional Gaussian, see Sect. 2.3.4.

Uniform distribution over an interval $a \le x \le b$: The density is

$$f(x) = \frac{1}{b-a} \quad \text{for } a \le x \le b\,; \quad f = 0 \text{ elsewhere.} \tag{B.10}$$

This uniform distribution has expected value $\frac{1}{2}(a+b)$ and variance $\frac{1}{12}(b-a)^2$. If the uniform distribution is considered over a higher-dimensional domain $\mathcal{D}$, then the value of the density is is the inverse of the volume of $\mathcal{D}$,

$$f = \frac{1}{\mathrm{vol}(\mathcal{D})} \cdot \mathbf{1}_{\mathcal{D}}\,.$$

For example, on a unit disc we have the value $f = 1/\pi$.

Estimates of mean and variance of a normally distributed random variable X from a sample of M realizations $x_1, \ldots, x_M$ are given by

$$\begin{aligned} \hat{\mu} &:= \frac{1}{M} \sum_{k=1}^{M} x_k\,, \\ \hat{s}^2 &:= \frac{1}{M-1} \sum_{k=1}^{M} (x_k - \hat{\mu})^2\,. \end{aligned} \tag{B.11}$$

These expressions of the sample mean $\hat{\mu}$ and the sample variance $\hat{s}^2$ satisfy $\mathsf{E}(\hat{\mu}) = \mu$ and $\mathsf{E}(\hat{s}^2) = \sigma^2$. That is, $\hat{\mu}$ and $\hat{s}^2$ are unbiased estimates. For the computation see Exercise 1.6, or [306]. The covariance (B.8) is calculated analogously.

Central Limit Theorem Suppose $X_1, X_2, \ldots$ are independent and identically distributed (i.i.d.) random variables, and $\mu := \mathsf{E}(X_i)$, $S_n := \sum_{i=1}^{n} X_i$, $\sigma^2 = \mathsf{E}(X_i - \mu)^2$. Then for each a

$$\lim_{n\to\infty} \mathsf{P}\left(\frac{S_n - n\mu}{\sigma\sqrt{n}} \le a \right) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{a} \mathrm{e}^{-z^2/2}\,\mathrm{d}z \quad (= F(a))\,. \tag{B.12}$$

As a consequence, the probability that $\hat{\mu}$ hits—for large enough n—the interval

$$\mu - a\frac{\sigma}{\sqrt{n}} \le \hat{\mu} \le \mu + a\frac{\sigma}{\sqrt{n}}$$

is $F(a) - F(-a) = 2F(a) - 1$. For example, $a = 1.96$ leads to a probability of 0.95. That is, the 95% confidence interval has a (half) width of about $2\sigma/\sqrt{n}$.

The **weak law of large numbers** states that for all $\epsilon > 0$

$$\lim_{n\to\infty} \mathsf{P}\left(\left|\frac{S_n}{n} - \mu\right| > \epsilon\right) = 0\,,$$

and the strong law says $\mathsf{P}(\lim_{n\to\infty} \frac{S_n}{n} = \mu) = 1$.
For a **discrete probability space** the sample space Ω is countable. The expectation and the variance of a discrete random variable X with realizations x_i are given by

$$\begin{aligned} \mu &= \mathsf{E}(X) = \textstyle\sum_{\omega\in\Omega} X(\omega)\mathsf{P}(\omega) = \sum_i x_i\, \mathsf{P}(X = x_i)\,, \\ \sigma^2 &= \textstyle\sum_i (x_i - \mu)^2\, \mathsf{P}(X = x_i)\,. \end{aligned} \tag{B.13}$$

Occasionally, the underlying probability measure P is mentioned in the notation. For example, a Bernoulli experiment[2] with $\Omega = \{\omega_1, \omega_2\}$ and $\mathsf{P}(\omega_1) = p$ has expectation

$$\mathsf{E}_\mathsf{P}(X) = pX(\omega_1) + (1-p)X(\omega_2)\,.$$

The probability that for n Bernoulli trials the event ω_1 occurs exactly k times, is

$$\mathsf{P}(X = k) = b_{n,p}(k) := \binom{n}{k} p^k (1-p)^{n-k} \quad \text{for } 0 \le k \le n\,. \tag{B.14}$$

The *binomial coefficient* defined as

$$\binom{n}{k} = \frac{n!}{(n-k)!\,k!}$$

states in how many ways k elements can be chosen out of a population of size n.
For the **binomial distribution** $b_{n,p}(k)$ the mean is $\mu = np$, and the variance $\sigma^2 = np(1-p)$. The probability that event ω_1 occurs at least M times is

$$\mathsf{P}(X \ge M) = B_{n,p}(M) := \sum_{k=M}^{n} \binom{n}{k} p^k (1-p)^{n-k}\,. \tag{B.15}$$

This follows from the axioms of the probability measure.

[2] Repeated independent trials, where only two possible outcomes are possible for each trial, such as tossing a coin.

For the **Poisson distribution** the probability that an event occurs exactly k times within a specified (time) interval is given by

$$\mathsf{P}(X = k) = \frac{a^k}{k!}\mathrm{e}^{-a} \quad \text{for } k = 0, 1, 2, \ldots \tag{B.16}$$

and a constant $a > 0$. Its mean and variance are both a.

Convergence in the Mean A sequence X_n is said to converge in the (square) mean to X, if $\mathsf{E}(X_n^2) < \infty$, $\mathsf{E}(X^2) < \infty$ and if

$$\lim_{n\to\infty} \mathsf{E}((X - X_n)^2) = 0\,.$$

A notation for convergence in the mean is

$$\text{l.i.m.}_{n\to\infty} X_n = X\,.$$

B.2 More Advanced Topics

General Itô Formula
Let $\mathrm{d}X_t = a(.)\mathrm{d}t + b(.)\mathrm{d}W_t$, where X_t is n-dimensional, $a(.)$ too, and $b(.)$ $(n \times m)$ matrix and W_t m-dimensional, with uncorrelated components, see (1.58). Let g be twice continuously differentiable, defined for (X, t) with values in $\mathbb{R}$. Then $g(X, t)$ is an Itô process with

$$\mathrm{d}g = \left[\frac{\partial g}{\partial t} + g_x^{tr} a + \frac{1}{2}\,\text{trace}\,(b^{tr} g_{xx} b)\right]\mathrm{d}t + g_x^{tr} b\,\mathrm{d}W_t\,. \tag{B.17}$$

g_x is the gradient vector of the first-order partial derivatives with respect to x, and g_{xx} is the matrix of the second-order derivatives, all evaluated at (X, t). The matrix $b^{tr} g_{xx} b$ is $m \times m$. (Recall that the trace of a matrix is the sum of the diagonal elements.)

Equation (B.17) is derived via Taylor expansion. The linear terms $g_x^{tr}\,\mathrm{d}X$ are straightforward. The quadratic terms are

$$\frac{1}{2}\,\mathrm{d}X^{tr} g_{xx}\,\mathrm{d}X\,,$$

from which the order $\mathrm{d}t$ terms remain

$$\frac{1}{2}(b\,\mathrm{d}W)^{tr} g_{xx} b\,\mathrm{d}W = \frac{1}{2}\,\mathrm{d}W^{tr} b^{tr} g_{xx} b\,\mathrm{d}W =: \frac{1}{2} dW^{tr} A\,\mathrm{d}W\,.$$

These remaining terms are

$$\frac{1}{2}\,\text{trace}\,(A)\,\mathrm{d}t\,.$$

A matrix manipulation shows that the elements of $b^{tr} g_{xx} b$ are

$$\sum_{i=1}^{n}\sum_{j=1}^{n} g_{x_i x_j} b_{il} b_{jk} \quad \text{for } l,k = 1,\ldots,m\,.$$

This is different from $bb^{tr} g_{xx}$, but the traces are equal:

$$\text{trace}\,(b^{tr} g_{xx} b) = \text{trace}\,(bb^{tr} g_{xx}) = \sum_{i,j} \frac{\partial^2 g}{\partial x_i \partial x_j} \underbrace{\sum_{k=1}^{m} b_{ik} b_{jk}}_{=:c_{ij}}\,.$$

Consult also [291].

Exercise Let X be vector and Y scalar, where $\mathrm{d}X = a_1\,\mathrm{d}t + b_1\,\mathrm{d}W$, $\mathrm{d}Y = a_2\,\mathrm{d}t + b_2\,\mathrm{d}W$, and consider $g(X,Y) := XY$. Show

$$\begin{aligned}\mathrm{d}(XY) &= Y\,\mathrm{d}X + X\,\mathrm{d}Y + \mathrm{d}X\,\mathrm{d}Y \\ &= (Xa_2 + Ya_1 + b_1 b_2)\,\mathrm{d}t + (Xb_2 + Yb_1)\,\mathrm{d}W\,.\end{aligned} \tag{B.18}$$

Application

$$\mathrm{d}S = rS\,\mathrm{d}t + \sigma S\,\mathrm{d}\hat{W} \;\Rightarrow\; \mathrm{d}(\mathrm{e}^{-rt}S) = \mathrm{e}^{-rt}\sigma S\,\mathrm{d}\hat{W} \tag{B.19}$$

for any Wiener process $\hat{W}$.

Filtration of a Stochastic Process
The filtration of a Brownian motion is defined as

$$\mathcal{F}_t^W := \sigma\{W_s \mid 0 \le s \le t\}\,. \tag{B.20}$$

Here $\sigma\{.\}$ denotes the smallest σ-algebra containing the sets put in braces. $\mathcal{F}_t^W$ is a model of the information available at time t, since it includes every event based on the history of W_s, $0 \le s \le t$. The null sets $\mathcal{N}$ are included in the sense $\mathcal{F}_t := \sigma(\mathcal{F}_t^W \cup \mathcal{N})$ ("augmented"). In the same way, the natural filtration of a general stochastic process X is built.

Conditional Expectation
We recall conditional expectation because it is needed for martingales. Let $\mathcal{G}$ be a sub σ-algebra of $\mathcal{F}$. $\mathsf{E}(X \mid \mathcal{G})$ is defined to be the (unique) $\mathcal{G}$-measurable random variable Y with the property

$$\mathsf{E}(XZ) = \mathsf{E}(YZ)$$

for all $\mathcal{G}$-measurable Z (such that $\mathsf{E}(XZ) < \infty$). This is the conditional expectation of X given $\mathcal{G}$. Or, following [108], an equivalent definition is via

$$\int_A \mathsf{E}(Y \mid \mathcal{G})\, \mathrm{d}\mathsf{P} = \int_A Y\, \mathrm{d}\mathsf{P} \quad \text{for all} \quad A \in \mathcal{G}\,.$$

In case $\mathsf{E}(X \mid Y)$, set $\mathcal{G} = \sigma(Y)$. For properties of conditional expectation consult, for example, [274, 340].

Martingales
Assume the standard scenario $(\Omega, \mathcal{F}, \mathcal{F}_t, \mathsf{P})$ with a filtration $\mathcal{F}_t \subset \mathcal{F}$.

Definition $\mathcal{F}_t$**-Martingale** *M_t with respect to P is a process, which is "adapted" (that is, $\mathcal{F}_t$-measurable), $\mathsf{E}(|M_t|) < \infty$, and*

$$\mathsf{E}(M_t \mid \mathcal{F}_s) = M_s \quad (\mathsf{P}\text{-a.s.}) \ \textit{for all} \ \ t, s \ \textit{with} \ \ s \le t\,. \tag{B.21}$$

The martingale property means that at time instant s with given information set $\mathcal{F}_s$ all variations of M_t for $t > s$ are unpredictable; M_s is the best forecast. The SDE of a martingale has no drift term.

Examples

- any Wiener process W_t,
- $W_t^2 - t$ for any Wiener process W_t,
- $\exp(\lambda W_t - \frac{1}{2}\lambda^2 t)$ for any $\lambda \in \mathbb{R}$ and any Wiener process W_t,
- $J_t - \lambda t$ for any Poisson process J_t with intensity λ.

For martingales, consult for instance [108, 283, 291, 307, 339, 340]. For an adapted process γ define a process Z_t^γ by

$$Z_t^\gamma := \exp\left(-\frac{1}{2}\int_0^t \gamma_s^2\, \mathrm{d}s - \int_0^t \gamma_s\, \mathrm{d}W_s\right). \tag{B.22}$$

Since $Z_0 = 1$, the integral equation

$$\log Z_t = \log Z_0 - \frac{1}{2}\int_0^t \gamma_s^2\, \mathrm{d}s - \int_0^t \gamma_s\, \mathrm{d}W_s$$

follows, which is the SDE

$$\mathrm{d}(\log Z_t) = (0 - \frac{1}{2}\gamma_t^2)\, \mathrm{d}t - \gamma_t\, \mathrm{d}W_t\,.$$

This is the Itô SDE for $\log Z_t$ when Z solves the drift-free $\mathrm{d}Z_t = -Z_t\gamma_t\, \mathrm{d}W_t$, $Z_0 = 1$. In summary, Z_t is the unique Itô process such that $\mathrm{d}Z_t = -Z_t\gamma_t\, \mathrm{d}W_t$, $Z_0 = 1$. Let Z^γ be a martingale. From the martingale properties, $\mathsf{E}(Z_T^\gamma) = \mathsf{E}(Z_0^\gamma) = 1$. Hence the

Radon-Nikodym framework assures that an equivalent probability measure $\mathsf{Q}(\gamma)$ can be defined by

$$\frac{\mathrm{d}\mathsf{Q}(\gamma)}{\mathrm{d}\mathsf{P}} = Z_T^\gamma \quad \text{or} \quad \mathsf{Q}(A) := \int_A Z_T^\gamma \, \mathrm{d}\mathsf{P} \,. \tag{B.23}$$

Girsanov's Theorem *Suppose a process γ is such that Z^γ is a martingale. Then*

$$W_t^\gamma := W_t + \int_0^t \gamma_s \, \mathrm{d}s \tag{B.24}$$

is a Wiener process and martingale under $\mathsf{Q}(\gamma)$.

B.3 State-Price Process

Normalizing
A fundamental result of Harrison and Pliska [170] states that the existence of a martingale implies an arbitrage-free market. This motivates searching for a martingale. Since martingales have no drift term, we attempt to construct SDEs without drift.

Let X_t be a vector of asset prices, and b_t the corresponding vector of a trading strategy. Then the scalar product $b_t^{tr} X_t$ represents the wealth of the portfolio. The trading strategy is self-financing when $\mathrm{d}(b^{tr} X) = b^{tr} \mathrm{d}X$.

Definition A scalar positive Itô process Y_t with the property that the product $Y_t X_t$ has zero drift is called **state-price process** or *pricing kernel* or *deflator* for X_t.
The importance of state-price processes is highlighted by the following theorem.

Theorem *Assume that for X_t a state-price process Y_t exists, b is self-financing, and $Yb^{tr}X$ is bounded below. Then*

(a) $Yb^{tr}X$ is a martingale, and
(b) the market does not admit self-financing arbitrage strategies.

Sketch of Proof (see [290], p.148)

(a) Y is a state-price process, hence there exists σ such that $\mathrm{d}(Y_t X_t) = \sigma \, \mathrm{d}W_t$ (zero drift). By Itô's lemma,

$$\mathrm{d}(Yb^{tr}X) = Y \, \mathrm{d}(b^{tr}X) + \mathrm{d}Yb^{tr}X + \mathrm{d}Y \, \mathrm{d}(b^{tr}X) \,.$$

Equation (B.18) and the self-financing property imply

$$\begin{aligned} \mathrm{d}(Yb^{tr}X) &= Yb^{tr}\,\mathrm{d}X + \mathrm{d}Yb^{tr}X + \mathrm{d}Yb^{tr}\,\mathrm{d}X \\ &= b^{tr}[Y\,\mathrm{d}X + \mathrm{d}YX + \mathrm{d}Y\,\mathrm{d}X] \\ &= b^{tr}\,\mathrm{d}(XY) = b^{tr}\sigma\,\mathrm{d}W =: \hat{\sigma}\,\mathrm{d}W\,, \end{aligned}$$

hence zero drift of $Yb^{tr}X$.
It remains to show that $Yb^{tr}X$ is a martingale.
Because of the boundedness, $\tilde{Z} := Yb^{tr}X - c$ is a positive scalar Itô process for some c, with zero drift. For every such process there is a $\tilde{\gamma}$ such that $\tilde{Z}$ has the form

$$\tilde{Z}_t = \tilde{Z}_0 Z_t^{\tilde{\gamma}}\,.$$

Hence $Yb^{tr}X = \tilde{Z} + c$ has the same properties as $Z^{\tilde{\gamma}}$, namely, it is a supermartingale. The final step is to show $\mathsf{E}(Z_t) =$constant. Now Q is defined via (B.23). (The last arguments are from martingale theory.)

(b) Assume arbitrage in the sense

$$\begin{aligned} &b_0^{tr}X_0 = 0\,,\ \mathsf{P}(b_t^{tr}X_t \geq 0) = 1\,, \\ &\mathsf{P}(b_t^{tr}X_t > 0) > 0 \quad \text{for some fixed } t\,. \end{aligned}$$

For that t:

$$b^{tr}X > 0 \quad \Rightarrow \quad Yb^{tr}X > 0\,.$$

Now $\mathsf{E}_{\mathsf{Q}}(Yb^{tr}X) > 0$ is intuitive. This amounts to

$$\mathsf{E}_{\mathsf{Q}}(Yb^{tr}X \mid \mathcal{F}_0) > 0\,.$$

Because $Yb^{tr}X$ is a martingale, $Y_0 b_0^{tr}X_0 > 0$ follows. This contradicts $b_0^{tr}X_0 = 0$, so the market is free of arbitrage.

Existence of a State-Price Process
In order to discuss the existence of a state-price process we investigate the drift term of the product Y_tX_t. To this end take X as satisfying the vector SDE

$$\mathrm{d}X = \mu^X\,\mathrm{d}t + \sigma^X\,\mathrm{d}W\,.$$

The coefficient functions μ^X and σ^X may vary with X. If no confusion arises, we drop the superscript X. Recall ($\longrightarrow$ Exercise 1.24) that each scalar positive Itô process must satisfy

$$\mathrm{d}Y = Y\alpha\,\mathrm{d}t + Y\beta\,\mathrm{d}W$$

for some α and β, where β and W can be vectors (β a one-row matrix). Without loss of generality, take the SDE for Y in the form

$$\mathrm{d}Y = -rY\,\mathrm{d}t - Y\gamma\,\mathrm{d}W\,. \tag{B.25}$$

(We leave the choice of the one-row matrix γ still open.) Itô's lemma (B.17) allows to calculate the drift of YX. By (B.18) the result is the vector

$$Y(\mu - rX - \sigma\gamma^{tr})\,.$$

Hence Y is a state-price process for X if and only if

$$\mu^X - rX = \sigma^X\gamma^{tr} \tag{B.26}$$

holds. This is a system of n equations for the m components of γ.

Special Case Geometric Brownian Motion For scalar $X = S$ and W, $\mu^X = \mu S$, $\sigma^X = \sigma S$, (B.26) reduces to

$$\mu - r = \sigma\gamma\,.$$

Given $\mu, \sigma \neq 0, r$, the Eq. (B.26) determines γ. (As explained in Sect. 1.7.3, γ is called the market price of risk.)
Discussion whether (B.26) admits a (unique) solution:

Case I: unique solution γ, and hence a unique state-price process.
Case II: multiple solutions: no arbitrage, but there are contingent claims that cannot be hedged.
Case III: no solution: The market admits arbitrage.

A market is said to be *complete*, if there is a unique martingale measure (Case I). This is equivalent to the statement that any contingent claim can be replicated with a self-financing portfolio of traded assets. Otherwise the market is called incomplete. As seen in Appendix A.4, the Black–Scholes market is complete, its price is unique. Models with jump processes are incomplete.

A solution of (B.26) for full rank of the matrix σ is given by

$$\gamma^* := (\mu - rX)^{tr}(\sigma\sigma^{tr})^{-1}\sigma\,,$$

which satisfies minimal length $\gamma^*\gamma^{*tr} \leq \gamma\gamma^{tr}$ for any other solution γ of (B.26), see [290]. Note that (B.26) provides zero drift of YX but is not sufficient for YX to be a martingale. But it is "almost" a martingale; a small additional condition suffices.

Those trading strategies b are said to be *admissible* if $Yb^{tr}X$ is a martingale.[3] There is ample literature on these topics; we just name [31, 84, 110, 282, 290, 314].

Application: Derivative Pricing Formula for European Options
Let X_t be a vector price process, and b a self-financing trading strategy such that a European claim C is replicated. That is, for $V_t = b_t^{tr} X_t$ the payoff is reached: $V_T = b_T^{tr} X_T = C$. (Compare Appendix A.4 for this argument.) We conclude from the above Theorem and from (B.21)

$$Y_t b_t^{tr} X_t = \mathsf{E}_{\mathsf{Q}}(Y_T b_T^{tr} X_T \mid \mathcal{F}_t) \,,$$

or

$$V_t = \frac{1}{Y_t} \mathsf{E}_{\mathsf{Q}}(Y_T C \mid \mathcal{F}_t) \,.$$

Specifically for the Black–Scholes model with $C = \Psi(S_T)$, the relation $\mathsf{E}_{\mathsf{Q}}(Y_T C \mid \mathcal{F}_0) = \mathsf{E}_{\mathsf{Q}}(Y_T C)$ holds, see [237, p. 69], or [193, p. 136]. Hence the value of European options is

$$V_0 = \frac{1}{Y_0} \mathsf{E}_{\mathsf{Q}}(Y_T C) \,.$$

This result is basic for Monte Carlo simulation, compare Sect. 3.5.1. Y_t represents a discounting process, for example, e^{-rt}. (Other discounting processes are possible, as long as they are tradable. They are called *numeraires*.) For a variable interest rate r_s,

$$V_t = \mathsf{E}_{\mathsf{Q}}(\exp(-\int_t^T r_s \, \mathrm{d}s) C \mid \mathcal{F}_t) \,.$$

In the special case r and γ constant, $Z_t = \exp(-\frac{1}{2}\gamma^2 t - \gamma W_t)$ and

$$\frac{V(t)}{\mathrm{e}^{rt}} = \mathsf{E}_{\mathsf{Q}} \left(\frac{C}{\mathrm{e}^{rT}} \mid \mathcal{F}_t \right) ,$$

from which

$$V(t) = \mathrm{e}^{-r(T-t)} \mathsf{E}_{\mathsf{Q}}(C \mid \mathcal{F}_t)$$

follows.

