Acta Numerica (2012), pp. 577–671
doi:10.1017/S0962492912000074
© Cambridge University Press, 2012
Printed in the United Kingdom

Pricing futures by deterministic methods*

Olivier Pironneau
Université Pierre et Marie Curie,
UMR CNRS 7598 Laboratoire Jacques Louis Lions,
F-75252 Paris, France
E-mail: olivier.pironneau@upmc.fr

In this article we will focus on only a small part of financial mathematics, namely the use of partial differential equations for pricing futures. Even within this narrow range it is hard to be systematic and complete, or even to do better than existing books such as Wilmott, Howison and Dewynne (1995), Achdou and Pironneau (2005), or software manuals such as Lapeyre, Martini and Sulem (2010). So this article may be valuable only to the extent that it reflects ten years of teaching, conferences and interaction with the protagonists of financial mathematics.

Also, because the theory of partial differential equations is not always well known, we have chosen a pragmatic approach and left out the details of the theory or the proofs of some results, and refer the reader to other books. The numerical algorithms, on the other hand, are given in detail.

CONTENTS

Introduction 579

PART 1: Modelling financial assets and derivatives

1.1 Financial assets 580
1.2 Models with local volatilities 582
1.3 Stochastic volatility models 583
1.4 Models with jump-diffusion processes 584
1.5 Financial derivatives 586
1.6 Conclusion 595

* Colour online for monochrome figures available at journals.cambridge.org/anu.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

PART 2: SDE-based numerical methods

2.1 Principles of numerical simulation by Monte Carlo 596
2.2 The Longstaff–Schwartz algorithm for Bermudas 598
2.3 Methods using the process PDF for basket options 599
2.4 A binomial tree for American options 600

PART 3: Some partial differential equations in finance

3.1 From SDEs to PDEs and PIDEs 601
3.2 Fokker–Planck and Kolmogorov equations 601
3.3 Some existence and regularity results 610

PART 4: Numerical methods

4.1 A finite difference method for the Black–Scholes PDE 616
4.2 Spectral methods 620
4.3 Finite element methods 621
4.4 Variational methods for American options 625
4.5 A stochastic volatility model solved by FEM 629
4.6 Asian puts 631
4.7 Automatically adapted meshes 633
4.8 Greeks 635
4.9 Reduced-order modelling 637

PART 5: Calibration

5.1 Introduction 645
5.2 Dupire's equation 646
5.3 Calibration of the local volatility 651
5.4 Calibration of implied volatility 654
5.5 Calibration on a reduced basis 656

PART 6: Open-source tools

6.1 Premia 656
6.2 QuantLib 658
6.3 FreeFem++ 659
6.4 CUDA and other tools 664

Conclusion 667

Acknowledgements 667

References 668

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Introduction

Except for the pioneering work of Louis Bachelier (1900), financial mathematics is relatively new. Perhaps the right landmarks are the paper by Black and Scholes (1973) and the work of Merton (1976), which led to their Nobel Prize. Forty years later, because of the explosion of applications, more and more mathematicians, physicists, economists and computer scientists have joined the fun. Despite several global crises the subject is exploding, and the literature around it.

Banks, insurance companies, *etc.*, have also been very innovative in inventing new contingent claims, and since no one sees the global picture, every day there seems to be a new challenge to solve. The field is huge: the derivatives market is worth many times the value of the underlying assets. The domain is very lively, with career opportunities unheard of in other fields of applied mathematics, and it attracts students too, because there is plenty to do and still many jobs, despite systemic instabilities.

It is true that the field of financial mathematics is in its infancy, the models are crude and focused on a small set of futures; global models are just being initiated with game theory and utility functions (Lions and Lasry 2007). The lack of global perspective among scientists and their inability to predict systemic risks have given 'quants' a bad name. A French politician has even accused financial mathematicians of 'crimes against humanity'. Let's be reasonable: if there is a culprit it is the internet. It has given 'banksters' a Kalashnikov – electronic trading – and mathematicians are only providing the system with tiny bullet-proof shields.

Futures are modelled by using advanced stochastic tools and then priced using simple Monte Carlo algorithms or analytical formulae of limited application. But the days of closed-form solutions are numbered because models are increasingly complex. Numerical algorithms will have to be more efficient because speed is becoming a dominant factor with high-frequency trading: 40 % of transactions are done automatically by 'robots', computer programs taking instant decisions based on overnight calculations and in-house know-how. Some banks currently double their computing facilities every six months and try to shorten the wires between their IT centres and stock markets to improve transaction speed (this is not a joke). At the current rate they will soon be the biggest users of high-performance computing (HPC).

Monte Carlo is often slow, but it is akin to the stochastic models and it is highly parallel and easy to implement in hybrid computers (GPUs). Alternatives are binomial or trinomial trees, fast-Fourier-transform-based methods, when the characteristic functions of the stochastic processes are known, and the partial differential equations derived via Itô's lemma.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Most option pricing problems can be formulated as partial differential equations (PDEs), partial integro-differential equations (PIDEs) or partial differential inequalities (PDIs). They result in clean and fast algorithms but are cursed by the dimensionality of the problem and hard parallelization procedures.

As a coauthor of Achdou and Pironneau (2005, 2009) I do not intend to present the same material here or to present it in the same way. Here mathematical details are not given, only the essence of the methods, the more recent developments and, in more detail, the numerical algorithms.

The material is divided into six parts.

- Part 1 starts with an introduction to financial assets, then a presentation of some of the models, followed by a list of the most popular derivatives.
- Part 2 begins with a short section on Monte Carlo methods and then deals with deterministic methods based on the knowledge of the probability density of the processes, or their characteristic functions.
- In Part 3 we shall derive the partial differential equations of finance and present some mathematical results for their existence and regularity.
- Then, in Part 4, the main algorithms based on PDE formulations are given. As in Achdou and Pironneau (2005), more emphasis is placed on variational methods with mesh adaptivity in mind. Some acceleration techniques, such as proper orthogonal decomposition and sparse grids, are presented.
- Calibration is dealt with briefly in Part 5, with a more consistent section on Dupire's equations.
- Part 6 is a review of some numerical tools and toolboxes for financial derivatives.

PART ONE
Modelling financial assets and derivatives

Pricing with mathematical and computational tools is practised by institutions or individuals who observe a financial asset, *i.e.*, a share, a bond, a commodity, *etc.*, and wish to draw some conclusion about their future value and the risk involved in dealing with them.

1.1. Financial assets

No arbitrage assumes that the asset's market price is public, fluid, and that it is not possible to make a profit in a loop of buys and sells without taking the risk of losing something. The current pricing theory assumes also that

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

there is a riskless asset, a *numéraire* (money in one currency for example), which can be borrowed or lent at a fixed *interest rate* r, possibly a function of time t. Consequently any riskless asset of price S_t at time t obeys the following:

$$\mathrm{d}S_t = S_t r(t)\,\mathrm{d}t, \quad S_0 \text{ known, } i.e., \; S_t = S_0 \mathrm{e}^{\int_0^t r(s)\,\mathrm{d}s}. \tag{1.1.1}$$

By convention $t = 0$ is today, and the price of the asset today, being public, is known.

Obviously most financial assets are not riskless, and for lack of information (or understanding) their variations are assumed to be random. The simplest model with randomness is

$$\mathrm{d}S_t = S_t(\mu_t\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t), \quad S_0 \text{ known}, \tag{1.1.2}$$

where $\mu_t = \mathbb{E}[\,\mathrm{d}S_t/S_t|S_0]$ is the tendency (or *drift*) of S_t, and σ_t, the *volatility* at time t, measures the amplitude of the random variations, and W_t is assumed to be a Brownian motion for simplicity. This last assumption will be relaxed later; $\mathbb{E}[\cdot|\cdot]$ is the conditional expected value in the same probability space where W_t is defined.

Remark 1.1.1. Recall that a Brownian motion W_t (also called a Wiener process) has zero mean and variance such that $\mathbb{E}[(W_t - W_s)^2] = t - s,\ t > s$, and $W_t - W_s$ is the Gaussian random variable $W_t - W_s = \mathbf{N}(0,1)\sqrt{t-s}$: its probability density function (PDF) is $\theta \to \exp\bigl(-\frac{\theta^2}{2(t-s)}\bigr)/\sqrt{2\pi(t-s)}$ and so its cumulative distribution function (CDF) is

$$\begin{aligned}\mathcal{P}(W_t - W_s < x) &= \frac{1}{\sqrt{2\pi(t-s)}}\int_{-\infty}^{x} \exp\left(-\frac{\theta^2}{2(t-s)}\right)\mathrm{d}\theta \\ &= \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2(t-s)}}\right)\right).\end{aligned} \tag{1.1.3}$$

Notice that the model is set in a probability space, about which not much is said. No arbitrage implies in this case the existence of a *risk-neutral probability measure* under which W_t is still a Brownian motion but[1] $\mathbb{E}[\,\mathrm{d}S_t/S_t|S_0] = r(t)$. Numerically, $\mathbb{E}$ is often replaced by a statistical average (Central Limit Theorem, Theorem 2.1.1), so the probability measure does not appear explicitly in the numerical computations. Hence, for practical purposes, the model for S_t is

$$\mathrm{d}S_t = S_t(r(t)\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t), \quad S_0 \text{ known}. \tag{1.1.4}$$

[1] If μ is a random variable in a probability space of measure $\mathcal{P}$ and $\mathcal{P}^*$ is another probability measure in the same space, then $\mathbb{E}^*[\mu] = \mathbb{E}\bigl[\frac{\mathrm{d}\mathcal{P}^*}{\mathrm{d}\mathcal{P}}\mu\bigr]$, where $\frac{\mathrm{d}\mathcal{P}^*}{\mathrm{d}\mathcal{P}}$ is the Radon–Nikodym derivative of $\mathcal{P}^*$ with respect to $\mathcal{P}$. Obviously, if ρ and ρ^* are the PDF of $\mathcal{P}$ and $\mathcal{P}^*$, $\frac{\mathrm{d}\mathcal{P}^*}{\mathrm{d}\mathcal{P}} = \frac{\rho^*}{\rho}$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The precise meaning of an SDE (stochastic differential equation) such as (1.1.4) requires the theory of stochastic integrals: the reader is referred to Lamberton and Lapeyre (1997) for more details. From a practical numerical point of view it suffices to view (1.1.4) as the limit of (2.1.3) below, when $\delta t \to 0$.

With Itô's lemma (3.2.7), it is not hard to show that (1.1.4) has an explicit solution when r and σ are constant.[2]

Proposition 1.1.2. The random process $X_t = \log S_t$ satisfies

$$\mathrm{d}X_t = \left(r - \frac{1}{2}\sigma_t^2\right)\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \tag{1.1.5}$$

and when σ_t and r are constant then

$$S_t = S_0 \mathrm{e}^{(r-\frac{\sigma^2}{2})t+\sigma(W_t-W_0)}. \tag{1.1.6}$$

Proof. By Itô's lemma,

$$\begin{aligned}\mathrm{d}\ln S_t &= (\ln S_t)'\,\mathrm{d}S_t + \frac{1}{2}(\ln S_t)''(\sigma S_t)^2\,\mathrm{d}t \\ &= \frac{\mathrm{d}S_t}{S_t} - \frac{\sigma^2 S_t^2\,\mathrm{d}t}{2S_t^2} = r\,\mathrm{d}t + \sigma\,\mathrm{d}W_t - \frac{1}{2}\sigma^2\,\mathrm{d}t.\end{aligned}$$

Hence $X_t = \ln S_t$ satisfies (1.1.5), and so

$$X_t - X_0 = \left(r - \frac{1}{2}\sigma^2\right)t + \sigma(W_t - W_0). \qquad \square$$

Remark 1.1.3. This shows that $Y_t = S_t\mathrm{e}^{-rt}$ is a martingale, *i.e.*,

$$\mathbb{E}(Y_t|Y_\tau) = Y_\tau \quad \forall \tau < t.$$

The concept of martingales (see Lamberton and Lapeyre (1997) for details) is important for arbitrage-free markets.

1.2. Models with local volatilities

In practice the Black–Scholes model with constant volatility is too simple. So it is common practice to *calibrate* the model by finding a so-called *local volatility surface* $\sigma(S,t)$ for (1.1.4) in order to reproduce the observations.[3]

One such local volatility model, known as CEV (constant elasticity of variance) has been proposed by Cox (1975):

$$\mathrm{d}S_t = S_t\big(r\,\mathrm{d}t + \delta S_t^{\frac{\beta}{2}-1}\,\mathrm{d}W_{1t}\big), \quad S_0 \text{ known}, \quad 0 < \beta < 2. \tag{1.2.1}$$

[2] If r is not constant, the same formula holds with $\int_0^t r(\tau)\,\mathrm{d}\tau$ instead of rt.