[3] Sufficient is that $Yb^{tr}X$ be bounded below, such that it can not become arbitrarily negative. This rules out the "doubling strategy." For our purpose, we may consider the criterion as technical. Glasserman [155] on p. 551: "It is common in applied work to assume that" a solution to an SDE with no drift term is a martingale.

Appendix C
Numerical Methods

C.1 Basic Numerical Tools

This appendix briefly describes numerical methods used in this text. For additional information and detailed discussion we refer to the literature, for example to [157, 168, 306, 308, 334, 347].

Condition
Suppose a function $f(x)$ is to be evaluated. When a small change Δx in x produces a large change Δf in f, we call the evaluation of f an *ill-conditioned* problem. This characterization expressing low opinion is justified in case the changes represent errors. Taylor expansion

$$f(x+\Delta x) = f(x) + f'(x)\Delta x + \frac{1}{2!}f''(x)\Delta x^2 + O(\Delta x^3)$$

leads to

$$\Delta f = \frac{\mathrm{d}f(x)}{\mathrm{d}x}\Delta x + O(\Delta x^2)\,.$$

Hence the derivative $\frac{\mathrm{d}f(x)}{\mathrm{d}x}$ is the amplification factor of Δx, also called the *absolute condition number*. Accuracy in the sense of correct digits is measured by the relative errors

$$\epsilon_x := \frac{\Delta x}{x}\,,\quad \epsilon_f := \frac{\Delta f}{f}\,.$$

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

From the above we obtain the amplification factor in the relative changes, with

$$\epsilon_f \approx \frac{\mathrm{d}f(x)}{\mathrm{d}x}\frac{x}{f}\,\epsilon_x\,.$$

In terms of error analysis, small condition numbers are desirable. But there are applications where a large value is welcome.

Example C.1 (Leverage) Let $V(S)$ denote the price of an option with underlying S. The number

$$l := \frac{\partial V}{\partial S}\frac{S}{V}\,, \qquad \text{with } \epsilon_V \approx l \cdot \epsilon_S$$

measures how much a rise of ϵ_S percent in S is amplified to a rise of ϵ_V percent in V. (Here a large value of the factor l will not be judged as "ill.") In our context, this relative condition number l is called *leverage*. Notice that "Delta" $= \frac{\partial V}{\partial S}$ is a factor in the leverage.

Interpolation
Suppose $n+1$ pairs of numbers $(x_i, y_i),\ i = 0, 1, \ldots, n$ are given, x_i ordered by magnitude, with $x_i \neq x_j$ for $i \neq j$. These points in the (x, y)-plane are to be connected by a curve. An interpolating function $\Phi(x)$ satisfies

$$\Phi(x_i) = y_i \quad \text{for } i = 0, 1, \ldots, n\,.$$

Depending on the choice of the class of functions Φ we distinguish different types of interpolation. A prominent example is furnished by polynomials,

$$\Phi(x) = P_n(x) = a_0 + a_1 x + \ldots + a_n x^n\ ;$$

the degree n matches the number $n+1$ of points. The evaluation of a polynomial is done by the *nested multiplication* given by

$$P_n(x) = (\ldots((a_n x + a_{n-1})x + a_{n-2})x + \ldots + a_1)x + a_0\ ,$$

which is also called *Horner's method*. An approach of polynomial interpolation is based on the *Lagrange polynomials*

$$L_k(x) := \prod_{\substack{i=0\\ i\neq k}}^{n} \frac{x - x_i}{x_k - x_i}\ ,$$

for $k = 0, \ldots, n$. By construction, the $L_k(x)$ are of degree n, and $L_k(x_k) = 1$, $L_k(x_i) = 0$ for $i \neq k$. Clearly, the polynomial

$$P(x) := L_0(x)y_0 + \ldots + L_n(x)y_n$$

interpolates $P(x_i) = y_i$ for $i = 0, \ldots, n$. To calculate $P(x)$ for a given x, use Neville's algorithm.

In case many points are given, the interpolation with one polynomial is generally not advisable since the high degree goes along with strong oscillations. A piecewise approach is preferred where low-degree polynomials are defined locally on one or more subintervals $x_i \le x \le x_{i+1}$ such that globally certain smoothness requirements are met. The simplest example is obtained when the points (x_i, y_i) are joined by straight-line segments in the order $x_0 < x_1 < \ldots < x_n$. The resulting *polygon* is globally continuous and linear over each subinterval. For the error of polygon approximation of a function we refer to Lemma 5.13. A C^2-smooth interpolation is given by the cubic *spline* using locally defined third-degree polynomials

$$S_i(x) := a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3 \quad \text{for } x_i \le x < x_{i+1}$$

that interpolate the points and are $\mathcal{C}^2$-smooth at the nodes x_i.

Interpolation is applied for graphical illustration, numerical integration, and for solving differential equations. Generally interpolation is used to approximate functions.

Rational Approximation
Rational approximation is based on

$$\Phi(x) = \frac{a_0 + a_1 x + \ldots + a_n x^n}{b_0 + b_1 x + \ldots + b_m x^m} \,. \tag{C.1}$$

Rational functions are advantageous in that they can approximate functions with poles. If the function that is to be approximated has a pole at $x = \xi$, then ξ must be zero of the denominator of Φ.

Quadrature
Approximating the definite integral

$$\int_a^b f(x)\,\mathrm{d}x$$

is a classic problem of numerical analysis. Simple approaches replace the integral by

$$\int_a^b P_m(x)\,\mathrm{d}x \,,$$

where the polynomial P_m approximates the function f. The resulting formulas are called *quadrature* formulas. For example, an equidistant partition of the interval $[a, b]$ into m subintervals defines nodes x_i and support points $(x_i, f(x_i))$, $i = 0, \ldots, m$ for interpolation. After integrating the resulting polynomial $P_m(x)$, the *Newton-Cotes formulas* result. The simplest case $m = 1$ defines the *trapezoidal rule*. We note in passing that the trapezoidal rule is also applied to differential equations

$\dot{y} = f(t, y)$. Derived from their equivalent integral equation, the discretized step

$$y(t + h) = y(t) + \frac{h}{2} [f(t, y(t)) + f(t + h, y(t + h))]$$

results.

For quadrature, a partition of the interval can be used favorably. Applying the trapezoidal rule in each of n subintervals of length

$$h := \frac{b - a}{n}$$

leads to the composite formula of the *composite trapezoidal sum*

$$T(h) := h \left[\frac{f(a)}{2} + f(a + h) + \ldots + f(b - h) + \frac{f(b)}{2} \right] . \tag{C.2}$$

The error of $T(h)$ satisfies a quadratic expansion

$$T(h) = \int_a^b f(x)\, \mathrm{d}x + c_1 h^2 + c_2 h^4 + \ldots ,$$

with a number of terms depending on the differentiability of f, and with constants $c_1, c_2, \ldots$ independent of h. This asymptotic expansion is fundamental for the high accuracy that can be achieved by *extrapolation*. Extrapolation evaluates $T(h)$ for a few h, for example, obtained by h_0, $h_1 = \frac{h_0}{2}$, $h_i = \frac{h_{i-1}}{2}$. Based on the values $T_i := T(h_i)$, an interpolating polynomial $\tilde{T}(h^2)$ is calculated with $\widetilde{T}(0)$ serving as approximation to the exact value $T(0)$ of the integral.

For $f \in \mathcal{C}^2[a, b]$, the error behavior reflected by the above expansion can be simplified to

$$\left| T(h) - \int_a^b f(x)\, \mathrm{d}x \right| \leq c\, h^2 ,$$

with a constant c, which depends on f''. That is, written with the Landau symbol O:

$$\text{The error is of the order } O(h^2) .$$

Zeros of Functions

The aim is to calculate a zero x^* of a function $f(x)$. An approximation is constructed in an iterative manner. Starting from some suitable initial guess x_0 a sequence

$x_1, x_2, \ldots$ is calculated such that the sequence converges to x^*. The approach is represented by Newton's method, which calculates the iterates by

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

for $k = 0, 1, 2, \ldots$ In the vector case a system of linear equations needs to be solved in each step,

$$Df(x_k)(x_{k+1} - x_k) = -f(x_k)\,, \tag{C.3}$$

where Df denotes the Jacobian matrix of all first-order partial derivatives.

Example C.2 (Yield to Maturity) Suppose a 3-year bond with a principal of \$100 that pays a 6% coupon annually. Further assume zero rates of 5.8% for the first year, 6.3% for a 2-year investment, and 6.4% for the 3-year maturity. Then the *present value* (sum of all discounted future cashflows) is

$$6\,\mathrm{e}^{-0.058} + 6\,\mathrm{e}^{-0.063*2} + 106\,\mathrm{e}^{-0.064*3} = 98.434$$

The *yield to maturity* (YTM) is the percentage rate of return y of the bond, when it is bought for the present value and is held to maturity. The YTM for the above example is the zero y of the cubic equation

$$0 = 98.434 - 6\,\mathrm{e}^{-y} - 6\,\mathrm{e}^{-2y} - 106\,\mathrm{e}^{-3y}$$

which is 0.06384, or 6.384%, obtained with one iteration of Newton's method (C.3), when started with 0.06 .

Convergence
There are modifications and alternatives to Newton's method. Different methods are distinguished by their convergence speed. Note that convergence is not guaranteed for any arbitrary choice of x_0. In the scalar case, *bisection* is a safe but slowly converging method. Newton's method for sufficiently regular problems shows fast convergence *locally*. That is, the error decays quadratically in a neighborhood of x^*,

$$\|x_{k+1} - x^*\| \le C\|x_k - x^*\|^p \quad \text{for } p = 2$$

for some constant C. This holds for an arbitrary vector norm $\|x\|$ such as

$$\begin{aligned} \|x\|_2 &:= \left(\sum_i x_i^2\right)^{1/2} \quad \textit{(Euclidian norm)} \\ \|x\|_\infty &:= \max_i |x_i| \quad \textit{(maximum norm)}, \end{aligned} \tag{C.4}$$

$i = 1, \ldots, n$ for $x \in \mathbb{R}^n$.

The derivative $f'(x_k)$ can be approximated by difference quotients. If the difference quotient is based on $f(x_k)$ and $f(x_{k-1})$, in the scalar case, the *secant method*

$$x_{k+1} = x_k - \frac{x_k - x_{k-1}}{f(x_k) - f(x_{k-1})} f(x_k) \tag{C.5}$$

results. It requires two initial guesses x_0 and x_1 to start the iteration. The secant method is generally faster than Newton's method if the speed is measured with respect to costs in evaluating $f(x)$ or $f'(x)$.

Gerschgorin's Theorem
A criterion for localizing the eigenvalues of a matrix A with elements a_{ij}, $i,j = 1,\ldots,n$ is given by Gerschgorin's theorem: Each eigenvalue lies in the union of the discs

$$\mathcal{D}_j := \{\, z \text{ complex and } |z - a_{jj}| \le \sum_{\substack{k=1 \\ k \neq j}}^{n} |a_{jk}| \,\}$$

($j = 1,\ldots,n$). The centers of the discs $\mathcal{D}_j$ are the diagonal elements of A, and the radii are given by the off-diagonal row sums (absolute values).

Triangular Decomposition
Let L denote a lower-triangular matrix (where the elements l_{ij} satisfy $l_{ij} = 0$ for $i < j$) and R an upper-triangular matrix (with elements $r_{ij} = 0$ for $i > j$), and the diagonal elements of L satisfy $l_{11} = \ldots = l_{nn} = 1$. Matrices A, L, R are supposed to be of size $n \times n$ and vectors x, b, ... have n components. Frequently, numerical methods must solve systems of linear equations

$$Ax = b\,.$$

A well-known direct method to solve this system is Gaussian elimination. First, in a "forward"-phase, an equivalent system

$$Rx = \widehat{b}$$

is calculated. Then, in a "backward"-phase starting with the last component x_n, all components of x are calculated one by one in the order $x_n, x_{n-1}, \ldots, x_1$. Gaussian elimination requires $\frac{2}{3}n^3 + O(n^2)$ arithmetic operations for full matrices A. With this count of $O(n^3)$, Gaussian elimination must be considered as an expensive endeavor, and is prohibitive for large values of n. (For alternatives, see iterative methods below in Appendix C.2.) The forward phase of Gaussian elimination is equivalent to an *LR-decomposition*. This means the factorization of the matrix A into the product of two triangular matrices L, R in the form

$$PA = LR\,.$$

Here P is a permutation matrix arranging for an exchange of rows that corresponds to the pivoting of the Gaussian algorithm. The LR-decomposition exists for all nonsingular A. After the LR-decomposition is calculated, only two equations with triangular matrices need to be solved,

$$Ly = Pb \quad \text{and} \quad Rx = y\,.$$

Tridiagonal Matrices

For tridiagonal matrices the LR-decomposition specializes to an algorithm that requires only $O(n)$ operations, which is inexpensive. Since several of the matrices in this book are tridiagonal, we include the algorithm. Let the tridiagonal system $Ax = b$ be in the form

$$\begin{pmatrix} \alpha_1 & \beta_1 & & & 0 \\ \gamma_2 & \alpha_2 & \beta_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \gamma_{n-1} & \alpha_{n-1} & \beta_{n-1} \\ 0 & & & \gamma_n & \alpha_n \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_{n-1} \\ x_n \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_{n-1} \\ b_n \end{pmatrix}. \tag{C.6}$$

Starting the Gaussian elimination with the first row to produce zeros in the subdiagonal during a forward loop, the algorithm is as follows:

Algorithm C.3 (Equation with Tridiagonal System)

$\hat{\alpha}_1 := \alpha_1,\ \hat{b}_1 := b_1$
(forward loop) for $i = 2, \ldots, n$*:*
$\quad \hat{\alpha}_i = \alpha_i - \beta_{i-1}\dfrac{\gamma_i}{\hat{\alpha}_{i-1}}, \quad \hat{b}_i = b_i - \hat{b}_{i-1}\dfrac{\gamma_i}{\hat{\alpha}_{i-1}}$
$x_n := \dfrac{\hat{b}_n}{\hat{\alpha}_n}$
(backward loop) for $i = n-1, \ldots, 1$*:*
$\quad x_i = \dfrac{1}{\hat{\alpha}_i}(\hat{b}_i - \beta_i x_{i+1})$

Here the "new" elements of the equivalent triangular system are indicated with a "hat;" the necessary checks for nonsingularity ($\hat{\alpha}_{i-1} \neq 0$) are omitted. The Algorithm C.3 needs about $8n$ operations. If one would start Gaussian elimination from the last row and produces zeros in the superdiagonal, an RL-decomposition results. The reader may wish to formulate the related backward/forward algorithm as an exercise.

Cholesky Decomposition

A real matrix A is called symmetric if $A^{tr} = A$, and is called *positive definite*, if $x^{tr}Ax > 0$ for all $x \neq 0$. For symmetric positive definite matrices there is exactly one lower-triangular matrix L with positive diagonal elements such that

$$A = LL^{tr}\,.$$

Here the diagonal elements of L are not normalized. For a computer program of Cholesky decomposition see [306].

Power Method
Assume an $(n \times n)$-matrix A with eigenvalues λ_j satisfying

$$|\lambda_1| > |\lambda_2| \geq \cdots \geq |\lambda_n|.$$

Then λ_1 is called dominant eigenvalue. Its eigenvector v (i.e., $Av = \lambda_1 v$ and $v \neq 0$) can be approximated iteratively by the *power method*: Start from any initial vector $x^{(0)} \neq 0$ and iterate for $k = 0, 1, 2, \ldots$

$$x^{(k+1)} := \frac{z}{\|z\|}, \quad \text{where } z := Ax^{(k)}$$

for any vector norm $\| \ \|$. The vectors $x^{(k)}$ converge towards v for $k \to \infty$, and the quotients $x_j^{(k+1)}/x_j^{(k)}$ for any index j such that $x_j^{(k)} \neq 0$ converge to λ_1. A general method for calculating all eigenvalues of a matrix is the QR-algorithm.

Fast Fourier Transform
A powerful tool is the Fast Fourier Transform (FFT). It transforms two strings of complex numbers onto each other,

$$g_0, \ldots, g_{n-1} \quad \longleftrightarrow \quad c_0, \ldots, c_{n-1}\,.$$

Typically n is large. FFT is based on the equivalence

$$c_\nu = \frac{1}{n} \sum_{j=0}^{n-1} g_j \mathrm{e}^{-\mathrm{i}\nu j \frac{2\pi}{n}} \quad \Longleftrightarrow \quad g_j = \sum_{\nu=0}^{n-1} c_\nu \mathrm{e}^{\mathrm{i}\nu j \frac{2\pi}{n}} \tag{C.7}$$

for $\nu, j = 0, 1, \ldots, n-1$, and the imaginary unit i. The FFT algorithm succeeds in $O(n \log n)$ operations, see [306].

C.2 Iterative Methods for $Ax = b$

The system of linear equations $Ax = b$ in $\mathbb{R}^n$ can be written

$$Mx = (M - A)x + b\,,$$

where M is a suitable matrix. For nonsingular M the system $Ax = b$ is equivalent to the fixed-point equation

$$x = (I - M^{-1}A)x + M^{-1}b\,,$$

which leads to the iteration

$$x^{(k+1)} = \underbrace{(I - M^{-1}A)}_{=:B} x^{(k)} + M^{-1}b\,. \tag{C.8}$$

The computation of $x^{(k+1)}$ is done by solving the system of equations $Mx^{(k+1)} = (M - A)x^{(k)} + b$. Subtracting the fixed-point equation and applying Lemma 4.2 shows

$$\text{convergence} \iff \rho(B) < 1\,;$$

$\rho(B)$ is the spectral radius of matrix B. For this convergence criterion there is a sufficient criterion that is easy to check. Natural matrix norms satisfy $\|B\| \geq \rho(B)$. Hence $\|B\| < 1$ implies convergence. Let b_{ij} denote the elements of B. Application to the matrix norms

$$\|B\|_\infty = \max_i \sum_{j=1}^{n} |b_{ij}|\,,$$

$$\|B\|_1 = \max_j \sum_{i=1}^{n} |b_{ij}|\,,$$

produces sufficient convergence criteria: The iteration converges if

$$\sum_{j=1}^{n} |b_{ij}| < 1 \quad \text{for } 1 \leq i \leq n$$

or if

$$\sum_{i=1}^{n} |b_{ij}| < 1 \quad \text{for } 1 \leq j \leq n\,.$$

By obvious reasons these criteria are called row sum criterion and column sum criterion. A *preconditioner* matrix M is constructed such that rapid convergence of (C.8) is achieved. Further, the structure of M must be simple so that the linear system is easily solved for $x^{(k+1)}$.

Simple examples are obtained by additive splitting of A into the form $A = D - L - U$, with

D diagonal matrix,
L strict lower-triangular matrix,
U strict upper-triangular matrix.

Jacobi's Method
Choosing $M := D$ implies $M - A = L + U$ and establishes the iteration

$$Dx^{(k+1)} = (L + U)x^{(k)} + b\,.$$

By the above convergence criteria a strict diagonal dominance of A is sufficient for the convergence of Jacobi's method.

Gauss–Seidel Method
Here the choice is $M := D - L$. This leads via $M - A = U$ to the iteration

$$(D - L)x^{(k+1)} = Ux^{(k)} + b\,.$$

SOR (Successive Overrelaxation)
The SOR method can be seen as a modification of the Gauss–Seidel method, where a *relaxation parameter* ω_{R} is introduced and chosen in a way that speeds up the convergence:

$$M := \frac{1}{\omega_{\mathrm{R}}}D - L \implies M - A = \left(\frac{1}{\omega_{\mathrm{R}}} - 1\right)D + U$$

$$\left(\frac{1}{\omega_{\mathrm{R}}}D - L\right)x^{(k+1)} = \left(\left(\frac{1}{\omega_{\mathrm{R}}} - 1\right)D + U\right)x^{(k)} + b$$

The SOR-method can be written as follows:

$$B_{\mathrm{R}} := \left(\frac{1}{\omega_{\mathrm{R}}}D - L\right)^{-1}\left(\left(\frac{1}{\omega_{\mathrm{R}}} - 1\right)D + U\right)$$

$$x^{(k+1)} = B_{\mathrm{R}}x^{(k)} + \left(\frac{1}{\omega_{\mathrm{R}}}D - L\right)^{-1} b$$

The Gauss–Seidel method is obtained as special case for $\omega_{\mathrm{R}} = 1$.

Choosing ω_{R}
The difference vectors $d^{(k+1)} := x^{(k+1)} - x^{(k)}$ satisfy

$$d^{(k+1)} = B_{\mathrm{R}}d^{(k)}\,. \tag{C.9}$$

This is the power method for eigenvalue problems. Hence the $d^{(k)}$ converge to the eigenvector of the dominant eigenvalue $\rho(B_{\mathrm{R}})$. Consequently, if (C.9) converges then

$$d^{(k+1)} = B_{\mathrm{R}}d^{(k)} \approx \rho(B_{\mathrm{R}})d^{(k)}\,,$$

and $|\rho(B_R)| \approx \frac{\|d^{(k+1)}\|}{\|d^{(k)}\|}$ for arbitrary vector norms. There is a class of matrices A with

$$\rho(B_{GS}) = (\rho(B_J))^2\,,\ B_J := D^{-1}(L+U)$$
$$\omega_{opt} = \frac{2}{1+\sqrt{1-\rho(B_J)^2}}\,,$$

see [347, 367]. Here B_J denotes the iteration matrix of the Jacobi method and B_{GS} that of the Gauss–Seidel method. For matrices A of that kind a few iterations with $\omega_R = 1$ suffice to estimate the value $\rho(B_{GS})$, which in turn gives an approximation to ω_{opt}. With our experience with Cryer's projected SOR applied to the valuation of options (Sect. 4.6.3) the simple strategy $\omega_R = 1$ is frequently recommendable.

This appendix has merely introduced classic iterative solvers, which are stationary in the sense that the preconditioner matrix M does not vary with k. For an overview on advanced nonstationary iterative methods see [29, 326].