[3] Hence it is important to remember that a mathematical result or a numerical method for (1.1.4) confined to constant volatilities is of limited interest.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

It is of numerical interest because it will lead to a semi-analytical solution, as the probability density function of S_T, knowing that $S_t = S$, is (see Schroder (1989))

$$\rho(S_T|S_t) = (2-\beta)k^{\frac{1}{1-2\beta}}x^{\frac{1}{4-2\beta}}w^{\frac{1-2\beta}{4-2\beta}}\mathrm{e}^{-x-w}I_{\frac{1}{2-\beta}}(2\sqrt{xw}), \tag{1.2.2}$$

with $\tau = T - t$, I_a the modified Bessel function of the first kind of order a, and

$$k = \frac{2r}{\delta^2(2-\beta)(\mathrm{e}^{r(2-\beta)\tau}-1)}, \quad x = kS^{2-\beta}\mathrm{e}^{r(2-\beta)\tau}, \quad w = kS_T^{2-\beta}. \tag{1.2.3}$$

1.3. Stochastic volatility models

By assuming a stochastic volatility defined by a scalar or vector SDE, one obtains a large class of models with fewer parameters than with a general local volatility surface.

As before,

$$\mathrm{d}S_t = S_t(\mu_t\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_{1t}), \tag{1.3.1}$$

where μ_t is the tendency of S_t, taken to be r_t, the interest rate, if a risk-neutral probability exists, and W_{1t} is a Brownian motion. Now we assume that σ_t is a stochastic process satisfying an SDE driven by a second Brownian motion W_{2t}, correlated to W_{1t}: $\mathbb{E}[\,\mathrm{d}W_{1t}\,\mathrm{d}W_{2t}] = \rho\,\mathrm{d}t$,

$$\mathrm{d}v_t = \kappa(m - v_t)\,\mathrm{d}t + \beta\,\mathrm{d}W_{2t}, \quad \sigma_t = f(v_t), \tag{1.3.2}$$

where κ, β, m are given, and are possibly functions of v_t and t, f is a given positive function and (v_t) is the driving process. Popular choices include the following.

- The log-normal process: $\kappa < 0, \beta$ constant, $m = 0$.
- The mean reverting Ornstein–Uhlenbeck (OU) process: $\kappa > 0, \beta, m > 0$ constant.
- The Cox–Ingersoll–Ross process (CIR): $\beta = \lambda\sqrt{v_t}$, $\kappa > 0, m > 0, \lambda$ constant.

The volatility β of v_t is commonly referred to as *volvol* (volatility of the volatility). In the case of OU and CIR , m is the limit of v_t when $t \to \infty$ (the process is said to be mean reverting), κ controls the rate at which the limit is reached, and β the randomness about this limit.

When the Brownian motion W_{2t} is correlated to W_{1t}, it can be written as a linear combination of W_{1t} and an independent Brownian motion Z_t:

$$\hat{Z}_t = \rho W_{1t} + \sqrt{1-\rho^2}Z_t, \tag{1.3.3}$$

where the *correlation factor* ρ lies in $[-1, 1]$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Table 1.3.1. Frequently used models of stochastic volatilities.

Authors	ρ	$f(y)$	v_t process
Hull and White (1987)	$\rho = 0$	$f(y) = \sqrt{y}$	log-normal
Stein and Stein (1991)	$\rho = 0$	$f(y) = \lvert y\rvert$	mean reverting OU
Heston (1993)	$\rho \neq 0$	$f(y) = \sqrt{y}$	CIR

Table 1.3.1 summarizes some popular choices for $f(v)$ and ρ.

Following Bervoets (2006), with $\tau = t - t_0$, the characteristic function of $X_t = \ln S_t$, knowing S_{t_0} and v_{t_0}, when S_t obeys Heston's model, is

$$\Phi(z) = \exp\left(\mathrm{i}zr\tau + \frac{\rho}{\lambda}(v_t - v_{t_0} - \kappa m\tau)\right)\phi\left(z\left(\frac{\kappa\rho}{\lambda} - \frac{1}{2}\right) + \frac{1}{2}\mathrm{i}z^2(1-\rho^2)\right), \tag{1.3.4}$$

where $\Phi(z) := \mathbb{E}[\mathrm{e}^{\mathrm{i}zX_t}]$ and

$$\phi(z) = \gamma(z)\frac{\mathrm{e}^{-\frac{1}{2}(\gamma(z)-\kappa)\tau}(1-\mathrm{e}^{-\kappa\tau})}{\kappa(1-\mathrm{e}^{-\gamma(z)\tau})} \times \mathrm{e}^{\frac{v_{t_0}+v_t}{\kappa^2}\left[\frac{\kappa(1+\mathrm{e}^{-\kappa\tau})}{1-\mathrm{e}^{-\kappa\tau}}\right]}\frac{I_{\frac{d}{2}-1}\left(\sqrt{v_{t_0}v_T}\frac{4\gamma(z)\mathrm{e}^{-\frac{1}{2}\gamma(z)\tau}}{\lambda^2(1-\mathrm{e}^{-\kappa\tau})}\right)}{I_{\frac{d}{2}-1}\left(\sqrt{v_{t_0}v_T}\frac{4\kappa\mathrm{e}^{-\frac{1}{2}\kappa\tau}}{\lambda^2(1-\mathrm{e}^{-\kappa\tau})}\right)}, \tag{1.3.5}$$

with $\gamma(z) = \sqrt{\kappa^2 - 2\lambda^2\mathrm{i}z}$, $d = 4m/\lambda^2$ and $I_a(x)$ the modified Bessel function of the first kind.

1.4. Models with jump-diffusion processes

To account for sudden discontinuous variations of S_t, one adds random jumps to W_t. Jump processes were introduced by P. Lévy (see Bertoin (1996)). For details the reader is referred to Cont and Tankov (2003). Below we present, without mathematical rigour, the minimum needed to generate a Lévy process numerically.

1.4.1. Poisson process

A positive random variable τ with exponential law (ELVR) is defined by the probability law

$$\mathcal{P}(\tau > y) = \mathrm{e}^{-\lambda y}. \tag{1.4.1}$$

Let $\{\tau_i\}$ be a set of independent ELVRs with the same λ. Let

$$T_n = \sum_1^n \tau_i, \quad N_t = \sum_{n\geq 1} \mathbf{1}_{t\geq T_n}. \tag{1.4.2}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

By definition, N_t is a *Poisson process*.

It is very useful in queuing theory: for instance, it can be used to model the number of buses which arrive at one bus station.

Proposition 1.4.1. $N_t - N_s$ and N_{t-s} have the same law:

$$P(N_t = n) = \mathrm{e}^{-\lambda t}\frac{(\lambda t)^n}{n!}.$$

Furthermore, if it is known that $N_T = n$, then $T_1, \ldots, T_n$ are uniformly distributed on $(0, T)$.

Definition 1.4.2. A *compound Poisson* process on f is $Z_t = \sum_1^{N_t} v_i$, where $\{v_i\}$ are independent random variables with law f.

Consequently a numerical method to generate a compound Poisson process Z_t is as follows.

(1) Generate N_T by (1.4.1), (1.4.2).
(2) Generate $\{U_i\}_1^{N_T}$ uniformly distributed on $(0, T)$.
(3) Generate N_T random variables $\{v_i\}_1^{N_T}$ with law f.
(4) Set $Z_t = \sum_1^{N_T} v_i \mathbf{1}_{U_i \le t}$.

1.4.2. Jump-diffusion process

Let W_t be a Brownian process and let Z_t be a compound Poisson process. Let $\sigma, \gamma \in \mathbb{R}$. Then

$$X_t = \gamma t + \sigma W_t + Z_t \tag{1.4.3}$$

is, by definition, a *jump-diffusion process*.

Theorem 1.4.3. Let $\nu(x) = \lambda f(x)$. The *characteristic function* of X_t is

$$\mathbb{E}(\mathrm{e}^{\mathrm{i}uX_t}) = \exp\left(t\left[\mathrm{i}\gamma u - \frac{\sigma^2 u^2}{2} + \int_{\mathbb{R}} (\mathrm{e}^{\mathrm{i}ux} - 1 - \mathrm{i}u\mathbf{1}_{|x|<1})\nu(\mathrm{d}x)\right]\right). \tag{1.4.4}$$

Conversely, if $\nu \in L^1(R)$ and $\int_{-1}^1 x^2 \nu(\mathrm{d}x) < \infty$, (1.4.4) defines the characteristic function of a *generalized* jump-diffusion process. Formula (1.4.4) is referred to as the *Lévy–Khinchin formula*.

Theorem 1.4.4. The infinitesimal generator of the semi-group associated to X_t is

$$\begin{aligned} L^X\phi &:= \lim_{t\to 0}\frac{1}{t}\mathbb{E}(\phi(x + X_t) - \phi(x)) \qquad (1.4.5)\\ &= \frac{\sigma^2}{2}\partial_{xx}\phi + \gamma\partial_x\phi + \int_{\mathbb{R}} \big(\phi(x+y) - \phi(x) - y\mathbf{1}_{|y|<1}\partial_x\phi(x)\big)\nu(\mathrm{d}y). \end{aligned}$$

The following property will be useful later.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Proposition 1.4.5. The process e^{X_t} is a martingale if and only if

$$\gamma + \frac{\sigma^2}{2} + \int_{\mathbb{R}} (e^x - 1 - x\mathbf{1}_{|x|<1})\nu(dx) = 0, \tag{1.4.6}$$

and with this value for γ the infinitesimal generator of X_t is

$$L^X\phi = \frac{\sigma^2}{2}(\partial_{xx}\phi - \partial_x\phi) + \int_{\mathbb{R}} \big(\phi(x+y) - \phi(x) - (e^y - 1)\partial_x\phi(x)\big)\nu(dy). \tag{1.4.7}$$

1.5. Financial derivatives

1.5.1. European options

Given a financial asset S_t, an option on S is a contract for buying or selling S at a future time T. Such a product is also called a *financial derivative* or a *future* on S.

The European contract is the simplest: it gives its owner the right to buy (call) or sell (put) S at time T (maturity) at price K (the strike).

Such contracts have a market value: they too can be bought or sold. No arbitrage implies that the fair value C_t (resp. P_t) of a European call (resp. put) at time $t < T$ is the expected value of the owner's profit at time T discounted by the interest rate r down to time t:

$$C_t = e^{-r(T-t)}\mathbb{E}[(S_T - K)^+|S_t], \quad P_t = e^{-r(T-t)}\mathbb{E}[(K - S_T)^+|S_t]. \tag{1.5.1}$$

Recall the notation $a^+ = \max(a, 0)$.

Note that $C_t - P_t$ pays $(S_T - K)^+ - (K - S_T)^+$ at maturity, which is equal to $(S_T - K)$. Because it is riskless, no arbitrage implies that they must be equal at time t discounted by r, namely

$$C_t - P_t = e^{-r(T-t)}(S_T - K). \tag{1.5.2}$$

This relation is known as the *put–call parity*.

Calling $P(S, t)$ the price of an option with maturity T and pay-off function $P_0(S)$, and assuming that r and $\sigma > 0$ are constant, the Black–Scholes formula is

$$P(S,t) = e^{-r(T-t)}\mathbb{E}^*\big(P_0\big(Se^{r(T-t)}e^{\sigma(W_T - W_t) - \frac{\sigma^2}{2}(T-t)}\big)\big), \tag{1.5.3}$$

and since, under P^*, $W_T - W_t$ is a centred Gaussian distribution with variance $T - t$,

$$P(S,t) = \frac{1}{\sqrt{2\pi}}e^{-r(T-t)}\int_{\mathbb{R}} P_0\big(Se^{(r - \frac{\sigma^2}{2})(T-t) + \sigma x\sqrt{T-t}}\big)e^{-\frac{x^2}{2}}\,dx. \tag{1.5.4}$$