C.3 Function Spaces

Let real-valued functions u, v, w be defined on $\mathcal{D} \subseteq \mathbb{R}^n$. We assume that $\mathcal{D}$ is a *domain*. That is, $\mathcal{D}$ is open, bounded and connected. The space of continuous functions is denoted $\mathcal{C}^0(\mathcal{D})$ or $\mathcal{C}(\mathcal{D})$. The functions in $\mathcal{C}^k(\mathcal{D})$ are k times continuously differentiable: All partial derivatives up to order k exist and are continuous on $\mathcal{D}$. The sets $\mathcal{C}^k(\mathcal{D})$ are examples of function spaces. Functions in $\mathcal{C}^k(\bar{\mathcal{D}})$ have in addition bounded and uniformly continuous derivatives and consequently can be extended to $\bar{\mathcal{D}}$.

Apart from being distinguished by differentiability, functions are also characterized by their integrability. The proper type of integral is the Lebesgue integral. The space of square-integrable functions is

$$\mathcal{L}^2(\mathcal{D}) := \left\{ v \mid \int_{\mathcal{D}} v^2 \,\mathrm{d}x < \infty \right\}\,. \tag{C.10}$$

For example, $v(x) = x^{-1/4} \in \mathcal{L}^2(0,1)$ but $v(x) = x^{-1/2} \notin \mathcal{L}^2(0,1)$. More general, for $p > 0$ the $\mathcal{L}^p$-spaces are defined by

$$\mathcal{L}^p(\mathcal{D}) := \left\{ v \mid \int_{\mathcal{D}} |v(x)|^p \,\mathrm{d}x < \infty \right\}\,.$$

For $p \geq 1$ these spaces have several important properties [3]. For example,

$$\|v\|_p := \left(\int_{\mathcal{D}} |v(x)|^p \,\mathrm{d}x \right)^{1/p} \tag{C.11}$$

is a norm.

In order to establish the existence of integrals such as

$$\int_a^b uv \, \mathrm{d}x, \quad \int_a^b u'v' \, \mathrm{d}x$$

we might be tempted to use a simple approach, defining a function space

$$\mathcal{H}^1(a,b) := \left\{ u \in \mathcal{L}^2(a,b) \mid u' \in \mathcal{L}^2(a,b) \right\}, \tag{C.12}$$

with $\mathcal{D} = (a,b)$. But a classic derivative u' may not exist for $u \in \mathcal{L}^2$ or needs not be square integrable. What is needed is a weaker notion of derivative.

Weak Derivatives
In $\mathcal{C}^k$-spaces classic derivatives are defined in the usual way. For $\mathcal{L}^2$-spaces *weak derivatives* are defined. For motivation let us review standard integration by parts

$$\int_a^b uv' \, \mathrm{d}x = -\int_a^b u'v \, \mathrm{d}x, \tag{C.13}$$

which is correct for all $u, v \in \mathcal{C}^1(a,b)$ with $v(a) = v(b) = 0$. For $u \notin \mathcal{C}^1$ the Eq. (C.13) can be used to define a weak derivative u' provided smoothness is transferred to v. For this purpose define

$$\mathcal{C}_0^\infty(\mathcal{D}) := \{ v \in \mathcal{C}^\infty(\mathcal{D}) \mid \text{ supp}(v) \text{ is a compact subset of } \mathcal{D} \} .$$

$v \in \mathcal{C}_0^\infty(\mathcal{D})$ implies $v = 0$ at the boundary of $\mathcal{D}$. For $\mathcal{D} \subseteq \mathbb{R}^n$ one uses the multi-index notation

$$\alpha := (\alpha_1, \ldots, \alpha_n), \quad \alpha_i \in \mathbb{N} \cup \{0\}$$

with

$$|\alpha| := \sum_{i=1}^n \alpha_i .$$

Then the partial derivative of order $|\alpha|$ is defined as

$$D^\alpha v := \frac{\partial^{|\alpha|}}{\partial x_1^{\alpha_1} \ldots \partial x_n^{\alpha_n}} v(x_1, \ldots, x_n) .$$

If a $w \in \mathcal{L}^2$ exists with

$$\int_{\mathcal{D}} u D^\alpha v \, \mathrm{d}x = (-1)^{|\alpha|} \int_{\mathcal{D}} wv \, \mathrm{d}x \quad \text{for all } v \in \mathcal{C}_0^\infty(\mathcal{D}) ,$$

the weak derivative of u with multi-index α is defined by $D^\alpha u := w$.

Sobolev Spaces
The definition (C.12) is meaningful if u' is considered as weak derivative in the above sense. More general, one defines the *Sobolev spaces*

$$\mathcal{H}^k(\mathcal{D}) := \left\{ v \in \mathcal{L}^2(\mathcal{D}) \mid D^\alpha v \in \mathcal{L}^2(\mathcal{D}) \text{ for } |\alpha| \le k \right\} . \tag{C.14}$$

The index $_0$ specifies the subspace of $\mathcal{H}^1$ that consists of those functions that vanish at the boundary of $\mathcal{D}$. For example,

$$\mathcal{H}_0^1(a,b) := \left\{ v \in \mathcal{H}^1(a,b) \mid v(a) = v(b) = 0 \right\} .$$

The Sobolev spaces $\mathcal{H}^k$ are equipped with the norm

$$\|v\|_k := \left(\sum_{|\alpha| \le k} \int_{\mathcal{D}} |D^\alpha v|^2 \, \mathrm{d}x \right)^{1/2} , \tag{C.15}$$

which is the sum of $\mathcal{L}^2$-norms of (C.11). For the special case discussed in Chap. 5 with $k = 1, n = 1, \mathcal{D} = (a,b)$, the norm is

$$\|v\|_1 := \left(\int_a^b (v^2 + (v')^2) \, \mathrm{d}x \right)^{1/2} .$$

Embedding theorems state which function spaces are subsets of other function spaces. In this way, elements of Sobolev spaces can be characterized and distinguished with respect to smoothness and integrability. For instance, the space $\mathcal{H}^1$ includes those functions that are globally continuous on all of $\mathcal{D}$ and its boundary and are *piecewise* $\mathcal{C}^1$-functions.

Hilbert Spaces
The function spaces $\mathcal{L}^2$ and $\mathcal{H}^k$ have numerous properties. Here we just mention that both spaces are *Hilbert spaces*. Hilbert spaces have an inner product $(\ ,\ )$ such that the space is complete with respect to the norm $\|v\| := \sqrt{(v,v)}$. In complete spaces every Cauchy sequence converges. In Hilbert spaces the *Schwarzian inequality*

$$|(u,v)| \le \|u\| \, \|v\| \tag{C.16}$$

holds. Examples of Hilbert spaces and their inner products are

$$\mathcal{L}^2(\mathcal{D}) \text{ with } (u,v)_0 := \int_{\mathcal{D}} u(x) v(x) \, \mathrm{d}x ,$$
$$\mathcal{H}^k(\mathcal{D}) \text{ with } (u,v)_k := \sum_{|\alpha| \le k} (D^\alpha u, D^\alpha v)_0 .$$

For further discussion of function spaces we refer, for instance, to [3, 162, 215, 377].

C.4 Minimization

Minimization methods are developed for a wide range of applications, including optimization under constraints or optimal control problems. Here we confine ourselves to a few introductory remarks on unconstrained minimization, setting the stage to solve a calibration problem. For general literature on minimization/optimization and parameter estimation refer, for example, to [306]. For the special application, curve fitting by least squares, see below.

In what follows, x is a vector in $\mathbb{R}^n$, and x^* a specific vector that minimizes a scalar function g locally,

$$g(x^*) \le g(x) \quad \text{for all } x \text{ in a neighborhood of } x^* .$$

A more ambitious task is to find a global minimum on the entire x-space. The vector x may represent n parameters of a model (c in Sect. 1.10), and g may stand for the least-squares function used for calibration, see (1.77). Since the methods of this appendix neglect possible constraints such as $x \ge 0$, we need to check x^* for feasibility after its calculation. For simplicity assume that at least one minimum exists.

A standard assumption of minimization methods is smoothness of g. Then, locally, the directional derivative in any direction $x - x^*$ is nonnegative,

$$(\operatorname{grad} g(x^*))^{tr}(x - x^*) \ge 0 .$$

In order to set up an iterative process to approach a minimum, one may look into the direction $-\operatorname{grad}(g(x))$ of steepest descent of g. This seems to be a convincing idea, but the steepest-descent method often requires a large number of iterations. A faster approach is obtained by invoking Newton's method. Recall that a necessary criterion for a minimum is the vanishing of all first-order partial derivatives,

$$\operatorname{grad} g(x^*) = 0 .$$

This suggests to apply a Newton-type method to search for a zero of

$$f(x) := \operatorname{grad} g(x) .$$

Then a sequence of iterates $x_1, x_2, \ldots$ is defined by (C.3),

$$H(x_k)(x_{k+1} - x_k) = -\operatorname{grad} g(x_k) , \tag{C.17}$$

where $H(x) = Df(x)$ denotes the Hesse matrix of all second-order partial derivatives of g,

$$H(x) = \begin{pmatrix} \frac{\partial^2 g}{\partial x_1 \partial x_1} & \cdots & \frac{\partial^2 g}{\partial x_1 \partial x_n} \\ \vdots & & \vdots \\ \frac{\partial^2 g}{\partial x_n \partial x_1} & \cdots & \frac{\partial^2 g}{\partial x_n \partial x_n} \end{pmatrix} .$$

The method defined by (C.17) is also called Gauss-Newton method. Locally, the convergence is fast, namely, of second order.

The evaluation of the Hessian $H(x)$ is cumbersome, in particular in finance, where g is not given explicitly and is approximated numerically. Therefore one resorts to cheaper approximations $\tilde{H}(x)$ of the Hessian. Such matrices $\tilde{H}$ are obtained by updates. The resulting method is then called quasi-Newton. One such approximation method is named BFGS,[1] see for example [61]. This Newton-type method of approximating x^* iteratively is a local method. The quality of the initial guess x_0 decides on how fast the convergence is, and to which local minimum the iteration goes. A combination of a steepest-descent method with a locally fast Newton-type method is provided by the Levenberg-Marquardt method, see [306].

When g is not smooth enough, or when differentiability is doubtful, or when g has many local minima, *simulated annealing* can be applied. This method works with random numbers searching the entire x-space. For references on simulated annealing see, for instance, [124, 223].

Frequently, a two-phase hybrid approach is applied. In a first phase the comparably slow simulated annealing is applied to single out globally candidates for minima. In the second phase these rough approximations are then used as initial vectors for the locally (fast) converging Newton-type method.

Another class of minimization methods is provided by genetic algorithms, where the minimum is approximated by constructing an evolution process. For applications to finance, see [34, 73].

Least Squares

Assume a set of N points

$$(x_k, y_k)\,,\ k = 1, \ldots, N,\ x_k \in \mathbb{R},\ y_k \in \mathbb{R}\,.$$

The aim is to construct a smooth curve $C(x)$ passing "nicely" through the cloud of points. This is the problem of *data fitting*, or *curve fitting*, and can be solved by simple linear algebra. Interpolation would not be the right answer when N is large. Rather one restricts the shape of C to be of a special kind. With $n+1$ free parameters

[1] After Broyden, Fletcher, Goldfarb, Shanno.

$a_0, \ldots, a_n$ and as many basis functions $\phi_0, \ldots, \phi_n$ we build C,

$$C(x) := \sum_{l=0}^{n} a_l \phi_l(x) \,.$$

In general, $n \ll N$. The simplest example is a polynomial,

$$C(x) = a_0 + a_1 x + \ldots + a_n x^n \,.$$

The basic strategy ("least squares") is to determine the parameters a_i such that the sum of squared differences between C and the data

$$\sum_{k=1}^{N} (C(x_k) - y_k)^2$$

gets minimal. Since the a's enter linearly in C, there is an $(N \times n)$-matrix A such that

$$A \begin{pmatrix} a_0 \\ \vdots \\ a_n \end{pmatrix} = \begin{pmatrix} C(x_1) \\ \vdots \\ C(x_N) \end{pmatrix} ,$$

and $\|Aa - y\|_2^2$ is minimal. Here we arrange the a's into a vector a, and the y's into a vector y, and use the norm from (C.4). The solution a of the least squares problem is that of the system of linear equations

$$A^{tr} A a = A^{tr} y \,,$$

and can be calculated via an orthogonal decomposition of A. Least squares is also called *regression*, or *best fit*.

C.5 Viscosity Solutions

For nonlinear problems, topics such as convergence are quite involved, in particular for nonsmooth solutions. We saw already for vanilla American options that solutions are not twice continuously differentiable. The nonlinearity of American-style options is a mild one, and rather straightforward numerical algorithms work (Chap. 4). But in general, nonlinear problems need not even have a unique solution. For motivation, let us look at the nonlinear PDE

$$\frac{\partial u}{\partial t} + \left| \frac{\partial u}{\partial x} \right| = 0 \quad \text{for } -\infty < x < \infty,\ t > 0$$

with initial condition $u(x,0) = |x|$, from [21]. This initial-value problem has two solutions,

$$u_1(x,t) = |x| - t\,, \text{ and}$$
$$u_2(x,t) = (|x| - t)^+\,.$$

Setting up a numerical scheme, the concern is to which of the two solutions the method will converge. (If it converges at all.)

This situation has lead to define a specific kind of weak solution, namely, the *viscosity solution*. For the above example, it can be shown that u_2 is the unique viscosity solution. Numerical methods can be set up that converge to a viscosity solution.

Assume (as in [21, 89]) a PDE that can be written as

$$H(x, u(x), \mathrm{D}u(x), \mathrm{D}^2u(x)) = 0\,,$$

where u is a scalar function, $\mathrm{D}u$ and D^2u correspond to first and second-order derivatives, and H is continuous. The notion of a viscosity solution requires the PDE to be *proper*, in the sense

$$H(x,u,p,A) \le H(x,v,p,A) \text{ for } u \le v \quad \text{(sign convention), and}$$
$$H(x,u,p,A) \le H(x,u,p,B) \text{ for } A \ge B \quad (\text{“}\textit{degenerate elliptic}\text{”})\,.$$

If we allow $x \in \mathcal{D} \subset \mathbb{R}^n$, $n > 1$, then $\mathrm{D}u$ represents the gradient and D^2u the Hesse matrix. The ellipticity means that H is nonincreasing in its second-order derivative matrix argument, which for scalar $q = A$ can also be written

$$H(x,u,p,q+\epsilon) \le H(x,u,p,q) \text{ for all } \epsilon \ge 0\,.$$

The first step towards the concept of a viscosity solution is to show that a classical (smooth) solution u can be characterized in an “unusual way” by comparing it to smooth test functions φ.

Theorem C.4 *Assume the PDE can be written $H(x,u,\mathrm{D}u,\mathrm{D}^2u) = 0$, $x \in \mathcal{D}$, with continuous and proper H. Then for $u \in C^2(\mathcal{D})$ the following is equivalent: u is (smooth) solution if and only if both criteria (a) and (b) hold:*

(a) All $\varphi \in C^2(\mathcal{D})$ with local minimum of $u - \varphi$ at x_0 satisfy

$$H(x_0, u(x_0), \mathrm{D}\varphi(x_0), \mathrm{D}^2\varphi(x_0)) \ge 0\,.$$

(b) All $\varphi \in C^2(\mathcal{D})$ with local maximum of $u - \varphi$ at x_0 satisfy

$$H(x_0, u(x_0), \mathrm{D}\varphi(x_0), \mathrm{D}^2\varphi(x_0)) \le 0\,.$$

Note that the above criteria (a) and (b) do *not* require the existence of first and second-order derivatives of u. Only $u \in C^0$ is used [for $u(x_0) = \varphi(x_0)$]. This situation suggests to define a weak solution u as follows.

Definition C.5 (Continuous Viscosity Solution) Let H be continuous and proper. Any continuous u ($u \in C^0(\mathcal{D})$) is called *continuous viscosity solution* of $H(x, u, \mathrm{D}u, \mathrm{D}^2u) = 0$ if and only if (a) and (b) are satisfied.

Example C.6 (Black–Scholes Equation) The Black–Scholes equation can be represented as above by an equation $H = 0$. To this end, set $x := (S, \tau)$, $u(x) := V(S, \tau)$, $p := \mathrm{D}u = (V_S, V_\tau)^{tr}$, $A := \mathrm{D}^2u$, and realize

$$\begin{aligned} H(x, u, p, A) :&= p^{tr} \begin{pmatrix} -rx_1 \\ 1 \end{pmatrix} - \frac{1}{2}\sigma^2 x_1^2 \begin{pmatrix} 1 \\ 0 \end{pmatrix}^{tr} A \begin{pmatrix} 1 \\ 0 \end{pmatrix} + ru \\ &= V_\tau - \frac{1}{2}\sigma^2 S^2 V_{SS} - rSV_S + rV\,. \end{aligned}$$

The sign convention holds for $r \geq 0$. For convenience rewrite H as $H(V, V_S, V_\tau, V_{SS})$. To check the ellipticity note that

$$\begin{aligned} H(u, y, z, q + \epsilon) &= z - \frac{1}{2}\sigma^2 S^2 (q + \epsilon) - rSy + ru \\ &= H(u, y, z, q) - \epsilon \frac{1}{2}\sigma^2 S^2 \ \leq H(u, y, z, q) \end{aligned}$$

holds for all $\epsilon \geq 0$ [134]. Hence H is proper.

Let V be a solution of $H(V, V_S, V_\tau, V_{SS}) = 0$, and $\varphi \in C^{2,1}$ be any test function with

$$V - \varphi \leq 0 \ \text{ and } \ \varphi(S_0, \tau_0) = V(S_0, \tau_0) \text{ for some } (S_0, \tau_0).$$

That is, at the point (S_0, τ_0) there is a local maximum of $f(S, \tau) := V(S, \tau) - \varphi(S, \tau)$. In case also $V \in C^{2,1}$, then the gradient vanishes,

$$\frac{\partial V(S_0, \tau_0)}{\partial S} = \frac{\partial \varphi(S_0, \tau_0)}{\partial S}\,, \quad \frac{\partial V(S_0, \tau_0)}{\partial \tau} = \frac{\partial \varphi(S_0, \tau_0)}{\partial \tau}\,,$$

and the Hessian is negative semidefinite, which specifically implies $f_{SS} \leq 0$, hence

$$\frac{\partial^2 V(S_0, \tau_0)}{\partial S^2} \leq \frac{\partial^2 \varphi(S_0, \tau_0)}{\partial S^2}\,.$$

For H this implies

$$H(V(S_0,\tau_0),\varphi_S(S_0,\tau_0),\varphi_\tau(S_0,\tau_0),\varphi_{SS}(S_0,\tau_0)) \le$$
$$H(V(S_0,\tau_0),\varphi_S(S_0,\tau_0),\varphi_\tau(S_0,\tau_0),V_{SS}(S_0,\tau_0)) = 0\,,$$

and criterion (b) holds. The analysis for (a) $V-\varphi \ge 0$ is analogous. Hence a classical solution V of the Black–Scholes equation is also a (continuous) viscosity solution.

Appendix D
Extended Tree Methods

Section 1.4 has introduced the basic version of of a binomial tree method for pricing options. This Appendix D discusses additional aspects of tree methods, such as convergence, discrete dividend payments, trinomial trees, and multidimensional trees.

D.1 Convergence to the Black–Scholes Formula

The standard binomial tree of Sect. 1.4 provides an approximation $V_0^{(M)}$

$$V_0^{(M)}(S_0) = \mathrm{e}^{-rT} \sum_{j=0}^{M} \binom{M}{j} p^j (1-p)^{M-j} \Psi(S_0) \tag{D.1}$$

to the value $V_0 := V(S_0, 0)$ of the continuous Black–Scholes model, where Ψ denotes the payoff of a European-style vanilla option. In a practical realization, for number of time levels $M \to \infty$, one observes convergence $V_0^{(M)} \to V_0$ with rates between $O(\frac{1}{\sqrt{M}})$ and $O(\frac{1}{M})$. The convergence is not monotonic. The slow rate $O(\frac{1}{\sqrt{M}})$ results from the central limit theorem, see Appendix B.1.[1] The $O(\frac{1}{M})$ convergence of the tilted tree of [244] was proved for European options, for American options see [242]. Notice that for the classical choice $ud = 1$ the parameters u, d, p of (1.18) are independent of S_0. Then the closed-form $V_0^{(M)}(S_0)$ of (D.1) is not smooth in S_0. For the tilted tree with the γ-strategy (1.23), $V_0^{(M)}(S_0)$ is smooth in S_0.[2]

[1] The proof is outlined in Exercise 1.10.

[2] This is illustrated in Topic 8 of *Topics fCF*.

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

Closeness to the Black–Scholes Equation

The above has discussed the convergence of $V_0^{(M)}$ to the value $V(S,0)$ of standard options. Other issues are the convergence for other payoffs, or the convergence of derivatives of V. Smoothness of the payoff and of the approximations are key problems. This appendix will only concentrate on a few aspects. We start with a result that the binomial scheme is close to the Black–Scholes equation.

A need to improve the smoothness can be motivated by studying an ideal setting: Assume $\tilde{V}$ to be a function $\in C^4$ satisfying the continuous version of the binomial formula (1.21)

$$\tilde{V}(S,t-\Delta t) = \mathrm{e}^{-r\Delta t}\left(p\tilde{V}(uS,t) + (1-p)\tilde{V}(dS,t)\right),$$

for any (S,t). After some routine calculations one obtains

$$\begin{aligned} 0 &= -\tilde{V}(S,t-\Delta t) + \mathrm{e}^{-r\Delta t}\left(p\tilde{V}(uS,t) + (1-p)\tilde{V}(dS,t)\right) \\ &= \left[\frac{\partial\tilde{V}}{\partial t} + rS\frac{\partial\tilde{V}}{\partial S} + \frac{1}{2}\sigma^2S^2\frac{\partial^2\tilde{V}}{\partial S^2} - r\tilde{V}\right]\Delta t + O(\Delta t^2), \end{aligned}$$

see Exercise 1.27. Dividing by Δt shows that the term in brackets vanishes up to terms order $O(\Delta t)$. This shows that the binomial formula is close to the Black–Scholes equation, up to terms of order $O(\Delta t)$ (see [234]).