When the option is a European option, a more explicit formula can be

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

deduced from (1.5.4). Take, for example, a call:

$$C(S,t) = \frac{1}{\sqrt{2\pi}} \int_{-d_2}^{+\infty} \left(S\mathrm{e}^{-\frac{\sigma^2}{2}(T-t)+\sigma x\sqrt{T-t}} - K\mathrm{e}^{-r(T-t)}\right)\mathrm{e}^{-\frac{x^2}{2}}\,\mathrm{d}x$$
$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d_2} \left(S\mathrm{e}^{-\frac{\sigma^2}{2}(T-t)-\sigma x\sqrt{T-t}} - K\mathrm{e}^{-r(T-t)}\right)\mathrm{e}^{-\frac{x^2}{2}}\,\mathrm{d}x, \quad (1.5.5)$$

where

$$d_1 = \frac{\log\left(\frac{S}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)}{\sigma\sqrt{T-t}} \quad \text{and} \quad d_2 = d_1 - \sigma\sqrt{T-t}. \quad (1.5.6)$$

Finally, introducing the cumulative distribution function for $\mathbf{N}(0,1)$,

$$N(d) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{d} \mathrm{e}^{-\frac{x^2}{2}}\,\mathrm{d}x, \quad (1.5.7)$$

and using (1.5.5), (1.5.6), we obtain the *Black–Scholes formula*.

Proposition 1.5.1. When σ and r are constant, the price of the call is given by

$$C(S,t) = SN(d_1) - K\mathrm{e}^{-r(T-t)}N(d_2), \quad (1.5.8)$$

and the price of the put is given by

$$P(S,t) = -SN(-d_1) + K\mathrm{e}^{-r(T-t)}N(-d_2), \quad (1.5.9)$$

where d_1 and d_2 are given by (1.5.6) and N is given by (1.5.7).

Proposition 1.5.2. With CEV volatilities $\sigma_t = \delta S_t^{\frac{\beta}{2}-1}$,

$$C_t = S_t Q\left(2kK^{2-\beta}; 2 + \frac{2}{2-\beta}, 2x\right)$$
$$- K\mathrm{e}^{-r(T-t)}\left(1 - Q\left(2x; \frac{2}{2-\beta}, 2kK^{2-\beta}\right)\right), \quad (1.5.10)$$

where Q is the non-central Chi-square distribution (see Schroder (1989)) with x, k defined by (1.2.3). When $\beta = 4/3$, the Q needed are

$$Q(z;1,\kappa) = N(\sqrt{\kappa} - \sqrt{z}) + N(-\sqrt{\kappa} - \sqrt{z})$$
$$Q(z;3,\kappa) = Q(z;1,\kappa) + \frac{1}{\sqrt{\kappa}}[N'(\sqrt{\kappa} - \sqrt{z}) - N'(\sqrt{\kappa} + \sqrt{z})]$$
$$Q(z;5,\kappa) = Q(z;1,\kappa) + \frac{1}{\sqrt{\kappa^3}}[(\kappa - 1 + \sqrt{\kappa z})N'(\sqrt{\kappa} - \sqrt{z})$$
$$- (\kappa - 1 - \sqrt{\kappa z})N'(\sqrt{\kappa} + \sqrt{z})], \quad (1.5.11)$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

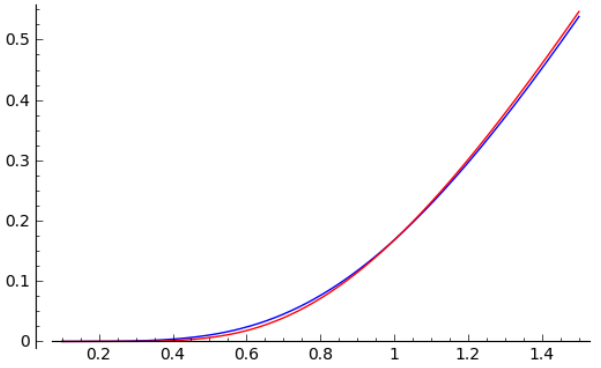


Figure 1.5.1. European call with CEV volatility, $\beta = \frac{4}{3}$, $\delta = 0.3$, $T = 2$, $r = 0$, $K = 1$. The integrated software Sage was used (www.sage.org) on (1.5.10). A comparison with the Black–Scholes solution is given with $\sigma = 0.3$.

where $N'(x) = \mathrm{e}^{\frac{x^2}{2}}/\sqrt{2\pi}$ is the standard normal density function and

$$N(x) = \frac{1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)}{2}$$

is the normal probability distribution.

Proof. See Schroder (1989). □

A numerical simulation shows $S_t \to C_t$ in Figure 1.5.1 when $t = 0$, $T = 2$, $\delta = 0.3$, $\beta = 4/3$, $K = 1$ and $r = 0$.

```
// Sage script
s=0.6;S=var('S'); z =var('z');
d=0.3; tau=2; y= 9/d^2/tau; x=y*S^(2/3);
p=sqrt(y)+sqrt(x); m = sqrt(x)-sqrt(y);
N(z)=(1+erf(z/sqrt(2)))/2;
Np(z) =  exp(-z^2/2)/sqrt(2*pi);
Q1= N(m)+N(-p)+x^(-3/2)*( (x-1+sqrt(x*y))*Np(m)
                         -(x-1-sqrt(x*y))*Np(p));
Q2= N(-m)+N(-p)+y^(-1/2)*(Np(-m)-Np(p));
C=S*Q1-1+Q2;
v=(S*(1-erf(-ln(S)/s - s/4))-1-erf(ln(S)/s - s/4))/2;
for i in range (1,15) : print  numerical_approx(C(0.1*i));
A=plot(C,(0.1,1.5),color='blue'); B=plot(v,(0.1,1.5),color='red'); A+B;
```

Proposition 1.5.3. If we use Heston's stochastic volatility model (1.3.1), (1.3.2), (1.3.3), knowing that $S = S_0, v = v_0$ at $t = 0$, then the European