[179] proves $O(\Delta t)$ convergence of $V^{(M)}$ for payoffs $\Psi \in C^2$, building on results known from the theory of PDEs. But the C^2-smoothness assumption is not satisfied at the boundary $t = T$ for vanilla calls or puts. For American-style options there is a second source of trouble: V is not smooth at the early-exercise curve. In summary, along the boundaries more effort is required to achieve better convergence.

Possibilities to Smooth the Payoff

As outlined in Sect. 1.4.5, a lack of smoothness may cause faulty oscillations. That is, the approximations $V_0^{(M)}$ for $V(S_0,0)$ show a wave-like behavior as the fineness M increases. Section 1.4.5 has recommended the tilted tree as a main vehicle to cope with the kink in the payoff, and to improve the convergence. In the following we describe two other approaches to smooth the effects of a non-smooth payoff Ψ.

1. Heston and Zhou [179] applies a smoothing approach of [231]. To this end, a payoff Ψ is approximated by a smoother version $\overline{\Psi}$,

$$\overline{\Psi}(S) := \frac{1}{2\xi}\int_{-\xi}^{\xi}\Psi(S-y)\,\mathrm{d}y\,, \tag{D.2}$$

for a choice of a small $\xi > 0$. At least for vanilla payoffs and binary options, $\overline{\Psi}$ can be calculated analytically. For the vanilla option, $\overline{\Psi} \in C^1$, and for a binary option, $\overline{\Psi} \in C^0$ ($\longrightarrow$ Exercise 1.12).

2. Above the last line of the tree, for $t > t_{M-1}$, an American option is always European-style, because for our discrete model exercising is only possible at the discrete times t_i. For many payoffs, there are analytic formulas for the European-style option [172]. Therefore the analytic formula holds on the strip $t_{M-1} < t < T$, and can be applied to obtain the continuation values at t_{M-1}. At t_{M-1}, the analytic formula does not have the singularity. In this way, a singularity of the payoff (at the strike K or elsewhere) is bypassed, and the continuation values at t_{M-1} are given by a smooth function. This remedy, suggested by [56], needs to be applied only for a few nodes about the singularity (about 10 for medium values of M), because farer away from the singularity (K) these continuation values are almost identical with the payoff. The main part of the binomial method then starts at t_{M-1}.

The γ-strategy of (1.23) can be combined with the second strategy of applying the BS-formula; but the additional improvement is small. For barrier options, the second strategy may be the best choice when the "symmetry" of the classic $ud = 1$ tree is advantageous for incorporating the barrier.

As a result of smoothing, dependable $O(\Delta t)$-convergence of $V_0^{(M)}$ can be observed for vanilla options. Then, from a practical point of view, the convergence of $V_0^{(M)}$ is close enough to $O(\frac{1}{M})$ to justify extrapolation (1.24). Then the convergence of $V^{(M,\mathrm{extr})}$ will be close to second-order $O(\Delta t^2)$. The additional $V^{(M/2)}$-approximation increases the costs only by about 25% .

For related literature, see also [305]. The aim "place a node at the strike" is handled in a somewhat more general way in [359], where any index can be chosen, not necessarily the $M/2$ middle node.

D.2 Discrete Dividend Payment

Assume the underlying asset pays a discrete dividend. If an amount D is paid at the discrete time t_D with $0 < t_D < T$, then by arbitrage there is a drop in the price S of the asset, which is basically[3] of that size:

$$S_{t_D^+} = S_{t_D^-} - D\,. \tag{D.3}$$

In contrast, for European options, the value V_t of an option on that asset remains continuous along S_t also at t_D, which for the value function amounts to

$$V(S_{t_D^-}, t_D^-) = V(S_{t_D^+}, t_D^+)\,.$$

[3] Because of taxes and fees, the jump is by the netto amount. So take as D the actual drop.

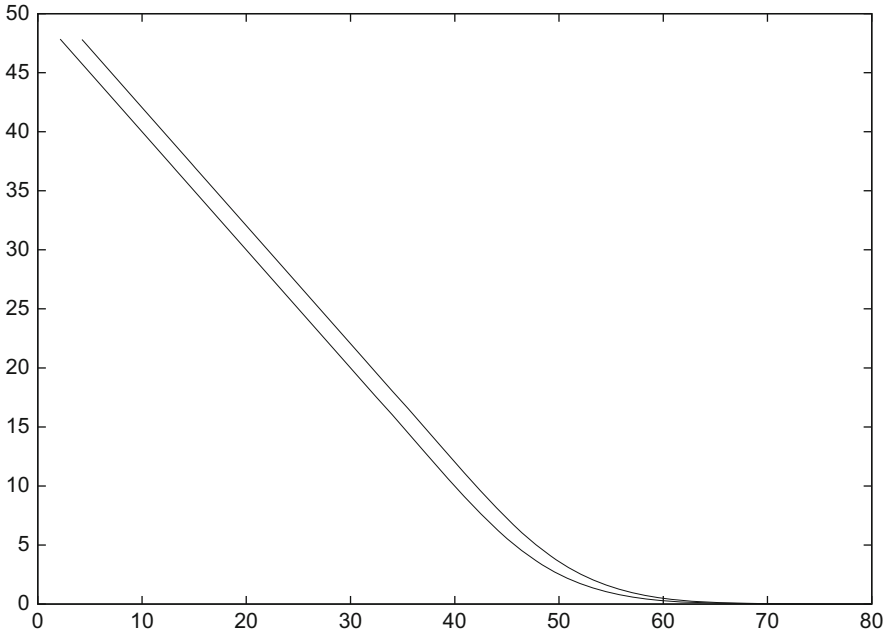

Fig. D.1 The gap in V, here for an American put. *Left-hand curve*: $V(S, t_D^+)$, *right-hand curve*: $V(S, t_D^-)$. Parameters: $K = 50$, $T = 5/12$, $r = 0.1$, $\sigma = 0.4$, $D = 2.06$, $T_D = 2.9166667$ (parameters as in [191])

This holds for any path S_t, hence

$$V(S, t_D^-) = V(S - D, t_D^+) \,. \tag{D.4}$$

The notation in (D.4) represents the left-sided and right-sided limits, as

$$V(S, t_D^-) := \lim_{t \to t^-, t < t^-} V(S, t) \,.$$

Passing t_D backwards from $t > t_D$ to $t < t_D$, Eq. (D.4) represents a "horizontal" shift by D of the profile $V(S, t_D^+)$ in S-direction, which amounts to a **jump condition** for $V(S, t)$ at t_D. The surface $V(S, t)$ is discontinuous at t_D. In Fig. D.1, the gap formed by the two limiting functions is shown for an American put, the horizontal shift (by D) is visible.[4]

Problem with Recombination

Let us consider the situation where D is a fixed amount independent of S. For the single dividend case, because of (D.3), the standard binomial tree is not

[4] To visualize the surface $V(S, t)$ and the gap, see Topic 5 in the *Topics fCF*.

recombining. To see this, assume for ease of demonstration $t_D = t_2$. In case no dividend is paid, for the t_2-line the three S-node values would be d^2S, udS, u^2S. The drop in asset price caused by the dividend payment D shifts these three node values to

$$d^2S - D, \quad udS - D, \quad u^2S - D\,.$$

Consequently, at the t_3-line, the tree splits to the values

$$d(d^2S - D),\ u(d^2S - D), \quad d(udS - D),\ u(udS - D), \quad d(u^2S - D),\ u(u^2S - D)\,.$$

Obviously, since $u \neq d$, and D is a constant value, these are six different nodes as compared to the four nodes in the no-dividend case. For the t_4-line, there are nine nodes rather than five without dividend. Clearly, this standard S-tree is no longer recombining, and the number of nodes increases more rapidly.

Exercise Let N_i denote the number of nodes of the standard binomial S-tree at t_i, and let t_k be the ex-dividend date. Show $N_{k+i} = (i+1)(k+1)$ for $i > 0$.

Numerical Approach for Binomial Trees
At time t with $0 < t < t_D$, the present value of the dividend is

$$\tilde{D}_t := D\,\mathrm{e}^{-r(t_D - t)}\mathbf{1}_{[0,t_D]}\,. \tag{D.5}$$

There may be several discrete dividend payments $D_1, \ldots, D_n$ paid at dates $t_1 < t_2 < \ldots < t_n$ with $0 < t_1, t_n < T$. Then the present value $\tilde{D}_t$ of all dividend payments is defined accordingly. For example, for $t < t_1$,

$$\tilde{D}_t = D_1\mathrm{e}^{-r(t_1 - t)} + \ldots + D_n\mathrm{e}^{-r(t_n - t)}\,,$$

and $\tilde{D}_t = 0$ for $t > t_n$. Since the portion $\tilde{D}_t$ of the asset price S_t is riskless, the complementary part

$$\tilde{S}_t := S_t - \tilde{D}_t \tag{D.6}$$

is the risky part. For illustration see Fig. D.2.

For ease of presentation, we stick to the case of a single dividend payment in the time interval, so $\tilde{S}_t = S_t - D\mathrm{e}^{-r(t_D - t)}$ for $t < t_D$. For the Black–Scholes model, the GBM for $\tilde{S}_t$ is described by

$$\frac{\mathrm{d}\tilde{S}}{\tilde{S}} = r\,\mathrm{d}t + \tilde{\sigma}\,\mathrm{d}W_t\,,$$

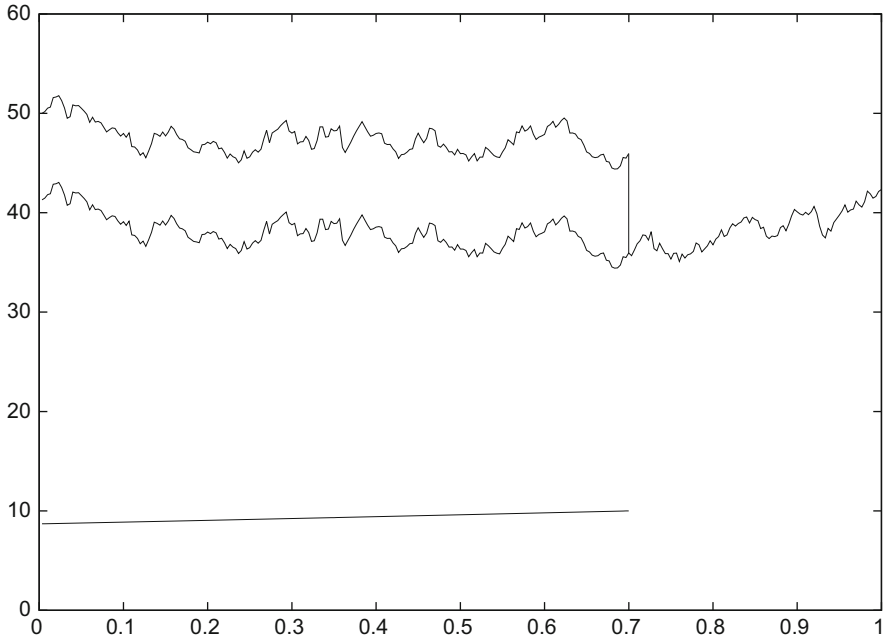

Fig. D.2 The price S_t of a fictive underlying with dividend payment of $D = 10$ at $t_D = 0.7$ (*upper curve*) ; *bottom curve*: $\widetilde{D}_t$ of (D.5) , "*middle*" *curve*: $\widetilde{S}_t$ of (D.7)

with $\tilde{\sigma}$ assumed constant, and we assume $\tilde{\sigma} = \sigma$.[5]

The key observation is that

$$\tilde{S}_t = \begin{cases} S_t & \text{for } t_D < t \le T \\ S_t - D\mathrm{e}^{-r(t_D - t)} & \text{for } 0 \le t < t_D \end{cases} \tag{D.7}$$

is *continuous across* t_D, because of (D.3). Hence the corresponding $\tilde{S}$-tree is a standard recombining tree. It is common for European options to evaluate the Black–Scholes formula with the asset price $\tilde{S}_0$ (the asset price reduced by the present value of the dividends). This is the value initiating the $\tilde{S}$-tree (Fig. D.3). S-values can be obtained out of the $\tilde{S}$-values via (D.7).

[5]The volatility $\tilde{\sigma}$ of $\widetilde{S}$ needs not be the same as that followed by the whole asset price. In general, $\tilde{\sigma}$ is slightly larger than σ, which might be realized by scaling σ with the artificial rule-of-thumb factor $\frac{S}{S - D\mathrm{e}^{-rt_D}}$. The scaling is not necessary in case σ is the implied volatility, then $\tilde{\sigma} = \sigma$ [191, 234]. Certainly, the assumption has an effect on the calculated price V.

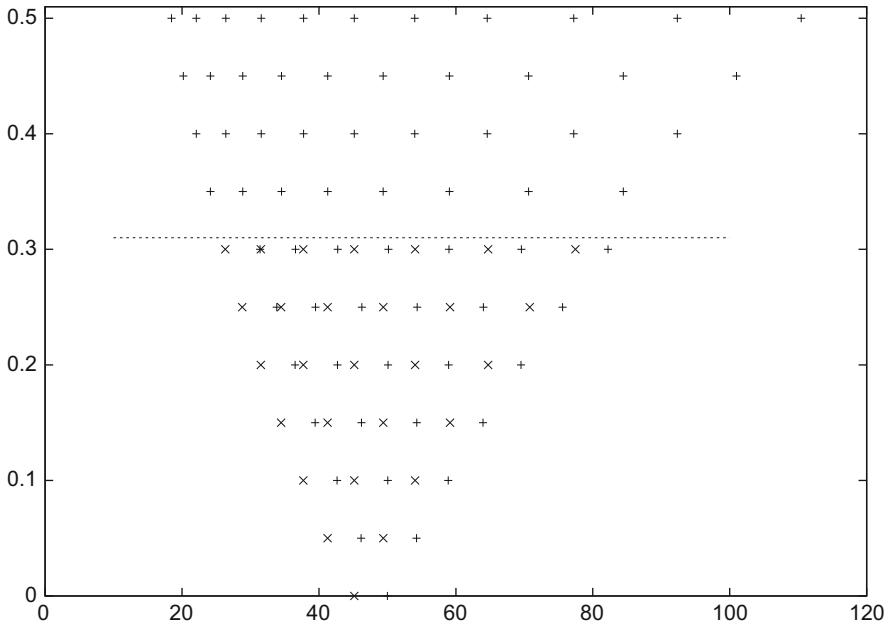

Fig. D.3 (S,t)-plane; tree for a dividend-paying underlying; data see Example D.2. + signs mark the S-tree; × signs mark the $\tilde{S}$-tree. The shift of the algorithm goes from × to +. $\tilde{S}_0 = 45.1526$

Algorithm D.1 (S-Tree for the Single-Dividend Case) *(fixed dividend payment D at t_D with k defined by $t_{k-1} \le t_D < t_k$)*

$$\tilde{S}_0 := S_0 - D\mathrm{e}^{-rt_D}\,,$$

$$S_{j,i} := \tilde{S}_0 u^j d^{i-j} + D\mathrm{e}^{-r(t_D - t_i)} \text{ for } j = 0, 1, \ldots, i \text{ and } 0 < i < k\,,$$

$$S_{j,i} := \tilde{S}_0 u^j d^{i-j} \text{ for } j = 0, 1, \ldots, i \text{ and } i \ge k\,.$$

The modified S-tree obtained this way is recombining, because the $\tilde{S}$-tree is, but not continuous. The valuation of the tree is the same as for the standard dividend-free tree; implementation is easy.[6] The case with several discrete dividend payments is set up analogously. For American-style options, the comparison of the continuation value with the payoff needs to be incorporated. In Algorithm D.1, the $\tilde{S}$-tree shows only implicitly. The algorithm was derived for $\tilde{S}$ and transformed back to S. The early-exercise check for an American option must be based on S.

[6] Program and test Algorithm D.1. For $m = 4$, plot the S-tree. For illustrations, see Topic 1 and Topic 5 of *Topics fCF*.

Example D.2 (Dividend Payment) We choose $T = 0.5$, $S_0 = 50$, $K = 55$, $r = 0.1$, $\sigma = 0.4$, a discretization with $M = 10$ time steps, and a discrete dividend payment of $D = 5$ at time $t_D = 0.31$. The S-tree is shown in the (S,t)-plane of Fig. D.3. The horizontal line (dashed) marks $t_D = 0.31$. Above that line, for $t > t_D$, the S-nodes are identical to those of the $\tilde{S}$-tree originating at $\tilde{S}_0$ (marked at the bottom). The "lower" part of that $\tilde{S}$-tree is indicated by $\times$ signs. Below the horizontal dashed line, for $t < t_D$, the S-nodes "+" are the shifted ones, obtained by (D.7) out of the $\tilde{S}$-tree with a shift approximately of the size $+D$.

Proportional Dividend
There is another discrete dividend arrangement, namely, a proportional dividend, $D = \tilde{q}S$ for a given rate $\tilde{q}$. As a result, for $q := 1 - \tilde{q}$,

$$S_{t_D^+} = S_{t_D^-} - \tilde{q}S_{t_D^-} = (1-\tilde{q})S_{t_D^-} = qS_{t_D^-}\,.$$

Then, illustrated similarly as above (assuming for simplicity $t_D = t_2$), the S-shifts at the t_2-line are not by a constant value. Rather the outcome is

$$qd^2S, \quad qudS, \quad qu^2S\,.$$

It is easy to check that there are only four nodes at the t_3-line. The tree of a discrete proportional dividend is recombining. (Check this by means of an appropriate illustration.)

American-Style Options
For the early exercising of a call, see [234], or [191]; for a call no specific numerical method is needed. The situation is more cumbersome for an American put. An analytical result on the early-exercise structure is Exercise 4.1b. For American-style puts the stopping region needs not be connected. At the ex-dividend date t_D, the early-exercise curve declines sharply to $S = 0$, cutting and bounding the part of the stopping region with $t \geq t_D$. For $t > t_D$ the early-exercise curve is identical to that of the no-dividend situation. For the time period

$$\max\{0,\tilde{t}\} \leq t < t_D \;\text{ with }\; \tilde{t} := t_D - \frac{1}{r}\log\left(\frac{D}{K}+1\right)$$

it is optimal to wait for the dividend and hold the option. In case $\tilde{t} > 0$, another part of the stopping region exists for $0 \leq t \leq \tilde{t}$, where early exercise is optimal.[7] In contrast, for a proportional dividend $D = \tilde{q}S$ there is a lower part of the stopping area extending to t_D, where the limiting early-exercise curve tends to $(S,t) = (0,t_D)$ with the slope

$$\lim_{t\to t_D,\, t<t_D} S_{\mathrm{f}}'(t) = -\frac{r}{\tilde{q}}\,.$$

[7] This is illustrated in Topic 1 of *Topics fCF*, see also Topic 5.

D.3 Trinomial Trees

Starting from a price S_i at t_i, the binomial tree allows for two states S_{i+1} at t_{i+1}, namely, uS_i and dS_i. Trinomial trees allow for three states

$$uS_i\,, \quad mS_i\,, \quad dS_i\,,$$

with corresponding probabilities p_1, p_2, p_3. That is, six unknown parameters need to be fixed, and six requirements are needed. Two of them are clear: Of course, $p_1 + p_2 + p_3 = 1$ makes one of the equations. Another equation should enforce the recombining property, which amounts to the requirement $m^2 = ud$.

Before we come to examples, let us study the structure of a trinomial tree, and discuss what accuracies we may expect. Certainly this will depend on the density of the tree. As may be visualized by means of a sketch, there are $2i + 1$ nodes for t_i. In particular, the payoff at $t = T$ is scanned at $2M + 1$ node values S in case the time interval is divided into M subintervals. In comparison, for a binomial tree, this density of nodes at T is reached with the *double* number of discrete t_i-values. So, naively, we might expect trinomial methods to reach a comparable accuracy in a more efficient way, since it needs fewer steps.

For a comparison, let us sum up the number of required arithmetic operations. We can neglect the overhead, which calculates parameters such as u or p in the beginning. For both binomial and trinomial trees, the number of operations is proportional to the number of nodes. A little reasoning leads to Table D.1. For a comparable resolution of the payoff, set $M = 2\tilde{M}$ for an equal number of nodes at $t = T$, and come up with $5\tilde{M}^2$ operations for the trinomial tree and $6\tilde{M}^2$ for the binomial tree. But empirical comparisons lead to the conclusion that smoothed binomial trees are significantly more efficient than trinomial trees! The above reasoning with placing emphasis on the number of steps is misleading because it neglects the fact that the binomial tree benefits from a smaller error due to the smaller Δt-step size.

But what then is the advantage of a trinomial method? The larger number of parameters adds to the *flexibility* of the trinomial tree. For example, it can be well aligned to barriers, even to double barriers.

Examples of Trinomial Trees

Possible sets of parameters must obey $p_i \geq 0$ (for $i = 1, 2, 3$), which leads to a limitation of the step size Δt. One choice of parameters assumes equal probabilities

Table D.1 Comparison binomial/trinomial tree method

Type of tree	Number of t_i-lines	Number of nodes	Number of operations per node	Total number of operations
Binomial	M	$\frac{1}{2}M^2 + \frac{3}{2}M + 1$	3	$\approx \frac{3}{2}M^2$
Trinomial	$\tilde{M}$	$\tilde{M}^2 + 2\tilde{M} + 1$	5	$\approx 5\tilde{M}^2$

[358], see Exercise 1.15. Another approach starts from $u = \mathrm{e}^{\sigma\sqrt{\lambda\Delta t}}$ with a free tune factor λ. One such example, taken from [191], is

$$\begin{aligned} u &= \mathrm{e}^{\sigma\sqrt{3\Delta t}}\,, \quad d = \tfrac{1}{u}\,, \quad m = 1\,, \\ p_1 &= \sqrt{\tfrac{\Delta t}{12\sigma^2}}\left(r - \tfrac{\sigma^2}{2}\right) + \tfrac{1}{6}\,, \\ p_2 &= \tfrac{2}{3}\,, \\ p_3 &= -\sqrt{\tfrac{\Delta t}{12\sigma^2}}\left(r - \tfrac{\sigma^2}{2}\right) + \tfrac{1}{6}\,. \end{aligned} \tag{D.8}$$

D.4 Multidimensional Trees

This subsection assumes an option on two underlying assets. The notation needs to be changed: $S_1(t)$, $S_2(t)$ are the prices of the two assets. And, to avoid overloaded indices, let us also use the notation $x := S_1$, $y := S_2$. Hence the price vectors $(S_1(t), S_2(t))$ for $0 \le t \le T$ are points in (x, y)-planes, one for each t. Accordingly, the nodes of the tree will be distributed in the three-dimensional (x, y, t)-space.[8]

Let t_ν denote the discrete t-values, with $\nu = 1, \ldots, M$ and $\Delta t = T/M$, $t_\nu = \nu\Delta t$. With index ν denoting the time level, the (x, y)-coordinates on the νth time level will be denoted x^ν and y^ν. Further assume a constant risk-free interest rate r, asset volatility parameters σ_1, σ_2, correlation ρ, and dividend yield rates δ_1, δ_2.

Similarly as for the basic tree, we have free choices to construct a multidimensional tree. Depending on the construction, there are four, five, or more outcomes out of each node. Here we discuss a tree with four states and preset probabilities, namely, the binomial-pyramid tree suggested by Rubinstein [323]. Four outcomes, each with a probability and two components, amount to 12 parameters. Choosing equal probabilities $p = \frac{1}{4}$, and admitting only two x-values, cuts the number of parameters down to six. That is, each node with position (x, y) develops for $t \to t + \Delta t$ to one of the four positions

$$(xu, yA),\ (xu, yB),\ (xd, yC),\ (xd, yD)\,, \tag{D.9}$$

for six parameters u, d, A, B, C, D (see Fig. D.4). Hence the x-coordinates of the tree are recombining as in the one-dimensional situation. The factors A, B, C, D control the y-coordinates. For t_1, let us arrange the y-factors as in an (x, y)-plane,

C A
D B

[8] Therefore such a tree for a two-color rainbow option is also called "three dimensional." Consult Topic 7 in case more insight is needed to visualize this setting.

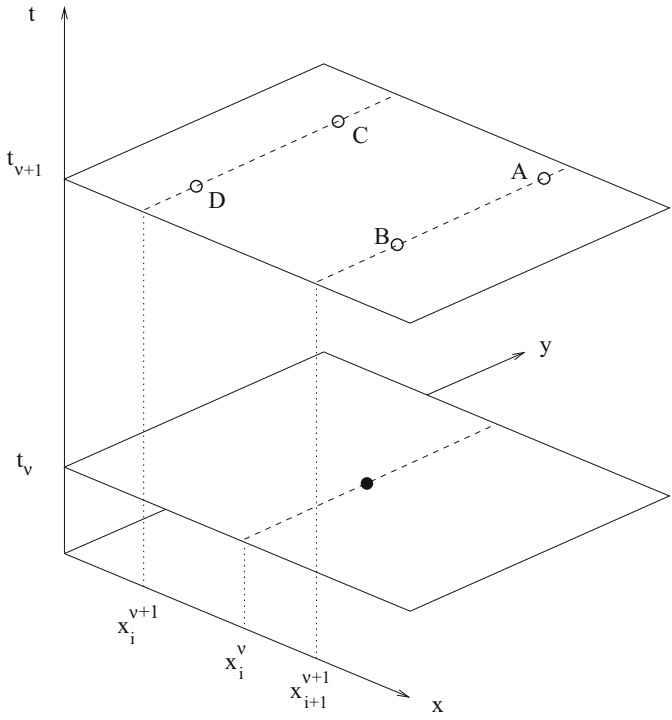

Fig. D.4 In the (x, y, t)-space: four factors A, B, C, D for the new y-coordinate in the $t_{\nu+1}$-time level

The left-hand factors correspond to $d{\cdot}x$, and the right-hand factors to $u{\cdot}x$ for an initial price x of S_1. At t_2, each of these four nodes splits again, namely, to y-coordinates

CC	CA	AC	AA
CD	CB	AD	AB
DC	DA	BC	BA
DD	DB	BD	BB

To be recombining, only nine nodes may survive for t_2, so the "middle" nodes must coincide. This is trivial for the four nodes on the "boundary" of these scheme ($CA = AC \ldots$). Only the center point needs special care: We require $AD = BC$ for the tree to be recombining.

Analogously as for the single-asset binomial tree let us assume for the bivariate case a joint lognormal distribution, see (6.1). As in Sect. 1.8.2, set $Y_1 := \log S_1 = \log x$ and $Y_2 := \log S_2 = \log y$. By (6.3),

$$\begin{aligned} \mathrm{d}Y_1 &= \mathrm{d}(\log x) = (r - \delta_1 - \tfrac{1}{2}\sigma_1^2)\,\mathrm{d}t + \sigma_1 \mathrm{d}\theta_1\,, \\ \mathrm{d}Y_2 &= \mathrm{d}(\log y) = (r - \delta_2 - \tfrac{1}{2}\sigma_2^2)\,\mathrm{d}t + \sigma_2(\mathrm{d}\theta_1\,\rho + \mathrm{d}\theta_2\,\sqrt{1-\rho^2})\,, \end{aligned} \tag{D.10}$$

where θ_1, θ_2 are independent normal variates, and the correlation ρ is taken care of. The continuous model (D.10) incorporates possible dividend yield rates δ_1, δ_2. For the setup of a tree, we discretize (D.10). Replacing the symbol d by the symbol Δ leads to

$$\begin{aligned} \Delta Y_1 &= \log x^{\nu+1} - \log x^{\nu} = \mu_1 \Delta t + \sigma_1 \Delta\theta_1 \\ \Delta Y_2 &= \log y^{\nu+1} - \log y^{\nu} = \mu_2 \Delta t + \sigma_2(\Delta\theta_1\, \rho + \Delta\theta_2 \sqrt{1-\rho^2}) \end{aligned} \tag{D.11}$$

with

$$\mu_i := r - \delta_i - \sigma_i^2/2 \;\text{ for } i = 1, 2\,.$$

Next we replace the $\Delta\theta$-variables by the simpler $\pm\sqrt{\Delta t}$ with equal probabilities for both signs,

$$\begin{aligned} \Delta Y_1 &= \mu_1 \Delta t \pm \sigma_1 \sqrt{\Delta t} \\ \Delta Y_2 &= \mu_2 \Delta t \pm \sigma_2 \rho \sqrt{\Delta t} \pm \sigma_2 \sqrt{1-\rho^2}\sqrt{\Delta t} \end{aligned} \tag{D.12}$$

This is the discrete problem.[9] The variables ΔY_1 and ΔY_2 of the discrete problem (D.12) satisfy the five additional equations coming from the requirement that the first and second moments of the discrete and the continuous problem must match. To verify this, check that for (D.12) the five equations

$$\mathsf{E}(\Delta Y_i) = \mu_i \Delta t\,, \quad \mathsf{Var}(\Delta Y_i) = \sigma_i^2 \Delta t\,, \quad \mathsf{Cov}(\Delta Y_1, \Delta Y_2) = \rho\sigma_1\sigma_2 \Delta t$$

hold for $i = 1, 2$ and the four probabilities $\frac{1}{4}$ associated to (D.9) ($\longrightarrow$ Exercise 6.2).

The factors u, d, A, B, C, D can be read off from (D.12). For example,

$$x^{\nu+1} = x^{\nu} \exp(\Delta Y_1) = x^{\nu} u \qquad \text{or} \qquad = x^{\nu} d\,,$$

which provides u and d, depending on the sign in (D.12). Analogously, $\exp(\Delta Y_2)$ gives the factors A,B,C,D, depending on the signs. Investigating possible combinations of signs in view of the requirement $AD = BC$ shows that A and D must be those versions of ΔY_2 where both signs are equal. The resulting tree is defined by

$$\begin{aligned} u &:= \exp(\mu_1 \Delta t + \sigma_1 \sqrt{\Delta t}) \\ d &:= \exp(\mu_1 \Delta t - \sigma_1 \sqrt{\Delta t}) \\ A &:= \exp(\mu_2 \Delta t + \sigma_2 \sqrt{\Delta t}\,[\rho + \sqrt{1-\rho^2}]) \\ B &:= \exp(\mu_2 \Delta t + \sigma_2 \sqrt{\Delta t}\,[\rho - \sqrt{1-\rho^2}]) \\ C &:= \exp(\mu_2 \Delta t - \sigma_2 \sqrt{\Delta t}\,[\rho - \sqrt{1-\rho^2}]) \\ D &:= \exp(\mu_2 \Delta t - \sigma_2 \sqrt{\Delta t}\,[\rho + \sqrt{1-\rho^2}])\,. \end{aligned} \tag{D.13}$$

[9] It is not the same as in (D.11), but we use the same notation ΔY.

With these factors, $AD = BC$ holds and the tree is recombining, and the first moments match those of the continuous problem.

For initial prices $x^0 := S_1^0 := S_1(0)$, $y^0 := S_2^0 := S_2(0)$, and time level t_ν, the S_1-components of the grid according to (D.9) distribute in the same way as for the one-dimensional tree,

$$x_i^\nu := S_1^0 u^i d^{\nu-i} \text{ for } i = 0, \dots, \nu\,. \tag{D.14}$$

As the reader may check in Exercise 6.2, the second (S_2)-components belonging to x_i^ν are

$$y_{i,j}^\nu := S_2^0 \exp(\mu_2 \nu \Delta t)\, \exp\left(\sigma_2 \sqrt{\Delta t}\left[\rho(2i-\nu) + \sqrt{1-\rho^2}(2j-\nu)\right]\right) \tag{D.15}$$

for $j = 0, \dots, \nu$.

In the t_3-time level, the tree has 16 nodes. Generally, in the t_ν-time level, there are $(\nu+1)^2$ nodes. This makes altogether

$$\frac{1}{6}(M+1)(M+2)(2M+3)$$

nodes of the entire three-dimensional tree, a complexity of $O(M^3)$.

The valuation of the tree is standard, with

$$V_{i,j}^{\text{cont}} = \exp(-r\Delta t)\,\tfrac{1}{4}\,(V_{i,j}^{\nu+1} + V_{i+1,j}^{\nu+1} + V_{i,j+1}^{\nu+1} + V_{i+1,j+1}^{\nu+1})\,. \tag{D.16}$$

For an example see Exercise 6.2 and the illustrations of Fig. 6.2.

In principle, this approach extends to multifactor options with $n > 2$ underlying assets. But costs of the order $O(M^{n+1})$ restrict the fineness M and thus the accuracy. For high values of n this may call for other methods.

A "five-jump model" (version with five possible outcomes in each mesh) is [213], setting for example $u_i = \mathrm{e}^{\sigma_i\sqrt{\lambda_i \Delta t}}$ for suitable parameters λ_i, $i = 1, 2$, and then deriving probabilities. A four-jump model is suggested in [48]. According to [213], five-jump models appear to have a smoother convergence.

D.5 Implied Trees for Variable Volatility

The Black–Scholes model assumes a constant volatility σ. But in market data of traded options, one observes the volatility smile, a non-constant dependence of σ on the strike K and on the time for maturity T. Inserting a local volatility $\sigma(S,t)$ into the Black–Scholes approach improves its pricing ability. (⟶ Appendix A.6)

Such an additional flexibility of the Black–Scholes approach can be adapted also by tree methods. Here we outline the implied tree of Derman and Kani [100].

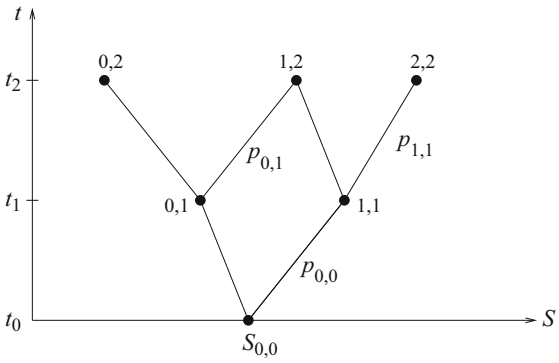

Fig. D.5 The first nodes [labeled (j,i)] of a general binomial grid, with probabilities $p_{j,i}$ and variable S-positions of the nodes; initial part of the tree

Implied trees are calibrated to market prices of options. Note the contrast to the standard tree, which is calibrated to the underlying process S_t independently of the option. We are back in the one-dimensional situation, with index i counting the time levels, and j labels the nodes on the level i.

Variable Grid

To allow for more generality, each mesh has its own local probability (Fig. D.5). For any node with position (j,i) we ask what the probability is that this node is reached. For each possible path that connects the root node $(0,0)$ to the (j,i)-node, the transition probability of this path is the product of the involved local probabilities. And the probability that the node is hit is the sum of the probabilities of all possible paths that lead to (j,i). Up to a discounting factor, this probability to hit a node (j,i) is defined by numbers $\lambda_{j,i}$,[10]

> $\lambda_{j,i}$ is the sum of the products of all riskless-discounted transition probabilities, with summation over all paths leading from the root $(0,0)$ to the node (j,i).

For example,

$$\lambda_{1,2} = \mathrm{e}^{-r2\Delta t}\,[p_{0,1}(1-p_{0,0}) + (1-p_{1,1})p_{0,0}]\,,$$

compare Fig. D.5. Clearly there is a forward recursion for these $\lambda_{j,i}$. Fixing $\lambda_{0,0} := 1$,

$$\lambda_{0,1} = \mathrm{e}^{-r\Delta t}\,\lambda_{0,0}(1-p_{0,0}), \quad \lambda_{1,1} = \mathrm{e}^{-r\Delta t}\,\lambda_{0,0}\,p_{0,0}$$

holds. We distinguish interior nodes of the tree, to which two entries exist, and boundary nodes, which have one entry only.

[10] The numbers $\lambda_{j,i}$ are also called Arrow–Debreu prices.

Recursion for the $\lambda_{j,i+1}$ along level t_{i+1}:

$$\begin{aligned}\lambda_{0,i+1} &= \mathrm{e}^{-r\Delta t}\,\lambda_{0,i}\,(1-p_{0,i})\,,\\ \lambda_{j+1,i+1} &= \mathrm{e}^{-r\Delta t}\,[\lambda_{j,i}\,p_{j,i}+\lambda_{j+1,i}\,(1-p_{j+1,i})] \quad \text{for } 0\le j\le i-1\,,\\ \lambda_{i+1,i+1} &= \mathrm{e}^{-r\Delta t}\,\lambda_{i,i}\,p_{i,i}\,.\end{aligned} \tag{D.17}$$

In the special case of the standard tree of Sect. 1.4, with $p_{j,i} = p$ for all j, i, the Bernoulli experiment results in the probabilities $b_{M,p}(j)$ from (B.14), which describe the probability that the node (j, M) is hit. Since the $\lambda_{j,i}$ distribute the discounting over the time slices, the relation

$$\lambda_{j,M} = \mathrm{e}^{-rT}\binom{M}{j}p^j(1-p)^{M-j}$$

with the binomial probability p holds. In this special case with constant p, as well as in the general case with arbitrary probabilities $p_{j,i}$, the final probability to hit the node (j, M) is $\mathrm{e}^{rT}\lambda_{j,M}$. In any case, the expectation (D.1) of a European vanilla option with payoff Ψ can be written

$$V(S_{0,0},0) = \sum_{j=0}^{M}\lambda_{j,M}\,\Psi(S_{j,M})\,. \tag{D.18}$$

Fixing the Probability

The method of Derman and Kani sets up reasonable probabilities $p_{j,i}$ and positions $(S_{j,i}, t_i)$ of the nodes (j, i). Thereby, the S-coordinates of the grid are designed such that the tree matches market data.

Assumptions D.3 *"Market prices" of European-style vanilla options can be obtained for any strike K and any maturity T, for put and call.*

In reality, market values $V(S_0, 0; K, T)$ are available only for a scarce set of pairs (K, T). But there are algorithms that apply interpolation and smoothing to construct intermediate values of V for almost any (K, T), see [128, 154]. So the Assumption D.3 is not as restrictive as it might seem. But it requires cumbersome preparatory work. Since this basic prerequisite is not explained in this text, we shall confine ourselves to key ideas.

Suppose all nodes are placed and all probabilities are fixed for the time level t_i. That is, the $2i + 2$ numbers

$$\begin{aligned}&S_{0,i}, S_{1,i}, \ldots, S_{i,i}\,,\\ &\lambda_{0,i}, \lambda_{1,i}, \ldots, \lambda_{i,i}\end{aligned}$$

are available. For the next time level t_{i+1} the $2i+4$ numbers

$$S_{0,i+1}, S_{1,i+1}, \ldots, S_{i+1,i+1} ,$$
$$\lambda_{0,i+1}, \lambda_{1,i+1}, \ldots, \lambda_{i+1,i+1}$$

are to be calculated. This requires $2i+3$ equations, because the recursion (D.17) for the $\lambda_{j,i}$ requires only $i+1$ probabilities

$$p_{0,i}, \ldots, p_{i,i} .$$

$i+1$ of the equations are easily set up, requesting as usual that the expectation over the time step Δt matches that of the continuous model. Similar as in Sect. 1.4, this requirement is written

$$p_{j,i}\, S_{j+1,i+1} + (1-p_{j,i})\, S_{j,i+1} = S_{j,i}\, \mathrm{e}^{(r-\delta)\Delta t} \tag{D.19}$$

for $0 \le j \le i$. This sets up $i+1$ equations for the probabilities $p_{j,i}$, which in turn fix the values $\lambda_{j,i+1}$ via the recursion (D.17), provided the S-values along t_{i+1} are fixed. It remains to set up $i+2$ equations for the free grid coordinates $S_{j,i+1}$ for $0 \le j \le i+1$.

At this stage, the market data enter. According to Assumptions D.3, (approximate) vanilla put and call prices can be made available for arbitrary "strikes" and "maturities." Consider t_{i+1} as an auxiliary maturity and choose the $i+1$ auxiliary strikes $S_{0,i}, \ldots, S_{i,i}$. So market values

$$\begin{aligned} C_{j,i} &:= V_{\text{call}}^{\text{market}}(S_{0,0}, 0;\, S_{j,i}, t_{i+1}) \\ P_{j,i} &:= V_{\text{put}}^{\text{market}}(S_{0,0}, 0;\, S_{j,i}, t_{i+1}) \end{aligned} \tag{D.20}$$

are known for $0 \le j \le i$. In (D.20), there are only $i+1$ independent option values, because European-style put and call options are related through the put-call parity.

Recursion Based on Call Data

Next we discuss how the market call values $C_{j,i}$ enter. For the strike $S_{j,i}$, we apply (D.18), where M is replaced by $i+1$. Then, by (D.20), the grid values $S_{k,i+1}$ are to be chosen such that

$$C_{j,i} = \sum_{k=0}^{i+1} \lambda_{k,i+1} \Psi(S_{k,i+1}) = \sum_{k=0}^{i+1} \lambda_{k,i+1} (S_{k,i+1} - S_{j,i})^+ .$$

Assuming an ordered grid in the sense of $S_{j,i+1} < S_{j,i} < S_{j+1,i+1}$, this requirement can be written

$$C_{j,i} = \sum_{k=j+1}^{i+1} \lambda_{k,i+1} (S_{k,i+1} - S_{j,i}) .$$

Substituting the recursion (D.17), after some formula manipulations, leads to a recursion between two neighboring (unknown) S-node values along the line t_{i+1},

$$S_{j+1,i+1} = f(S_{j,i+1}) \,, \tag{D.21}$$

for a rather involved function f. Equation (D.21) fixes a new node $S_{j+1,i+1}$ after the previous node $S_{j,i+1}$ was set. Then the probabilities are given by (D.19) and (D.17). The recursion (D.21) is started in the middle of the line t_{i+1}, which can be fixed by centering at the $S_{0,0}$ of the classical tree (Sect. 1.4).

For the other half of the S-Values along line t_{i+1} a similar recursion is obtained from put values $P_{j,i}$, according to (D.20). For details see the original paper [100], or the explanation in [337]. In practice, both the approach to establish the bold Assumptions D.3 and an ordered generation of S-nodes are tricky. In the end, the local volatilities are obtained by

$$\sigma_{j,i} = \sqrt{\frac{p_{j,i}(1-p_{j,i})}{\Delta t}} \, \log \frac{S_{j+1,i+1}}{S_{j,i}} \,. \tag{D.22}$$

Appendix E
Complementary Material

This appendix lists useful formula without further explanation. Many formulas can be found in [172].

E.1 Bounds for Options

The following bounds for vanilla options can be derived based on arbitrage arguments, see [88, 191, 196, 234, 269]. If neither the subscript $_{\mathrm{C}}$ nor $_{\mathrm{P}}$ is listed, the inequality holds for both put and call. If neither the $^{\mathrm{Eur}}$ nor the $^{\mathrm{Am}}$ is listed, the inequality holds for both American and European options. We always assume $r > 0$ (Fig. E.1).

(a) Bounds valid for both American and European options, no matter whether dividends are paid or not:

$$
\begin{aligned}
0 \le V_{\mathrm{C}}(S_t, t) &\le S_t \\
0 \le V_{\mathrm{P}}(S_t, t) &\le K \\
V^{\mathrm{Eur}}(S_t, t) &\le V^{\mathrm{Am}}(S_t, t) \\
S_t - K &\le V_{\mathrm{C}}^{\mathrm{Am}}(S_t, t) \\
K - S_t &\le V_{\mathrm{P}}^{\mathrm{Am}}(S_t, t) \\
V_{\mathrm{P}}^{\mathrm{Eur}}(S_t, t) &\le K\mathrm{e}^{-r(T-t)}
\end{aligned}
$$

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

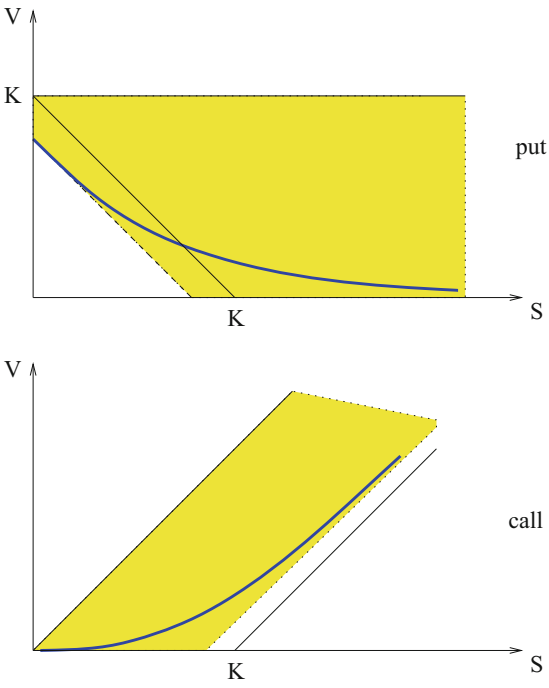

Fig. E.1 Bounding curves for the value of vanilla put and call options ($r > 0$, $\delta = 0$); for both put and call a European value function is plotted, with $r > 0$, $\delta = 0$

Lower bounds incorporating a continuous dividend yield δ (set $\delta = 0$ in case there is no dividend yield): The above relations and the put-call parity (A.16) imply

$$S_t \, \mathrm{e}^{-\delta(T-t)} - K\mathrm{e}^{-r(T-t)} \leq V_{\mathrm{C}}(S_t, t)$$
$$K\mathrm{e}^{-r(T-t)} - S_t \, \mathrm{e}^{-\delta(T-t)} \leq V_{\mathrm{P}}(S_t, t)$$

The zero of the lower bound is $K\mathrm{e}^{(\delta-r)(T-t)}$.