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

call is

$$C = \frac{1}{2}(S_0 - K\mathrm{e}^{-rT}) + \frac{1}{\pi}\int_0^\infty (\mathrm{e}^{rT} f_1 - K f_2)\,\mathrm{d}u,$$

$$f_j = \mathrm{Re}\frac{\phi(u + (j-2)\mathrm{i})}{\mathrm{i}uK^{\mathrm{i}u}},$$

$$\phi(u) = \mathrm{e}^{rT} S_0^{\mathrm{i}u}\left(\frac{1 - g\mathrm{e}^{-dT}}{1-g}\right)^{-2\frac{m\kappa}{\lambda^2}} \exp\left(\frac{m\kappa T}{\lambda^2}(\kappa - \rho\lambda\mathrm{i}u - d)\right)$$
$$\times \exp\left(\frac{v_0}{\lambda^2}(\kappa - \rho\lambda\mathrm{i}u + d)\frac{1 - \mathrm{e}^{dT}}{1 - g\mathrm{e}^{dT}}\right),$$

$$\text{where}\quad d = \sqrt{(\rho\lambda u\mathrm{i} - \kappa)^2 + \lambda^2(\mathrm{i}u + u^2)}, \quad g = \frac{\kappa - \rho\lambda\mathrm{i}u - d}{\kappa - \rho\lambda\mathrm{i}u + d}. \qquad (1.5.12)$$

The proof is due to Cox, Ingersoll and Ross (1985), and revised by Albrecher, Mayer, Schoutens and Tistaert (2007). A summary can be found in Mikhailov and Nögel (2003), and hints on how to compute the complex integral in Albrecher *et al.* (2007).

```
# Sage script for Heston model by a complex integral when K=1
umax=1000; N=10000; du=umax/N;
S0=1; T=2;Y0=0.0175; r=0.025;
kappa=1.5768;m=0.0398; lambd=0.5751;rho=-0.5711;
aa= m*kappa*T/lambd^2; bb= -2*m*kappa/lambd^2;
P=0;
for i in range (1,N) :
     u2=i*du
     u1=u2-I
     a1=rho*lambd*u1*I
     a2=rho*lambd*u2*I
     d1=sqrt((a1-kappa)^2+lambd^2*(u1*I+u1^2))
     d2=sqrt((a2-kappa)^2+lambd^2*(u2*I+u2^2))
     g1=(kappa-a1-d1)/(kappa-a1+d1)
     g2=(kappa-a2-d2)/(kappa-a2+d2)
     b1=exp(u1*I*(log(S0)+r*T)) *( (1-g1*exp(-d1*T))/(1-g1) )^bb
     b2=exp(u2*I*(log(S0)+r*T)) *( (1-g2*exp(-d2*T))/(1-g2) )^bb
     phi1=b1*exp(aa*(kappa-a1-d1)+Y0*(kappa-a1-d1)*(1-exp(-d1*T))/
       (1-g1*exp(-d1*T))/lambd^2)
     phi2=b2*exp(aa*(kappa-a2-d2)+Y0*(kappa-a2-d2)*(1-exp(-d2*T))/
       (1-g2*exp(-d2*T))/lambd^2)
     P+= real((phi1-phi2)/(u2*I))*du
print numerical_approx((S0-exp(-r*T))/2+P/pi)
```

1.5.2. Barrier option

Barriers are introduced in case the market goes wild. The contracts include a clause that cancels the deal if S_t crosses over a specified price S_M or under S_m.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

1.5.3. American and Bermuda options

With American options the owner of the contract can exercise his right and force the deal any time before maturity T. To decide to exercise the option at time τ one must compare the profit gain now, namely $S_\tau - K$ for a call, with the expected profit if one waits, namely $\mathrm{e}^{-r(T-\tau)}\mathbb{E}(S_T - K)^+$. Thus the value of the American put option is

$$P_t = \sup_{\tau \in \mathcal{T}(t,T)} \mathbb{E}[\mathrm{e}^{-r(\tau-t)}(K - S_\tau)^+ | S_t], \tag{1.5.13}$$

where $\mathcal{T}$ is the set of stopping times in (t, T) adapted to the probability space.

Bermuda option

The mechanism of Bermudas is similar to Americans except that the decisions to exercise are made only at fixed, given time intervals – for instance every month. The Bermuda islands are between America and Europe, hence the name.

1.5.4. Path-dependent options

A path-dependent or Asian (or Asiatic) option tries to smooth the oscillations of the underlying asset at maturity. Some average of S_t over a time period is used in place of S_T in the pay-off $(A_T - K)^+$ for a call, with arithmetic averaging, for instance

$$A_t = \frac{1}{t}\int_0^t S_\tau \, \mathrm{d}\tau,$$

or geometric averaging,

$$A_t = \exp\left(\frac{1}{t}\int_0^t \log(S_\tau)\, \mathrm{d}\tau\right).$$

Another example is the call with floating strike with pay-off $(S_T - A_T)^+$. No arbitrage gives

$$C_t = \mathrm{e}^{-r(T-t)}\mathbb{E}[(S_T - A_T)^+ | S_t, A_t]. \tag{1.5.14}$$

Another class of Asian options, the so-called *lookback options*, involve the extremal values of the asset price for $t \le T$. The floating strike lookback call has a pay-off of $(S_T - \min_{0\le\tau\le T} S_\tau)^+$, whereas the lookback put has a pay-off of $(\max_{0\le\tau\le T} S_\tau - S_T)^+$. Similarly, the fixed strike lookback call (resp. put) has a pay-off of $(\max_{0\le\tau\le T} S_\tau - K)^+$ (resp. $(K - \min_{0\le\tau\le T} S_\tau)_+$). One can also define lookback options on averages.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

1.5.5. Basket options

Consider d risky assets S_{it}, $i = 1, \ldots, d$, each modelled by a separate SDE:

$$\mathrm{d}S_{it} = S_{it}(r\,\mathrm{d}t + \sigma_i\,\mathrm{d}W_{it}). \tag{1.5.15}$$

However, the standard Brownian motions (W_{it}), $1 \le i \le d$, are possibly correlated: $\rho_{i,j} := \frac{1}{t}\mathbb{E}[W_{it}W_{jt}]$. As usual r is the interest rate. In what follows, the notation $\mathbf{S}$ is used for the vector $(S_1, \ldots, S_d)^T$.

Remark 1.5.4. As in the one-dimensional case, if σ_i is constant for all i:

$$S_{it} = S_{i0}\exp\left(\left(r - \frac{1}{2}\sigma_i^2\right)t + \sigma_i(W_{it} - W_{0t})\right). \tag{1.5.16}$$

A European option with pay-off $\pi(\mathbf{S}_T)$ at maturity T will be priced by

$$P_t = \mathrm{e}^{-r(T-t)}\mathbb{E}[\pi(\mathbf{S}_T)|\mathbf{S}_t]. \tag{1.5.17}$$

The pay-off function could be a function of a weighted sum of the assets, *i.e.*, in the case of a put:

$$\pi(\mathbf{S}) = \left(K - \sum_{i=1}^{d}\alpha_i S_i\right)^+.$$

As in the scalar case, the put–call parity holds:

$$C_t - P_t = \sum_{i=1}^{d}\alpha_i S_{it} - K\mathrm{e}^{-r(T-t)}.$$

The pay-off could also be a function of $\max_{i=1,\ldots,d} S_i$: these options are called *best-of* options. The best-of put option yields

$$\pi(\mathbf{S}) = \left(K - \max_{i=1,\ldots,d} S_i\right)^+.$$

In contrast with the previous case, there is no put–call parity for best-of options.

For a detailed mathematical treatment of the problem in the presence of jumps, see Farkas, Reich and Schwab (2006).

Remark 1.5.5. Another presentation can be made with independent Brownian motions $\tilde{W}_{it}$ and

$$\mathrm{d}S_{it} = S_{it}\left(r\,\mathrm{d}t + \sum_{j=1}^{i}\sigma_{ij}\,\mathrm{d}\tilde{W}_{jt}\right), \tag{1.5.18}$$

with $\sum_{k\le i,j}\sigma_{ik}\sigma_{jk} = \sigma_i\sigma_j\rho_{ij}$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Then, when r, σ_{ij} are constant,

$$S_{iT}(\mathbf{X}) = S_{i0} \exp\left(\left(r - \frac{1}{2}\sum_j \sigma_{ij}^2\right)T + \sum_j \sigma_{ij} X_j \sqrt{T}\right), \tag{1.5.19}$$

where $\mathbf{X} = \{X_j\}_1^d$ are d normal random variables. The computation of $\pi(\mathbf{S}_0) := \mathrm{e}^{-rT}\mathbb{E}[(K - H(\mathbf{S}_T))^+ | \mathbf{S}_0]$ reduces to the multiple integral

$$\pi(\mathbf{S}) = \mathrm{e}^{-rT} \int_{\mathbb{R}^d} (K - H(\mathbf{S}_T(\mathbf{X})))^+ \prod_j \frac{\mathrm{e}^{\frac{-X_j^2}{2}}}{\sqrt{2\pi}}\, \mathrm{d}X_1, \ldots, \mathrm{d}X_d \tag{1.5.20}$$

$$= \frac{\mathrm{e}^{-rT}}{(2\pi)^{\frac{d}{2}}} \int_{\mathbb{R}^d} (K - H(S_{i0}\, \mathrm{e}^{(r - \frac{1}{2}\sum_j \sigma_{ij}^2)T + \sum_j \sigma_{ij} X_j \sqrt{T}}))^+ \, \mathrm{e}^{-\frac{|\mathbf{X}|^2}{2}}\, \mathrm{d}\mathbf{X}.$$

1.5.6. Convertible bonds

Historically, bonds issued by a company, whose share is denoted here by S_t, were refunded in cash. Convertible bonds can be refunded in shares of the company. When this happens the debts of the company are reduced. Conversely, the bond's owner can participate in the growth of the company.

The market for convertible bonds is also very large (>\$500 billion in 2002). The challenge here is to price such a convertible bond and possibly tell when it is best to convert bonds into shares. The difficulty comes from the spread between the put price B_p and the call price B_c.

Following Ayache, Forsyth and Vetzal (2003) we assume that S_t is driven by a standard SDE such as (1.1.1); the bond is exchangeable for κS_t.

Like any contingent claim on S_t, no arbitrage implies that the value of the convertible bond is

$$V_0 = \mathrm{e}^{-r(T-t)}\mathbb{E}[\max\{B_T, \kappa S_T\} | S_0],$$

where B_T is the agreed refund value of the bond at maturity. However, there are constraints, due to the convertibility of V_t into κS_t or B_c and of B_p into V_t: if $B_c < \kappa S_t$ then $V_t = \kappa S_t$, otherwise

$$\max\{B_p, \kappa S_t\} \le V_t \le B_c. \tag{1.5.21}$$

So the situation is similar to American options (though more complex), and even more so if the bond is itself a risky asset.

1.5.7. Interest rates, swaps and swaptions

The fluctuation of interest rate r cannot be ignored for assets with long maturity, such as zero coupon bonds sometimes used in retirement plans. The receiver of a given sum today, B_0, signs an obligation to pay a certain amount B_T at time T, and no yearly interest (zero coupon) is paid in between. A fair price for the bond would be $B_0 = \mathbb{E}[\exp(\int_0^T r_s\, \mathrm{d}s) B_T | r_0]$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The simplest model to describe the fluctuations of the interest rate is the one-factor model of Vasicek:

$$\mathrm{d}r_t = \lambda(m - r_t)\,\mathrm{d}t + \eta\,\mathrm{d}W_{1t}, \quad r_0 \text{ given.} \tag{1.5.22}$$

Similar problems arise in conjunction with derivatives on S_t when T is large. SABR (stochastic alpha, beta, rho: see Brigo and Mercurio (2007)) can be used for S_t,

$$\frac{\mathrm{d}S_t}{S_t} = r_t\,\mathrm{d}t + \sigma_t S_t^{\beta-1}\,\mathrm{d}W_{2t}, \quad S_0 \text{ given,} \tag{1.5.23}$$

combined with Vasicek and Hull–White as in Chen, Grzelak and Oosterlee (2011),

$$\sigma_t = g(t)\kappa_t, \quad \frac{\mathrm{d}\kappa_t}{\kappa_t} = h(t)\,\mathrm{d}W_{3t}, \quad \kappa_0 \text{ given,} \tag{1.5.24}$$

with given functions $g(t), h(t)$, constant parameters $0 < \beta < 1$, and $\lambda, \eta, m > 0$, and given correlations $\rho_{ij}\,\mathrm{d}t = \mathbb{E}[\,\mathrm{d}W_{it}\,\mathrm{d}W_{jt}]$, $i, j = 1, 2, 3$.