(b) For bounds on the early-exercise boundary, see Appendix A.5.

(c) Monotonicity of the value function:
Monotonicity with respect to S:

$$V_{\mathrm{C}}(S_1, t) < V_{\mathrm{C}}(S_2, t) \text{ for } S_1 < S_2 \,,$$
$$V_{\mathrm{P}}(S_1, t) > V_{\mathrm{P}}(S_2, t) \text{ for } S_1 < S_2 \,,$$

which implies

$$\frac{\partial V_{\mathrm{C}}}{\partial S} > 0 \,, \quad \frac{\partial V_{\mathrm{P}}}{\partial S} < 0 \,.$$

Monotonicity of American options with respect to time:

$$V_{\mathrm{C}}^{\mathrm{Am}}(S,t_1) \geq V_{\mathrm{C}}^{\mathrm{Am}}(S,t_2) \text{ for } t_1 < t_2\,,$$
$$V_{\mathrm{P}}^{\mathrm{Am}}(S,t_1) \geq V_{\mathrm{P}}^{\mathrm{Am}}(S,t_2) \text{ for } t_1 < t_2\,,$$

which implies

$$\frac{\partial V^{\mathrm{Am}}}{\partial t} \leq 0\,.$$

Options are convex with respect to K and with respect to S. This holds for the standard Black–Merton–Scholes model; for other models relations are more complicated [118].

To express monotonicity with respect to the strike K or to the time to expiration T, we indicate dependencies by writing $V(S,t;T,K)$, and only quote the parameter that is changed.

$$V^{\mathrm{Am}}(\,.\,;T_1) \leq V^{\mathrm{Am}}(\,.\,;T_2) \;\text{ for } T_1 < T_2$$
$$V_{\mathrm{C}}(\,.\,;K_1) \geq V_{\mathrm{C}}(\,.\,;K_2) \;\text{ for } K_1 < K_2$$
$$V_{\mathrm{P}}(\,.\,;K_1) \leq V_{\mathrm{P}}(\,.\,;K_2) \;\text{ for } K_1 < K_2$$
$$V(\,.\,;\sigma_1) \leq V(\,.\,;\sigma_2) \;\text{ for } \sigma_1 < \sigma_2$$

The first of these inequalities implies that the value of a perpetual option ($T \to \infty$) is an upper bound to the value of an American option.

(d) Put-call relation for American options:

$$K\mathrm{e}^{-r(T-t)} + V_{\mathrm{C}}^{\mathrm{Am}}(S,t) \leq S + V_{\mathrm{P}}^{\mathrm{Am}}(S,t)\,.$$

This holds no matter whether dividends are paid or not. If the asset pays no dividends, then also the upper bound

$$S + V_{\mathrm{P}}^{\mathrm{Am}}(S,t) - V_{\mathrm{C}}^{\mathrm{Am}}(S,t) \leq K$$

holds.

E.2 Approximation Formula

Distribution Function of the Standard Normal Distribution

$$f(x) := \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)$$

$$F(x) := \int_{-\infty}^{x} f(t)\,\mathrm{d}t$$

The calculation of F can be based on the error function, see Exercise 1.5. Applying quadrature is not the most efficient way to approximate the integral. For full double-precision accuracy, there are generic codes available (as the function `derf` in FORTRAN). For such high accuracy—according to our findings—it is also recommendable to approximate F by a spline.

Frequently lower accuracy suffices. Related approximations of the error function can be found in [171], which is a rich source of approximation formulas for all kind of functions and different requirements of precision. Here we present an algorithm from [1, formula (26.2.17)], which does not make use of the error function.

Let us define

$$z := \frac{1}{1 + 0.2316419x}$$

and the coefficients

$$\begin{aligned} a_1 &= 0.319381530 & a_4 &= -1.821255978 \\ a_2 &= -0.356563782 & a_5 &= 1.330274429 \\ a_3 &= 1.781477937\,. \end{aligned}$$

Then

$$F(x) = 1 - f(x)\left(a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4 + a_5 z^5\right) + \varepsilon(x)\,,$$

for $0 \le x < \infty$ with an absolute error ε bounded by

$$|\varepsilon(x)| < 7.5 * 10^{-8}\,.$$

Hence we have the approximating formula

$$F(x) \approx 1 - f(x)z((((a_5 z + a_4)z + a_3)z + a_2)z + a_1)\,,$$

which requires 17 arithmetic operations and the evaluation of the exponential function to obtain an accuracy of about seven decimals. For $x < 0$ apply $F(x) = 1 - F(-x)$. To save time, the evaluation of the exponential function should not use

the generic double-precision code since this would be too much effort for a final seven-digit accuracy. An alternative can be found in [171]. A seven-digit version for F that does not need the exponential function, is formula (26.2.119) in [1].

Inversion Formula

A FORTRAN code for the inversion of the normal distribution can be found in

http://lib.stat.cmu.edu/apstat/111.

(Many other codes relevant for statistical computation can be obtained via the `.../apstat` page.) Here we report the formula of [278] to approximate the inverse function of the standard normal distribution

$$F(x) := \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{t^2}{2}\right) \mathrm{d}t\,.$$

That is, we calculate $x = G(u)$ such that $G(u) \approx F^{-1}(u)$. The interval $0 < u < 1$ is truncated to $10^{-12} \le u \le 1 - 10^{-12}$. Symmetry with respect to $(x, u) = (0, 0.5)$ is exploited. The interval is subdivided into two relevant parts, namely,

$$0.08 < u < 0.92 \quad \text{and} \quad 0.92 \le u \le 1 - 10^{-12}\,.$$

The part $10^{-12} \le u \le 0.08$ is obtained by symmetry. For each of the two subintervals an appropriate approximation is given. In the middle part of the interval a rational approximation in the form

$$(u - 0.5)\frac{\sum_{j=0}^{3} a_j (u - 0.5)^{2j}}{1 + \sum_{j=0}^{3} b_j (u - 0.5)^{2j}}$$

is used, whereas the tails are approximated by a polynomial in $\log(-\log r)$, where $10^{-12} \le r \le 0.08$.

Algorithm E.1 (Inversion of the Standard Normal Distribution)

input: u, drawn from $\mathcal{U}(0, 1)$
$y := u - 0.5$
in case $|y| < 0.42$*:*
$r := y^2$
$x := y\dfrac{((a_3 r + a_2)r + a_1)r + a_0}{(((b_3 r + b_2)r + b_1)r + b_0)r + 1}$
in case $|y| \ge 0.42$*:*
$r := u$ *, in case* $y > 0$ *set* $r := 1 - u$
$r := \log(-\log r)$
$x := c_0 + r(c_1 + r(c_2 + r(c_3 + r(c_4 + r(c_5 + r(c_6 + r(c_7 + rc_8)))))))$
in case $y < 0$ *set* $x := -x$
output: x

The coefficients of the above algorithm are given by[1]

$$
\begin{aligned}
a_0 &= 2.50662823884,\\
a_1 &= -18.61500062529,\\
a_2 &= 41.39119773534,\\
a_3 &= -25.44106049637
\end{aligned}
$$

$$
\begin{aligned}
b_0 &= -8.47351093090,\\
b_1 &= 23.08336743743,\\
b_2 &= -21.06224101826,\\
b_3 &= 3.13082909833
\end{aligned}
$$

$$
\begin{aligned}
c_0 &= 0.3374754822726147,\\
c_1 &= 0.9761690190917186,\\
c_2 &= 0.1607979714918209,\\
c_3 &= 0.0276438810333863,\\
c_4 &= 0.0038405729373609,\\
c_5 &= 0.0003951896511919,\\
c_6 &= 0.0000321767881768,\\
c_7 &= 0.0000002888167364,\\
c_8 &= 0.0000003960315187
\end{aligned}
$$

The rational approximation formula for $|y| < 0.42$ (that is, $0.08 < u < 0.92$) is reported to have a largest absolute error of $3 \cdot 10^{-9}$.

E.3 Software

A dedicated computer person will program the mathematics such that the resulting codes run with utmost possible speed. Such a person will probably use compilers like C, C++, or FORTRAN to create production codes, where the speed counts. But there are packages available that make programming, implementing, testing, and graphics more comfortable. For example, MATLAB offers a platform for scientific computation and numerical experiments, and includes a Financial Derivatives Toolbox.[2]

Several programs related to finance have been published. For MATLAB codes see [50, 181], for MATHEMATICA codes see [349], and C++ programs are in [2, 246]. For elementary computations, spreadsheets are also used. Programs in various levels can also found, for example, in [172, 191]. Pseudo codes for several types of options can be found in [82].

[1] These digits are listed in [278].

[2] Figures 3.7, 5.10, 6.3, 6.4, 7.1 are based on MATLAB graphics. The other figures in this book were prepared using `xfig` and `gnuplot`.

For partial differential equations, the finite-element program PDE2D is available via the University of Texas, El Paso. See also the finite-element programs referred to in [2], such as `FreeFem++`. The PREMIA project offers codes via www-rocq.inria.fr/mathfi. For further hints and test algorithms see the platform www.compfin.de .

References

1. Abramowitz, M., Stegun, I.: Handbook of Mathematical Functions. With Formulas, Graphs, and Mathematical Tables. Dover, New York (1968)
2. Achdou, Y., Pironneau, O.: Computational Methods for Option Pricing. SIAM, Philadelphia (2005)
3. Adams, R.A.: Sobolev Spaces. Academic Press, New York (1975)
4. AitSahlia, F., Carr, P.: American options: a comparison of numerical methods. In: Rogers, L.C.G., Talay, D. (eds.) Numerical Methods in Finance, pp. 67–87. Cambridge University Press, Cambridge (1997)
5. Alfonsi, A.: On the discretization schemes for the CIR (and Besselsquared) processes. Monte Carlo Methods Appl. **11**, 355–384 (2005)
6. Almendral, A., Oosterlee, C.W.: Numerical valuation of options with jumps in the underlying. Appl. Numer. Math. **53**, 1–18 (2005)
7. Almendral, A., Oosterlee, C.W.: Highly accurate evaluation of European and American options under the Variance Gamma process. J. Comput. Finance **10**(1), 21–42 (2006)
8. Andersen, L., Andreasen, J.: Jump diffusion process: volatility smile fitting and numerical methods for option pricing. Rev. Deriv. Res. **4**, 231–262 (2000)
9. Andersen, L., Broadie, M.: Primal-dual simulation algorithm for pricing multidimensional American options. Manag. Sci. **50**, 1222–1234 (2004)
10. Andersen, L.B.G., Brotherton-Ratcliffe, R.: The equity option volatility smile: an implicit finite-difference approach. J. Comput. Finance **1**(2), 5–38 (1997/1998)
11. Ané, T., Geman, H.: Order flow, transaction clock, and normality of asset returns. J. Finance **55**, 2259–2284 (2000)
12. Arnold, L.: Stochastic Differential Equations (Theory and Applications). Wiley, New York (1974)
13. Arouna, B.: Robbins-Monro algorithms and variance reduction in finance. J. Comput. Finance **7**(2), 35–61 (2003)
14. Artzner, P., Delbaen, F., Eber, J.-M., Heath, D.: Coherent measures of risk. Math. Finance **9**, 203–228 (1999)
15. Avellaneda, M.: Quantitative Modeling of Derivative Securities. From Theory to Practice. Chapman & Hall, Boca Raton (2000)
16. Avellaneda, M., Levy, A., Parás, A.: Pricing and hedging derivative securities in markets with uncertain volatilities. Appl. Math. Finance **2**, 73–88 (1995)
17. Avellaneda, M., Parás, A.: Dynamic hedging portfolios for derivative securities in the presence of large transaction costs. Appl. Math. Finance **1**, 165–194 (1994)

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0

18. Avellaneda, M., Parás, A.: Managing the volatility risk of derivative securities: the Lagrangian volatility model. Appl. Math. Finance **3**, 21–53 (1996)
19. Babuška, I., Strouboulis, T.: The Finite Element Method and Its Reliability. Oxford Science, Oxford (2001)
20. Ball, C.A., Roma, A.: Stochastic volatility option pricing. J. Financ. Quant. Anal. **29**, 589–607 (1994)
21. Barles, G.: Convergence of numerical schemes for degenerate parabolic equations arising in finance theory. In: Rogers, L.C.G., Talay, D. (eds.) Numerical Methods in Finance, pp. 2–21. Cambridge University Press, Cambridge (1997)
22. Barles, G., Burdeau, J., Romano, M., Samsœn, N.: Critical stock prices near expiration. Math. Finance **5**, 77–95 (1995)
23. Barles, G., Daher, Ch., Romano, M.: Convergence of numerical schemes for parabolic equations arising in finance theory. Math. Models Methods Appl. Sci. **5**, 125–143 (1995)
24. Barles, G., Soner, H.M.: Option pricing with transaction costs and a nonlinear Black-Scholes equation. Finance Stochast. **2**, 369–397 (1998)
25. Barndorff-Nielsen, O.E.: Processes of normal inverse Gaussian type. Finance Stochast. **2**, 41–68 (1997)
26. Barone-Adesi, G., Whaley, R.E.: Efficient analytic approximation of American option values. J. Finance **42**, 301–320 (1987)
27. Barone-Adesi, G., Whaley, R.E.: On the valuation of American put options on dividend-paying stocks. Adv. Futures Options Res. **3**, 1–13 (1988)
28. Barraquand, J., Pudet, T.: Pricing of American path-dependent contingent claims. Math. Finance **6**, 17–51 (1996)
29. Barrett, R., et al.: Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods. SIAM, Philadelphia (1994)
30. Bates, D.: Jumps and stochastic volatility: the exchange rate processes implicit in Deutschmark options. Rev. Financ. Stud. **9**, 69–107 (1996)
31. Baxter, M., Rennie, A.: Financial Calculus. An Introduction to Derivative Pricing. Cambridge University Press, Cambridge (1996)
32. Behrends, E.: Introduction to Markov Chains. Vieweg, Braunschweig (2000)
33. Bellman, R.: Dynamic Programming. Princeton University Press, Princeton (1957)
34. Ben Hamida, S., Cont, R.: Recovering volatility from option prices by evolutionary optimization. J. Comput. Finance **8**(4), 43–76 (2005)
35. Bensoussan, A.: On the theory of option pricing. Acta Appl. Math. **2**, 139–158 (1984)
36. Berridge, S.J., Schumacher, J.M.: Pricing high-dimensional American options using local consistency conditions. In: Appleby, J.A.D., et al. (eds.) Numerical Methods for Finance. Chapman & Hall, Boca Raton (2008)
37. Billingsley, P.: Probability and Measure. Wiley, New York (1979)
38. Bischi, G.I., Sushko, I. (eds.): Dynamic Modelling in Economics & Finance. Special Issue of Chaos, Solitons and Fractals **29**(3) (2006)
39. Bischi, G.I., Valori, V.: Nonlinear effects in a discrete-time dynamic model of a stock market. Chaos Solitons Fractals **11**, 2103–2121 (2000)
40. Björk, T.: Arbitrage Theory in Continuous Time. Oxford University Press, Oxford (1998)
41. Black, F., Scholes, M.: The pricing of options and corporate liabilities. J. Polit. Econ. **81**, 637–659 (1973)
42. Blomeyer, E.C.: An analytic approximation for the American put price for options with dividends. J. Financ. Quant. Anal. **21**, 229–233 (1986)
43. Bouchaud, J.-P., Potters, M.: Theory of Financial Risks. From Statistical Physics to Risk Management. Cambridge University Press, Cambridge (2000)
44. Bouleau, N.: Martingales et Marchés Financiers. Edition Odile Jacob, Paris (1998)
45. Box, G.E.P., Muller, M.E.: A note on the generation of random normal deviates. Ann. Math. Stat. **29**, 610–611 (1958)
46. Boyle, P.P.: Options: a Monte Carlo approach. J. Financ. Econ. **4**, 323–338 (1977)

47. Boyle, P., Broadie, M., Glasserman, P.: Monte Carlo methods for security pricing. J. Econ. Dyn. Control **21**, 1267–1321 (1997)
48. Boyle, P.P., Evnine, J., Gibbs, S.: Numerical evaluation of multivariate contingent claims. Rev. Financ. Stud. **2**, 241–250 (1989)
49. Brachet, M.-E., Taflin, E., Tcheou, J.M.: Scaling transformation and probability distributions for time series. Chaos Solitons Fractals **11**, 2343–2348 (2000)
50. Brandimarte, P.: Numerical Methods in Finance and Economics. A MATLAB-Based Introduction. Wiley, Hoboken (2006)
51. Breen, R.: The accelerated binomial option pricing model. J. Financ. Quant. Anal. **26**, 153–164 (1991)
52. Brennan, M.J., Schwartz, E.S.: The valuation of American put options. J. Finance **32**, 449–462 (1977)
53. Brenner, S.C., Scott, L.R.: The Mathematical Theory of Finite Element Methods, 2nd edn. Springer, New York (2002)
54. Brent, R.P.: On the periods of generalized Fibonacci recurrences. Math. Comput. **63**, 389–401 (1994)
55. Briani, M., La Chioma, C., Natalini, R.: Convergence of numerical schemes for viscosity solutions to integro-differential degenerate parabolic problems arising in financial theory. Numer. Math. **98**, 607–646 (2004)
56. Broadie, M., Detemple, J.: American option valuation: new bounds, approximations, and a comparison of existing methods. Rev. Financ. Stud. **9**, 1211–1250 (1996)
57. Broadie, M., Detemple, J.: Recent advances in numerical methods for pricing derivative securities. In: Rogers, L.C.G., Talay, D. (eds.) Numerical Methods in Finance, pp. 43–66. Cambridge University Press, Cambridge (1997)
58. Broadie, M., Glasserman, P.: Pricing American-style securities using simulation. J. Econ. Dyn. Control **21**, 1323–1352 (1997)
59. Broadie, M., Glasserman, P.: A stochastic mesh method for pricing high-dimensional American options. J. Comput. Finance **7**(4), 35–72 (2004)
60. Brock, W.A., Hommes, C.H.: Heterogeneous beliefs and routes to chaos in a simple asset pricing model. J. Econ. Dyn. Control **22**, 1235–1274 (1998)
61. Broyden, C.G.: The convergence of a class of double-rank minimization algorithms 1. General considerations. IMA J. Appl. Math. **6**, 76–90 (1970)
62. Bruti-Liberati, N., Platen, E.: On weak predictor-corrector schemes for jump-diffusion processes in finance. Research Paper, University of Sydney (2006)
63. Bunch, D.S., Johnson, H.: A simple and numerically efficient valuation method for American puts using a modified Geske-Johnson approach. J. Finance **47**, 809–816 (1992)
64. Caflisch, R.E., Morokoff, W., Owen, A.: Valuation of mortgaged-backed securities using Brownian bridges to reduce effective dimension. J. Comput. Finance **1**(1), 27–46 (1997)
65. Carmona, R., Durrleman, V.: Generalizing the Black–Scholes formula to multivariate contingent claims. J. Comput. Finance **9**(2), 43–67 (2005)
66. Carr, P., Faguet, D.: Fast accurate valuation of American options. Working paper, Cornell University (1995)
67. Carr, P., Geman, H., Madan, D.B., Yor, M.: Stochastic volatility for Lévy processes. Math. Finance **13**, 345–382 (2003)
68. Carr, P., Madan, D.B.: Option valuation using the fast Fourier transform. J. Comput. Finance **2**(4), 61–73 (1999)
69. Carr, P., Wu, L.: Time-changed Lévy processes and option pricing. J. Financ. Econ. **71**, 113–141 (2004)
70. Carriere, J.F.: Valuation of the early-exercise price for options using simulations and nonparametric regression. Insur. Math. Econ. **19**, 19–30 (1996)
71. Cash, J.R.: Two new finite difference schemes for parabolic equations. SIAM J. Numer. Anal. **21**, 433–446 (1984)
72. Chan, T.F., Golub, G.H., LeVeque, R.J.: Algorithms for computing the sample variance: analysis and recommendations. Am. Stat. **37**, 242–247 (1983)

73. Chen, S.-H. (ed.): Genetic Algorithms and Genetic Programming in Computational Finance. Kluwer, Boston (2002)
74. Chen, X., Chadam, J.: Analytical and numerical approximations for the early exercise boundary for American put options. Dyn. Continuous Discrete Impulsive Syst. A **10**, 649–660 (2003)
75. Chen, X., Chadam, J.: A mathematical analysis of the optimal exercise boundary for American put options. SIAM J. Math. Anal. **38**, 1613–1641 (2007)
76. Chiarella, C., Dieci, R., Gardini, L.: Speculative behaviour and complex asset price dynamics. In: Bischi, G.I. (ed.) Proceedings Urbino 2000 (2000)
77. Choi, H.I., Heath, D., Ku, H.: Valuation and hedging of options with general payoff under transaction costs. J. Kor. Math. Soc. **41**, 513–533 (2004)
78. Chung, K.L., Williams, R.J.: Introduction to Stochastic Integration. Birkhäuser, Boston (1983)
79. Ciarlet, P.G.: Basic error estimates for elliptic problems. In: Ciarlet, P.G., Lions, J.L. (eds.) Handbook of Numerical Analysis, Vol. II. Elsevier/North-Holland, Amsterdam (1991)
80. Ciarlet, P., Lions, J.L.: Finite Difference Methods (Part 1) Solution of Equations in $\mathbb{R}^n$. North-Holland/Elsevier, Amsterdam (1990)
81. Clarke, N., Parrot, A.K.: Multigrid for American option pricing with stochastic volatility. Appl. Math. Finance **6**, 177–179 (1999)
82. Clewlow, L., Strickland, C.: Implementing Derivative Models. Wiley, Chichester (1998)
83. Coleman, T.F., Li, Y., Verma, Y.: A Newton method for American option pricing. J. Comput. Finance **5**(3), 51–78 (2002)
84. Cont, R., Tankov, P.: Financial Modelling with Jump Processes. Chapman & Hall, Boca Raton (2004)
85. Cont, R., Voltchkova, E.: Finite difference methods for option pricing in jump-diffusion and exponential Lévy models. SIAM J. Numer. Anal. **43**, 1596–1626 (2005)
86. Cox, J.C., Ingersoll, J.E., Ross, S.A.: A theory of the term structure of interest rates. Econometrica **53**, 385–407 (1985)
87. Cox, J.C., Ross, S., Rubinstein, M.: Option pricing: a simplified approach. J. Financ. Econ. **7**, 229–263 (1979)
88. Cox, J.C., Rubinstein, M.: Options Markets. Prentice Hall, Englewood Cliffs (1985)
89. Crandall, M., Ishii, H., Lions, P.L.: User's guide to viscosity solutions of second order partial differential equations. Bull. Am. Math. Soc. **27**, 1–67 (1992)
90. Crank, J.: Free and Moving Boundary Problems. Clarendon Press, Oxford (1984)
91. Crank, J.C., Nicolson, P.: A practical method for numerical evaluation of solutions of partial differential equations of the heat-conductive type. Proc. Camb. Philos. Soc. **43**, 50–67 (1947)
92. Cryer, C.: The solution of a quadratic programming problem using systematic overrelaxation. SIAM J. Control **9**, 385–392 (1971)
93. Cyganowski, S., Kloeden, P., Ombach, J.: From Elementary Probability to Stochastic Differential Equations with MAPLE. Springer, Heidelberg (2001)
94. Dahlbokum, A.: Empirical performance of option pricing models based on time-changed Lévy processes. Available at SSRN: http://ssrn.com/abstract=1675321 (2010)
95. Dai, M.: A closed-form solution for perpetual American floating strike lookback options. J. Comput. Finance **4**(2), 63–68 (2000)
96. Dai, T.-S., Lyuu, Y.-D.: The bino-trinomial tree: a simple model for efficient and accurate option pricing. J. Deriv. **17**, 7–24 (2010)
97. Dana, R.-A., Jeanblanc, M.: Financial Markets in Continuous Time. Springer, Berlin (2003)
98. Dempster, M.A.H., Hutton, J.P.: Pricing American stock options by linear programming. Math. Finance **9**, 229–254 (1999)
99. Dempster, M.A.H., Hutton, J.P., Richards, D.G.: LP valuation of exotic American options exploiting structure. J. Comput. Finance **2**(1), 61–84 (1998)
100. Derman, E., Kani, I.: Riding on a smile. Risk **7**, 32–39 (1994)
101. Detemple, J.: American options: symmetry properties. In: Jouini, E., et al. (eds.) Option Pricing, Interest Rates and Risk Management. Cambridge University Press, Cambridge (2001)

102. Deutsch, H.-P.: Derivatives and Internal Models. Palgrave, Houndmills (2002)
103. Devroye, L.: Non-uniform Random Variate Generation. Springer, New York (1986)
104. d'Halluin, Y., Forsyth, P.A., Labahn, G.: A semi-Lagrangian approach for American Asian options under jump diffusion. SIAM J. Sci. Comput. **27**, 315–345 (2005)
105. d'Halluin, Y., Forsyth, P.A., Vetzal, K.R.: Robust numerical methods for contingent claims under jump diffusion processes. IMA J. Numer. Anal. **25**, 87–112 (2005)
106. Dieci, R., Bischi, G.-I., Gardini, L.: From bi-stability to chaotic oscillations in a macroeconomic model. Chaos Solitons Fractals **12**, 805–822 (2001)
107. Doeblin, W.: Sur l'équation de Kolmogorov (1940)
108. Doob, J.L.: Stochastic Processes. Wiley, New York (1953)
109. Dowd, K.: Beyond Value at Risk: The New Science of Risk Management. Wiley, Chichester (1998)
110. Duffie, D.: Dynamic Asset Pricing Theory, 2nd edn. Princeton University Press, Princeton (1996)
111. Duffie, D., Pan, J., Singleton, K.: Transform analysis and asset pricing for affine jump-diffusions. Econometrica **68**, 1343–1376 (2000)
112. Dupire, B.: Pricing with a smile. Risk **7**, 18–20 (1994)
113. Eberlein, E., Frey, R., Kalkbrener, M., Overbeck, L.: Mathematics in financial risk management. Jahresber. DMV **109**, 165–193 (2007)
114. Eberlein, E., Keller, U.: Hyperbolic distributions in finance. Bernoulli **1**, 281–299 (1995)
115. Egloff, D.: Monte Carlo algorithms for optimal stopping and statistical learning. Ann. Appl. Probab. **15**, 1396–1432 (2005)
116. Ehrhardt, M. (ed.): Nonlinear Models in Mathematical Finance. New Research Trends in Option Pricing. Nova Science, Hauppauge (2008)
117. Ekström, E., Lötstedt, P., Tysk, J.: Boundary values and finite difference methods for the single factor term structure equation. Appl. Math. Finance **16**, 253–259 (2009)
118. El Karoui, N., Jeanblanc-Picqué, M., Shreve, S.E.: Robustness of the Black and Scholes formula. Math. Finance **8**, 93–126 (1998)
119. Elliott, C.M., Ockendon, J.R.: Weak and Variational Methods for Moving Boundary Problems. Pitman, Boston (1982)
120. Elliott, R.J., Kopp, P.E.: Mathematics of Financial Markets. Springer, New York (1999)
121. Embrechts, P., Klüppelberg, C., Mikosch, T.: Modelling Extremal Events. Springer, Berlin (1997)
122. Ender, M.: Model risk in option pricing. www.risknet.de/risknet-elibrary/kategorien/market-risk (2008)
123. Epps, T.W.: Pricing Derivative Securities. World Scientific, Singapore (2000)
124. Faigle, U., Schrader, R.: On the Convergence of Stationary Distributions in Simulated Annealing Algorithms. Inf. Process. Lett. **27**, 189–194 (1988)
125. Fang, F., Oosterlee, C.W.: A novel option pricing method based on Fourier-cosine series expansions. SIAM J. Sci. Comput. **31**, 826–848 (2008)
126. Fang, F., Oosterlee, C.W.: Pricing early-exercise and discrete barrier options by fourier-cosine series expansions. Numer. Math. **114**, 27–62 (2009)
127. Feller, W.: An Introduction to Probability Theory and Its Applications. Wiley, New York (1950)
128. Fengler, M.R.: Semiparametric Modeling of Implied Volatility. Springer, Berlin (2005)
129. Figlewski, S., Gao, B.: The adaptive mesh model: a new approach to efficient option pricing. J. Financ. Econ. **53**, 313–351 (1999)
130. Fishman, G.S.: Monte Carlo. Concepts, Algorithms, and Applications. Springer, New York (1996)
131. Fisz, M.: Probability Theory and Mathematical Statistics. Wiley, New York (1963)
132. Föllmer, H., Schied, A.: Stochastic Finance: An Introduction to Discrete Time. de Gruyter, Berlin (2002)
133. Forsyth, P.A., Vetzal, K.R.: Quadratic convergence for valuing American options using a penalty method. SIAM J. Sci. Comput. **23**, 2095–2122 (2002)

134. Forsyth, P.A., Vetzal, K.R.: Numerical methods for nonlinear PDEs in finance. In: Duan, J.-C., Härdle, W.K., Gentle, J.E. (eds.) Handbook of Computational Finance, pp. 503–528. Springer, Berlin (2012)
135. Forsyth, P.A., Vetzal, K.R., Zvan, R.: A finite element approach to the pricing of discrete lookbacks with stochastic volatility. Appl. Math. Finance **6**, 87–106 (1999)
136. Forsyth, P.A., Vetzal, K.R., Zvan, R.: Convergence of numerical methods for valuing path-dependent options using interpolation. Rev. Deriv. Res. **5**, 273–314 (2002)
137. Fournié, E., Lasry, J.-M., Lebuchoux, J., Lions, P.-L., Touzi, N.: An application of Malliavin calculus to Monte Carlo methods in finance. Finance Stochast. **3**, 391–412 (1999)
138. Franke, J., Härdle, W., Hafner, C.M.: Statistics of Financial Markets. Springer, Berlin (2004)
139. Freedman, D.: Brownian Motion and Diffusion. Holden Day, San Francisco (1971)
140. Frey, R., Patie, P.: Risk management for derivatives in illiquid markets: a simulation-study. In: Sandmann, K., Schönbucher, P. (eds.) Advances in Finance and Stochastics. Springer, Berlin (2002)
141. Frey, R., Stremme, A.: Market volatility and feedback effects from dynamic hedging. Math. Finance **7**, 351–374 (1997)
142. Frutos, J. de: A spectral method for bonds. Comput. Oper. Res. **35**, 64–75 (2008)
143. Fu, M.C., et al.: Pricing American options: a comparison of Monte Carlo simulation approaches. J. Comput. Finance **4**(3), 39–88 (2001)
144. Fusai, G., Sanfelici, S., Tagliani, A.: Practical problems in the numerical solution of PDEs in finance. Rend. Studi Econ. Quant. **2001**, 105–132 (2002)
145. Gander, M.J., Wanner, G.: From Euler, Ritz, and Galerkin to modern computing. SIAM Rev. **54**, 627–666 (2012)
146. Geman, H., et al., eds.: Mathematical Finance. Bachelier Congress 2000. Springer, Berlin (2002)
147. Gentle, J.E.: Random Number Generation and Monte Carlo Methods. Springer, New York (1998)
148. Gerstner, T., Griebel, M.: Numerical integration using sparse grids. Numer. Algorithms **18**, 209–232 (1998)
149. Gerstner, T., Griebel, M.: Dimension-adaptive tensor-product quadrature. Computing **71**, 65–87 (2003)
150. Geske, R., Johnson, H.E.: The American put option valued analytically. J. Finance **39**, 1511–1524 (1984)
151. Giles, M.: Variance reduction through multilevel Monte Carlo path calculations. In: Appleby, J.A.D., et al. (eds.) Numerical Methods for Finance. Chapman & Hall, Boca Raton (2008)
152. Giles, M., Glasserman, P.: Smoking adjoints: fast Monte Carlo methods. Risk **19**, 88–92 (2006)
153. Gilks, W.R., Richardson, S., Spiegelhalter, D.J. (eds.): Markov Chain Monte Carlo in Practice. Chapman & Hall, Boca Raton (1996)
154. Glaser, J., Heider, P.: Arbitrage-free approximation of call price surfaces and input data risk. Quant. Finance **12**, 61–73 (2012). doi:10.1080/14697688.2010.514005
155. Glasserman, P.: Monte Carlo Methods in Financial Engineering. Springer, New York (2004)
156. Glover, K.J., Duck, P.W., Newton, D.P.: On nonlinear models of markets with finite liquidity: some cautionary notes. SIAM J. Appl. Math. **70**, 3252–3271 (2010)
157. Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn. The John Hopkins University Press, Baltimore (1996)
158. Goodman, J., Ostrov, D.N.: On the early exercise boundary of the American put option. SIAM J. Appl. Math. **62**, 1823–1835 (2002)
159. Grandits, P.: Frequent hedging under transaction costs and a nonlinear Fokker-Planck PDE. SIAM J. Appl. Math. **62**, 541–562 (2001)
160. Grüne, L., Kloeden, P.E.: Pathwise approximation of random ODEs. BIT **41**, 710–721 (2001)
161. Hackbusch, W.: Multi-Grid Methods and Applications. Springer, Berlin (1985)
162. Hackbusch, W.: Elliptic Differential Equations: Theory and Numerical Treatment. Springer Series in Computational Mathematics, vol. 18. Berlin, Springer (1992)

163. Haentjens, T., in 't Hout, K.: ADI finite difference discretization of the Heston-Hull-White PDE. In: Simos, T.E., et al. (eds.) Numerical Analysis and Applied Mathematics. AIP Conference Proceedings, vol. 1281, pp. 1995–1999 (2010)
164. Häggström, O.: Finite Markov Chains and Algorithmic Applications. Cambridge University Press, Cambridge (2002)
165. Hairer, E., Nørsett, S.P., Wanner, G.: Solving Ordinary Differential Equations I. Nonstiff Problems. Springer, Berlin (1993)
166. Halton, J.H.: On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. Numer. Math. **2**, 84–90 (1960)
167. Hammersley, J.M., Handscomb, D.C.: Monte Carlo Methods. Methuen, London (1964)
168. Hämmerlin, G., Hoffmann, K.-H.: Numerical Mathematics. Springer, Berlin (1991)
169. Han, H., Wu, X.: A fast numerical method for the Black–Scholes equation of American options. SIAM J. Numer. Anal. **41**, 2081–2095 (2003)
170. Harrison, J.M., Pliska, S.R.: Martingales and stochastic integrals in the theory of continuous trading. Stoch. Process. Appl. **11**, 215–260 (1981)
171. Hart, J.F.: Computer Approximations. Wiley, New York (1968)
172. Haug, E.G.: The Complete Guide to Option Pricing Formulas, 2nd edn. 2007. McGraw-Hill, New York (1998)
173. He, C., Kennedy, J.S., Coleman, T., Forsyth, P.A., Li, Y., Vetzal, K.: Calibration and hedging under jump diffusion. Rev. Deriv. Res. **9**, 1–35 (2006)
174. Heider, P.: A condition number for the integral representation of American options. J. Comput. Finance **11**(2), 95–103 (2007/08)
175. Heider, P.: A second-order Nyström-type discretization for the early-exercise curve of American put options. Int. J. Comput. Math. **86**, 982–991 (2009)
176. Heider, P.: Numerical methods for non-linear Black–Scholes equations. Appl. Math. Finance **17**, 59–81 (2010)
177. Heider, P., Schaeling, D.: Numerical methods for American options in nonlinear Black–Scholes models. Preprint, Universität Köln (2010)
178. Heston, S.L.: A closed-form solution for options with stochastic volatility with applications to bond and currency options. Rev. Financ. Stud. **6**, 327–343 (1993)
179. Heston, S., Zhou, G.: On the rate of convergence of discrete-time contingent claims. Math. Finance **10**, 53–75 (2000)
180. Higham, D.J.: An algorithmic introduction to numerical solution of stochastic differential equations. SIAM Rev. **43**, 525–546 (2001)
181. Higham, D.J.: An Introduction to Financial Option Valuation. Cambridge University Press, Cambridge (2004)
182. Higham, D.J., Kloeden, P.E.: Numerical methods for nonlinear stochastic differential equations with jumps. Numer. Math. **101**, 101–119 (2005)
183. Higham, N.J.: Accuracy and Stability of Numerical Algorithms. SIAM, Philadelphia (1996)
184. Higham, N.J.: Computing the nearest correlation matrix — a problem from finance. IMA J. Numer. Anal. **22**, 329–343 (2002)
185. Hilber, N., Matache, A.-M., Schwab, C.: Sparse wavelet methods for option pricing under stochastic volatility. J. Comput. Finance **8**(4), 1–42 (2005)
186. Hofmann, N., Platen, E., Schweizer, M.: Option pricing under incompleteness and stochastic volatility. Math. Finance **2**, 153–187 (1992)
187. Hoggard, T., Whalley, A.E., Wilmott, P.: Hedging option portfolios in the presence of transaction costs. Adv. Futur. Options Res. **7**, 21–35 (1994)
188. Holmes, A.D., Yang, H.: A front-fixing finite element method for the valuation of American options. SIAM J. Sci. Comput. **30**, 2158–2180 (2008)
189. Honoré, P., Poulsen, R.: Option pricing with EXCEL. In: Nielsen, S. (ed.): Programming Languages and Systems in Computational Economics and Finance, pp. 369–402. Kluwer, Amsterdam (2002)
190. Huang, J.-Z., Subrahmanyam, M.G., Yu, G.G.: Pricing and hedging American options: a recursive integration method. Rev. Financ. Stud. **9**, 227–300 (1996)

191. Hull, J.C.: Options, Futures, and Other Derivatives, 4th edn. Prentice Hall, Upper Saddle River (2000)
192. Hull, J., White, A.: The use of the control variate technique in option pricing. J. Financ. Quant. Anal. **23**, 237–251 (1988)
193. Hunt, P.J., Kennedy, J.E.: Financial Derivatives in Theory and Practice. Wiley, Chichester (2000)
194. Ikonen, S., Toivanen, J.: Pricing American options using LU decomposition. Appl. Math. Sci. **1**, 2529–2551 (2007)
195. Ikonen, S., Toivanen, J.: Operator splitting methods for pricing American options under stochastic volatility. Numer. Math. **113**, 299–324 (2009)
196. Ingersoll, J.E.: Theory of Financial Decision Making. Rowmann and Littlefield, Savage (1987)
197. Int-Veen, R.: Avoiding numerical dispersion in option valuation. Report Universität Köln 2002; Comput. Vis. Sci. **10**, 185–195 (2007)
198. Isaacson, E., Keller, H.B.: Analysis of Numerical Methods. Wiley, New York (1966)
199. Jacod, J., Protter, P.: Probability Essentials, 2nd edn. Springer, Berlin (2003)
200. Jäckel, P.: Monte Carlo Methods in Finance. Wiley, Chichester (2002)
201. Jaillet, P., Lamberton, D., Lapeyre, B.: Variational inequalities and the pricing of American options. Acta Appl. Math. **21**, 263–289 (1990)
202. Jamshidian, F.: An analysis of American options. Rev. Futur. Mark. **11**, 72–80 (1992)
203. Jiang, L., Dai, M.: Convergence of binomial tree method for European/American path-dependent options. SIAM J. Numer. Anal. **42**, 1094–1109 (2004)
204. Johnson, H.E.: An analytic approximation for the American put price. J. Financ. Quant. Anal. **18**, 141–148 (1983)
205. Jonen, C.: An efficient implementation of a least-squares Monte Carlo method for valuing American-style options. Int. J. Comput. Math. **86**, 1024–1039 (2009)
206. Jonen, C.: Efficient Pricing of High-Dimensional American-Style Derivatives: A Robust Regression Monte Carlo method. PhD dissertation, Universität Köln (2011). http://kups.ub.uni-koeln.de/4442
207. Joshi, M.S.: The Concepts and Practice of Mathematical Finance. Cambridge University Press, Cambridge (2003)
208. Ju, N.: Pricing an American option by approximating its early exercise boundary as a multipiece exponential function. Rev. Financ. Stud. **11**, 627–646 (1998)
209. Kaebe, C., Maruhn, J.H., Sachs, E.W.: Adjoint-based Monte Carlo calibration of financial market models. Finance Stochast. **13**, 351–379 (2009)
210. Kahaner, D., Moler, C., Nash, S.: Numerical Methods and Software. Prentice Hall Series in Computational Mathematics. Prentice Hall, Englewood Cliffs (1989)
211. Kallast, S., Kivinukk, A.: Pricing and hedging American options using approximations by Kim integral equations. Eur. Finance Rev. **7**, 361–383 (2003)
212. Kallsen, J.: A didactic note on affine stochastic volatility models. In: Kabanov, Y., et al. (eds.) From Stochastic Calculus to Mathematical Finance. Springer, Berlin (2006)
213. Kamrad, B., Ritchken, P.: Multinomial approximating models for options with k state variables. Manag. Sci. **37**, 1640–1652 (1991)
214. Kangro, R., Nicolaides, R.: Far field boundary conditions for Black-Scholes equations. SIAM J. Numer. Anal. **38**, 1357–1368 (2000)
215. Kantorovich, L.W., Akilov, G.P.: Functional Analysis in Normed Spaces. Pergamon Press, Elmsford (1964)
216. Karatzas, I., Shreve, S.E.: Brownian Motion and Stochastic Calculus, 2nd edn. Springer Graduate Texts. Springer, New York (1991)
217. Karatzas, I., Shreve, S.E.: Methods of Mathematical Finance. Springer, New York (1998)
218. Kat, H.M.: Pricing Lookback options using binomial trees: an evaluation. J. Financ. Eng. **4**, 375–397 (1995)
219. Kebaier, A.: Statistical Romberg extrapolation: a new variance reduction method and applications to option pricing. Ann. Appl. Probab. **15**, 2681–2705 (2005)

220. Kemna, A.G.Z., Vorst, A.C.F.: A pricing method for options based on average asset values. J. Bank. Finance **14**, 113–129 (1990)
221. Khaliq, A.Q.M., Voss, D.A., Yousuf, M.: Pricing exotic options with L-stable Padé schemes. J. Bank. Finance **31**, 3438–3461 (2007)
222. Kim, J.: The analytic valuation of American options. Rev. Financ. Stud. **3**, 547–572 (1990)
223. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**, 671–680 (1983)
224. Klassen, T.R.: Simple, fast and flexible pricing of Asian options. J. Comput. Finance **4**(3), 89–124 (2001)
225. Kloeden, P.E., Platen, E.: Numerical Solution of Stochastic Differential Equations. Springer, Berlin (1992)
226. Knuth, D.: The Art of Computer Programming, vol. 2. Addison-Wesley, Reading (1995)
227. Kocis, L., Whiten, W.J.: Computational investigations of low-discrepancy sequences. ACM Trans. Math. Softw. **23**, 266–294 (1997)
228. Korn, R., Müller, S.: The decoupling approach to binomial pricing of multi-asset options. J. Comput. Finance **12**(3), 1–30 (2009)
229. Kou, S.G.: A jump diffusion model for option pricing. Manag. Sci. **48**, 1086–1101 (2002)
230. Kovalov, P., Linetsky, V., Marcozzi, M.: Pricing multi-asset American options: a finite element method-of-lines with smooth penalty. J. Sci. Comput. **33**, 209–237 (2007)
231. Kreiss, H.O., Thomée, V., Widlund, O.: Smoothing of initial data and rates of convergence for parabolic difference equations. Commun. Pure Appl. Math. **23**, 241–259 (1970)
232. Kröner, D.: Numerical Schemes for Conservation Laws. Wiley, Chichester (1997)
233. Krylov, N.V.: Controlled Diffusion Processes. Springer, Heidelberg (1980)
234. Kwok, Y.K.: Mathematical Models of Financial Derivatives. Springer, Singapore (1998)
235. Kwok, Y.K., Leung, K.S., Wong, H.Y.: Efficient options pricing using the Fast Fourier Transform. In: Duan, J.-C., Härdle, W.K., Gentle, J.E. (eds.) Handbook of Computational Finance, pp. 579–604. Springer, Berlin (2012)
236. Lambert, J.D.: Numerical Methods for Ordinary Differential Systems. The Initial Value Problem. Wiley, Chichester (1991)
237. Lamberton, D., Lapeyre, B.: Introduction to Stochastic Calculus Applied to Finance. Chapman & Hall, London (1996)
238. Lange, K.: Numerical Analysis for Statisticians. Springer, New York (1999)
239. L'Ecuyer, P.: Tables of linear congruential generators of different sizes and good lattice structure. Math. Comput. **68**, 249–260 (1999)
240. Leentvaar, C.C.W., Oosterlee, C.W.: On coordinate transformation and grid stretching for sparse grid pricing of basket options. J. Comput. Math. **222**, 193–209 (2008)
241. Lehn, J.: Random number generators. GAMM-Mitteilungen **25**, 35–45 (2002)
242. Leisen, D.P.J.: Pricing the American put option: a detailed convergence analyis for binomial models. J. Econ. Dyn. Control **22**, 1419–1444 (1998)
243. Leisen, D.P.J.: The random-time binomial model. J. Econ. Dyn. Control **23**, 1355–1386 (1999)
244. Leisen, D.P.J., Reimer, M.: Binomial models for option valuation – examining and improving convergence. Appl. Math. Finance **3**, 319–346 (1996)
245. Leland, H.E.: Option pricing and replication with transaction costs. J. Finance **40**, 1283–1301 (1985)
246. Levy, G.: Computational finance using C and C#. Elsevier, Amsterdam (2008)
247. Longstaff, F.A., Schwartz, E.S.: Valuing American options by simulation: a simple least-squares approach. Rev. Financ. Stud. **14**, 113–147 (2001)
248. Lord, R., Fang, F., Bervoets, F., Oosterlee, C.W.: A fast and accurate FFT-based method for pricing early-exercise options under Lévy processes. SIAM J. Sci. Comput. **30**, 1678–1705 (2008)
249. Lux, T.: The socio-economic dynamics of speculative markets: interacting agents, chaos, and the fat tails of return distributions. J. Econ. Behav. Organ. **33**, 143–165 (1998)

250. Lyons, T.J.: Uncertain volatility and the risk-free synthesis of derivatives. Appl. Math. Finance **2**, 117–133 (1995)
251. Lyuu, Y.-D.: Financial Engineering and Computation. Principles, Mathematics, Algorithms. Cambridge University Press, Cambridge (2002)
252. MacMillan, L.W.: Analytic approximation for the American put option. Adv. Futur. Opt. Res. **1**, 119–139 (1986)
253. Madan, D.B., Seneta, E.: The variance-gamma (V.G.) model for share market returns. J. Bus. **63**, 511–524 (1990)
254. Mainardi, R., Roberto, M., Gorenflo, R., Scalas, E.: Fractional calculus and continuous-time finance II: the waiting-time distribution. Physica A **287**, 468–481 (2000)
255. Maller, R.A., Solomon, D.H., Szimayer, A.: A multinomial approximation for American option prices in Lévy process models. Math. Finance **16**, 613–633 (2006)
256. Mandelbrot, B.B.: A multifractal walk down Wall Street. Sci. Am. **280**, 70–73 (1999)
257. Manteuffel, T.A., White, A.B., Jr.: The numerical solution of second-order boundary value problems on nonuniform meshes. Math. Comput. **47**, 511–535 (1986)
258. Marchesi, M., Cinotti, S., Focardi, S., Raberto, M.: Development and testing of an artificial stock market. In: Bischi, G.I. (ed.) Proceedings Urbino 2000 (2000)
259. Marsaglia, G.: Random numbers fall mainly in the planes. Proc. Natl. Acad. Sci. USA **61**, 23–28 (1968)
260. Marsaglia, G., Bray, T.A.: A convenient method for generating normal variables. SIAM Rev. **6**, 260–264 (1964)
261. Marsaglia, G., Tsang, W.W.: The ziggurat method for generating random variables. J. Stat. Softw. **5**(8), 1–7 (2000)
262. Mascagni, M.: Parallel pseudorandom number generation. SIAM News **32**, 5 (1999)
263. Matache, A.-M., von Petersdorff, T., Schwab, C.: Fast deterministic pricing of options on Lévy driven assets. Report 2002–11, Seminar for Applied Mathematics, ETH Zürich (2002)
264. Matsumoto, M., Nishimura, T.: Mersenne twister: a 623-dimensionally equidistributed uniform pseudorandom number generator. ACM Trans. Model. Comput. Simul. **8**, 3–30 (1998)
265. Mayo, A.: Fourth order accurate implicit finite difference method for evaluating American options. In: Proceedings of Computational Finance, London (2000)
266. McCarthy, L.A., Webber, N.J.: Pricing in three-factor models using icosahedral lattices. J. Comput. Finance **5**(2), 1–33 (2001/02)
267. McDonald, R.L., Schroder, M.D.: A parity result for American options. J. Comput. Finance **1**(3), 5–13 (1998)
268. Mel'nikov, A.V., Volkov, S.N., Nechaev, M.L.: Mathematics of Financial Obligations. American Mathematical Society, Providence (2002)
269. Merton, R.C.: Theory of rational option pricing. Bell J. Econ. Manag. Sci. **4**, 141–183 (1973)
270. Merton, R.: Option pricing when underlying stock returns are discontinous. J. Financ. Econ. **3**, 125–144 (1976)
271. Merton, R.C.: Continuous-Time Finance. Blackwell, Cambridge (1990)
272. Metwally, S.A.K., Atiya, A.: Using Brownian bridge for fast simulation of jump-diffusion processes and barrier options. J. Derv. **10**, 43–54 (2002)
273. Meyer, G.H.: Numerical Investigation of early exercise in American puts with discrete dividends. J. Comput. Finance **5**(2), 37–53 (2002)
274. Mikosch, T.: Elementary Stochastic Calculus, with Finance in View. World Scientific, Singapore (1998)
275. Mil'shtein, G.N.: Approximate integration of stochastic differential equations. Theory Probab. Appl. **19**, 557–562 (1974)
276. Milshtein, G.N.: A method of second-order accuracy integration of stochastic differential equations. Theory Probab. Appl. **23**, 396–401 (1978)
277. van Moerbeke, P.: On optimal stopping and free boundary problems. Rocky Mt. J. Math. **4**, 539–578 (1974)
278. Moro, B.: The full Monte. Risk **8**, 57–58 (1995)

279. Morokoff, W.J.: Generating quasi-random paths for stochastic processes. SIAM Rev. **40**, 765–788 (1998)
280. Morokoff, W.J., Caflisch, R.E.: Quasi-random sequences and their discrepancies. SIAM J. Sci. Comput. **15**, 1251–1279 (1994)
281. Morton, K.W.: Numerical Solution of Convection-Diffusion Problems. Chapman & Hall, London (1996)
282. Musiela, M., Rutkowski, M.: Martingale Methods in Financial Modelling, 2nd edn. 2005. Springer, Berlin (1997)
283. Neftci, S.N.: An Introduction to the Mathematics of Financial Derivatives. Academic Press, San Diego (1996)
284. Newton, N.J.: Continuous-time Monte Carlo methods and variance reduction. In: Rogers, L.C.G., Talay, D. (eds.) Numerical Methods in Finance, pp. 22–42. Cambridge University Press, Cambridge (1997)
285. Niederreiter, H.: Quasi-Monte Carlo methods and pseudo-random numbers. Bull. Am. Math. Soc. **84**, 957–1041 (1978)
286. Niederreiter, H.: Random Number Generation and Quasi-Monte Carlo Methods. Society for Industrial and Applied Mathematics, Philadelphia (1992)
287. Niederreiter, H., Jau-Shyong Shiue, P. (eds.): Monte Carlo and Quasi-Monte Carlo methods in scientific computing. In: Proceedings of a Conference at the University of Nevada, Las Vegas, Nevada, USA, 1994. Springer, New York (1995)
288. Nielsen, B.F., Skavhaug, O., Tveito, A.: Penalty and front-fixing methods for the numerical solution of American option problems. J. Comput. Finance **5**(4), 69–97 (2002)
289. Nielsen, B.F., Skavhaug, O., Tveito, A.: Penalty methods for the numerical solution of American multi-asset option problems. J. Comput. Appl. Math. **222**, 3–16 (2008)
290. Nielsen, L.T.: Pricing and Hedging of Derivative Securities. Oxford University Press, Oxford (1999)
291. Øksendal, B.: Stochastic Differential Equations. Springer, Berlin (1998)
292. Omberg, E.: The valuation of American put options with exponential exercise policies. Adv. Futur. Opt. Res. **2**, 117–142 (1987)
293. Oosterlee, C.W.: On multigrid for linear complementarity problems with application to American-style options. Electron. Trans. Numer. Anal. **15**, 165–185 (2003)
294. Panini, R., Srivastav, R.P.: Option pricing with Mellin transforms. Math. Comput. Model. **40**, 43–56 (2004)
295. Papageorgiou, A., Traub, J.F.: New results on deterministic pricing of financial derivatives. Columbia University Report CUCS-028-96 (1996)
296. Paskov, S., Traub, J.: Faster valuation of financial derivatives. J. Portf. Manag. **22**, 113–120 (1995)
297. Pelsser, A., Vorst, T.: The binomial model and the Greeks. J. Deriv. **1**, 45–49 (1994)
298. Peyret, R., Taylor, T.D.: Computational Methods for Fluid Flow. Springer, New York (1983)
299. Pham, H.: Optimal stopping, free boundary, and American option in a jump-diffusion model. Appl. Math. Optim. **35**, 145–164 (1997)
300. Pironneau, O., Hecht, F.: Mesh adaption for the Black & Scholes equations. East-West J. Numer. Math. **8**, 25–35 (2000)
301. Platen, E.: An introduction to numerical methods for stochastic differential equations. Acta Numer. **8**, 197–246 (1999)
302. Pliska, S.R.: Introduction to Mathematical Finance. Discrete Time Models. Blackwell, Malden (1997)
303. Pooley, D.M., Forsyth, P.A., Vetzal, K.R.: Numerical convergence properties of option pricing PDEs with uncertain volatility. IMA J. Numer. Anal. **23**, 241–267 (2003)
304. Pooley, D.M., Forsyth, P.A., Vetzal, K., Simpson, R.B.: Unstructured meshing for two asset barrier options. Appl. Math. Finance **7**, 33–60 (2000)
305. Pooley, D.M., Vetzal, K.R., Forsyth, P.A.: Convergence remedies for non-smooth payoffs in option pricing. J. Comput. Finance **6**(4), 25–40 (2003)

306. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes in FORTRAN. The Art of Scientific Computing, 2nd edn. Cambridge University Press, Cambridge (1992)
307. Protter, P.E.: Stochastic Integration and Differential Equations. Springer, Berlin (2004)
308. Quarteroni, A., Sacco, R., Saleri, F.: Numerical Mathematics. Springer, New York (2000)
309. Quecke, S.: Efficient numerical methods for pricing American options under Lévy models. PhD-dissertation, Universität Köln (2007). http://kups.ub.uni-koeln.de/2018
310. Rannacher, R.: Finite element solution of diffusion problems with irregular data. Numer. Math. **43**, 309–327 (1984)
311. Rebonato, R.: Interest-Rate Option Models: Understanding, Analysing and Using Models for Exotic Interest-Rate Options. Wiley, Chichester (1996)
312. Reisinger, C.: Numerische Methoden für hochdimensionale parabolische Gleichungen am Beispiel von Optionspreisaufgaben. PhD Thesis, Universität Heidelberg (2004)
313. Rendleman, R.J., Bartter, B.J.: Two-state option pricing. J. Finance **34**, 1093–1110 (1979)
314. Revuz, D., Yor, M.: Continuous Martingales and Brownian Motion. Springer, Berlin (1991)
315. Ribeiro, C., Webber, N.: A Monte Carlo method for the normal inverse Gaussian option valuation model using an inverse Gaussian bridge. Working paper, City University, London (2002)
316. Ribeiro, C., Webber, N.: Valuing path dependent options in the variance-gamma model by Monte Carlo with a gamma bridge. J. Comput. Finance **7**(2), 81–100 (2003/04)
317. Ripley, B.D.: Stochastic Simulation. Wiley Series in Probability and Mathematical Statistics. Wiley, New York (1987)
318. Risken, H.: The Fokker-Planck Equation. Springer, Berlin (1989)
319. Rogers, L.C.G.: Monte Carlo valuation of American options. Math. Finance **12**, 271–286 (2002)
320. Rogers, L.C.G., Shi, Z.: The value of an Asian option. J. Appl. Probab. **32**, 1077–1088 (1995)
321. Rogers, L.C.G., Talay, D. (eds.): Numerical Methods in Finance. Cambridge University Press, Cambridge (1997)
322. Rubinstein, M.: Implied binomial trees. J. Finance **69**, 771–818 (1994)
323. Rubinstein, M.: Return to oz. Risk **7**(11), 67–71 (1994)
324. Rubinstein, R.Y.: Simulation and the Monte Carlo Method. Wiley, New York (1981)
325. Ruppert, D.: Statistics and Finance. An Introduction. Springer, New York (2004)
326. Saad, Y.: Iterative Methods for Sparse Linear Systems, 2nd edn. SIAM, Philadelphia (2003)
327. Saito, Y., Mitsui, T.: Stability analysis of numerical schemes for stochastic differential equations. SIAM J. Numer. Anal. **33**, 2254–2267 (1996)
328. Sato, K.-I.: Lévy Processes and Infinitely Divisible Distributions. Cambridge University Press, Cambridge (1999)
329. Schöbel, R., Zhu, J.: Stochastic volatility with an Ornstein-Uhlenbeck process: an extension. Eur. Finance Rev. **3**(1), 23–46 (1999)
330. Schönbucher, P.J., Wilmott, P.: The feedback effect of hedging in illiquid markets. SIAM J. Applied Mathematics **61**, 232–272 (2000)
331. Schoenmakers, J.G.M., Heemink, A.W.: Fast Valuation of Financial Derivatives. J. Comput. Finance **1**, 47–62 (1997)
332. Schoutens, W.: Lévy Processes in Finance. Wiley, Chichester (2003)
333. Schuss, Z.: Theory and Applications of Stochastic Differential Equations. Wiley Series in Probability and Mathematical Statistics. Wiley, New York (1980)
334. Schwarz, H.R.: Numerical Analysis. Wiley, Chichester (1989)
335. Schwarz, H.R.: Methode der finiten Elemente. Teubner, Stuttgart (1991)
336. Seydel, R.: Practical Bifurcation and Stability Analysis, 3rd edn. Springer Interdisciplinary Applied Mathematics, vol. 5. Springer, New York (2010)
337. Seydel, R.U.: Lattice approach and implied trees. In: Duan, J.-C., Härdle, W.K., Gentle, J.E. (eds.) Handbook of Computational Finance, pp. 551–577. Springer, Berlin (2012)
338. Seydel, R.U.: Risk and computation. In: Glau, K., Scherer, M., Zagst, R. (eds.) Innovations in Quantitative Risk Management, pp. 305–316. Springer, Heidelberg (2015)

339. Shiryaev, A.N.: Essentials of Stochastic Finance. Facts, Models, Theory. World Scientific, Singapore (1999)
340. Shreve, S.E.: Stochastic Calculus for Finance II. Continuous-Time Models. Springer, New York (2004)
341. Smith, G.D.: Numerical Solution of Partial Differential Equations: Finite Difference Methods, 2nd edn. Clarendon Press, Oxford (1978)
342. Smithson, C.: Multifactor options. Risk **10**(5), 43–45 (1997)
343. Spanier, J., Maize, E.H.: Quasi-random methods for estimating integrals using relatively small samples. SIAM Rev. **36**, 18–44 (1994)
344. Stauffer, D.: Percolation models of financial market dynamics. Adv. Complex Syst. **4**, 19–27 (2001)
345. Steele, J.M.: Stochastic Calculus and Financial Applications. Springer, New York (2001)
346. Steiner, M., Wallmeier, M., Hafner, R.: Baumverfahren zur Bewertung diskreter Knock-Out-Optionen. OR Spektrum **21**, 147–181 (1999)
347. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis. Springer, Berlin (1996)
348. Stoer, J., Witzgall, C.: Convexity and Optimization in Finite Dimensions I. Springer, Berlin (1970)
349. Stojanovic, S.: Computational Financial Mathematics Using MATHEMATICA. Birkhäuser, Boston (2003)
350. Strang, G.: Computational Science and Engineering. Wellesley, Cambridge (2007)
351. Strang, G., Fix, G.: An Analysis of the Finite Element Method. Prentice-Hall, Englewood Cliffs (1973)
352. Sweby, P.K.: High resolution schemes using flux limiters for hyperbolic conservation laws. SIAM J. Numer. Anal. **21**, 995–1011 (1984)
353. Tavella, D., Randall, C.: Pricing Financial Instruments. The Finite Difference Method. Wiley, New York (2000)
354. Tezuka, S.: Uniform Random Numbers: Theory and Practice. Kluwer, Dordrecht (1995)
355. Thomas, D.B., Luk, W., Leong, P.H.W., Villasenor, J.D.: Gaussian random number generators. ACM Comput. Surv. **39**(4), Article 11 (2007)
356. Thomas, J.W.: Numerical Partial Differential Equations: Finite Difference Methods. Springer, New York (1995)
357. Thomas, J.W.: Numerical Partial Differential Equations. Conservation Laws and Elliptic Equations. Springer, New York (1999)
358. Tian, Y.: A modified lattice approach to option pricing. J. Futur. Mark. **13**, 563–577 (1993)
359. Tian, Y.: A flexible binomial option pricing model. J. Futur. Mark. **19**, 817–843 (1999)
360. Tilley, J.A.: Valuing American options in a path simulation model. Trans. Soc. Actuar. **45**, 83–104 (1993)
361. Topper, J.: Finite element modeling of exotic options. In: OR Proceedings 1999, pp. 336–341 (2000)
362. Topper, J.: Financial Engineering with Finite Elements. Wiley, New York (2005)
363. Traub, J.F., Wozniakowski, H.: The Monte Carlo algorithm with a pseudo-random generator. Math. Comput. **58**, 323–339 (1992)
364. Trottenberg, U., Oosterlee, C., Schüller, A.: Multigrid. Academic Press, San Diego (2001)
365. Tsay, R.S.: Analysis of Financial Time Series. Wiley, New York (2002)
366. van der Vorst, H.A.: Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. SIAM J. Sci. Stat. Comput. **13**, 631–644 (1992)
367. Varga, R.S.: Matrix Iterative Analysis. Prentice Hall, Englewood Cliffs (1962)
368. Vellekoop, M.H., Nieuwenhuis, J.W.: Efficient pricing of derivatives on assets with discrete dividends. Appl. Math. Finance **13**, 265–284 (2006)
369. Vichnevetsky, R.: Computer Methods for Partial Differential Equations. Volume I. Prentice-Hall, Englewood Cliffs (1981)
370. Villeneuve, S., Zanette, A.: Parabolic ADI methods for pricing American options on two stocks. Math. Oper. Res. **27**, 121–149 (2002)
371. Vretblad, A.: Fourier Analysis and Its Applications. Springer, New York (2003)

372. Wallace, C.S.: Fast pseudorandom numbers for normal and exponential variates. ACM Trans. Math. Softw. **22**(1), 119–127 (1996)
373. Wang, X., Phillips, P.C.B., Yu, J.: Bias in estimating multivariate and univariate diffusion. J. Econ. **161**, 228–245 (2011)
374. Wesseling, P.: Principles of Computational Fluid Dynamics. Springer, Berlin (2001)
375. Wilmott, P.: Derivatives. Wiley, Chichester (1998)
376. Wilmott, P., Dewynne, J., Howison, S.: Option Pricing. Mathematical Models and Computation. Oxford Financial Press, Oxford (1996)
377. Wloka, J.: Partial Differential Equations. Cambridge University Press, Cambridge (1987)
378. Zagst, R.: Interest-Rate Management. Springer, Berlin (2002)
379. Zhang, J.E.: A semi-analytical method for pricing and hedging continuously sampled arithmetic average rate options. J. Comput. Finance **5**(1), 59–79 (2001)
380. Zhao, Y., Ziemba, W.T.: Hedging errors with Leland's option model in the presence of transaction costs. Finance Res. Lett. **4**, 49–58 (2007)
381. Zhu, Y.-I., Wu, X., Chern, I.-L.: Derivative Securities and Difference Methods. Springer, New York (2004)
382. Zienkiewicz, O.C.: The Finite Element Method in Engineering Science. McGraw-Hill, London (1977)
383. Zvan, R., Forsyth, P.A., Vetzal, K.R.: Robust numerical methods for PDE models of Asian options. J. Comput. Finance **1**(2), 39–78 (1997/98)
384. Zvan, R., Forsyth, P.A., Vetzal, K.R.: Penalty methods for American options with stochastic volatility. J. Comput. Appl. Math. **91**, 199–218 (1998)
385. Zvan, R., Forsyth, P.A., Vetzal, K.R.: Discrete Asian barrier options. J. Comput. Finance **3**(1), 41–67 (1999)
386. Zvan, R., Vetzal, K.R., Forsyth, P.A.: PDE methods for pricing barrier options. J. Econ. Dyn. Control **24**, 1563–1590 (2000)

Index

R.U. Seydel, *Tools for Computational Finance*, Universitext,
DOI 10.1007/978-1-4471-7338-0