Another way is to use a Heston–Hull–White model in conjunction with (1.5.22):

$$\begin{aligned} \frac{\mathrm{d}S_t}{S_t} &= r_t\,\mathrm{d}t + \sqrt{\mu_t}\,\mathrm{d}W_{2t}, \quad S_0 \text{ given,} \\ \mathrm{d}\mu_t &= \kappa(\overline{\mu} - \mu_t)\,\mathrm{d}t + \gamma\sqrt{\mu_t}\,\mathrm{d}W_{1t}, \quad \mu_0 \text{ given.} \end{aligned} \tag{1.5.25}$$

Interest rate swap

An interest rate swap is an agreement between the *payer* A and the *receiver* B to exchange interest rate payments on a notational (imaginary) amount, under some conditions on time and amount.

Typically A must pay the LIBOR floating interest rate (London interbank offered rate) while B must pay a fixed interest rate on the same notational amount.

If A wants the security of B's rate, he has to pay B a fixed coupon (called the *fixed leg*), which is the interest rate that B has to pay plus a premium, say semi-annually.

In exchange B will give to A the LIBOR rate plus possibly a fixed coupon, the two making the *floating leg* paid at some time interval δ (quarterly, for instance). Naturally the value of a swap is the difference between the two legs; at time 0 it is usually 0.

The modelling begins with an assumption on the short interest rate r_t, for instance the SDE (1.5.22). Next the dematerialized sum on which the swap is based could be modelled as a zero coupon bond B_t driven by

$$\frac{\mathrm{d}B_t}{B_t} = r_t\,\mathrm{d}t + \sigma_t^B\,\mathrm{d}W_t^B, \quad B_0 \text{ given.} \tag{1.5.26}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

If L_t is the value of the LIBOR, then $B_{t+\delta} = B_t/(1+\delta L_t)$. The LIBOR is also modelled by an SDE,

$$\frac{\mathrm{d}L_t}{L_t} = \mu_t\,\mathrm{d}t + \sigma_t^L\,\mathrm{d}W_t^L, \quad L_0 \text{ given.} \tag{1.5.27}$$

Model coherence requires us to relate σ_t^B to L_t and σ_t^L, thus making σ^B stochastic even if σ^L is a fixed number.

If the times of swaps are $\{T_i\}_1^I$, and $B_t(T)$ denotes the value at t of the zero coupon fixed at T, then the value of the swap is

$$V_t(T,T_I) = \frac{B_t(T) - B_t(T_I)}{\sum_1^I \delta B_t(T_i)}.$$

Interest rate derivatives, swaptions

An interest rate derivative is a future where the underlying asset is the right to pay or receive a notional amount at a given interest rate. The interest rate derivative market is huge, more than \$500 trillion in 2010. Interest rate derivatives are used to hedge interest rate risk. Swaptions are one such tool. A swaption is a future on the swap: it is priced as

$$SW = B_t(T)\mathbb{E}\left[\sum_1^I \mathrm{e}^{-\int_T^{T_i} r_s\,\mathrm{d}s}\delta(V_T(T,T_I)-K)^+\right],$$

where T is the maturity of the contract and K the strike. More details can be found in El Karoui and E. Gobet (2011) and Xuan Luo (2007).

1.5.8. CDSs: credit default swaps

A CDS is a type of insurance contract against the event that a firm or a sovereign defaults on its debt. The seller of the swap will refund some of the loss to the buyer in case of default, in exchange for annuities (spread) up to the maturity of the contract.

In 2008, CDSs (there were \$58 trillion of these, twice the US stock market) and CDOs triggered the domino effect in the subprime crisis that led to the bankruptcy of AIG and Lehman Brothers.

In an interesting article, Ang and Longstaff (2011) model the contract assuming two causes trigger default: one due to a jump in the model for the underlying asset, a bond, and characterized by the intensity of the jump, ξ_t, and the other due to a systemic jump, also characterized by the intensity of the jump λ_t. Both are modelled by a mean reverting process:

$$\begin{aligned}\mathrm{d}\xi_t &= a(\xi_m - \xi_t)\,\mathrm{d}t + c\sqrt{\xi_t}\,\mathrm{d}W_{1t},\\ \mathrm{d}\lambda_t &= \alpha(\lambda_m - \lambda)\,\mathrm{d}t + \sigma\sqrt{\lambda}\,\mathrm{d}W_{2t}.\end{aligned} \tag{1.5.28}$$

For the second cause the probability of default given a systemic jump, γ, is specific to the bond.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Ang and Longstaff (2011) show that the instantaneous probability of default is $\gamma\lambda + \xi$. Assuming that, in case of default, the bond holder recovers $1 - w$ of the par value of the bond, the protection leg of the CDS and the spread leg are

$$w\mathbb{E}\left[\int_0^T \mathrm{e}^{-rt}(\gamma\lambda + \xi)\mathrm{e}^{-\int_0^t p_\tau \,\mathrm{d}\tau}\,\mathrm{d}t\right] \quad \text{and} \quad s\mathbb{E}\left[\int_0^T \mathrm{e}^{-rt}\mathrm{e}^{-\int_0^t p_\tau \,\mathrm{d}\tau}\,\mathrm{d}t\right]. \tag{1.5.29}$$

A formula for s, the spread paid by the buyer of the default protection, is found by writing that both legs are equal (no arbitrage).

1.5.9. CDOs: collateralized debt obligations

Companies who own a hundred or more underlying assets structure them into a single financial product of the same value. Yet the product is usually divided into several tranches, each with its own default risk.

Following Joshi (2011), let us consider a very simple model for CDOs with a single tranche. Each asset has its own SDE,

$$\mathrm{d}S_{it} = S_{it}(r\,\mathrm{d}t + \sigma_i\,\mathrm{d}W_{it}), \quad \text{with } W_{it} = \rho Z_t + \sqrt{1-\rho^2}Z_{it},$$

where Z_t and Z_{it} are independent Wiener processes. We assume also that each asset defaults according to a Poisson process with deterministic intensity λ_i (*cf.* (1.4.1)). Then the cumulative default probability $\mathcal{P}_i$ and hence the correlated defaults τ_i are

$$\mathcal{P}_i(\tau) = 1 - \mathrm{e}^{-\int_0^\tau \lambda_i(s)\,\mathrm{d}s}, \quad \tau_i = \mathcal{P}_i^{-1}(\mathcal{P}(W_i)),$$

where $\mathcal{P}(x) = \big(1 + \mathrm{erf}\big(\frac{x}{\sqrt{2}}\big)\big)/2$ is the normal CDF. If λ_j is independent of time,

$$\tau_i = \frac{1}{\lambda_i}\ln\left(\frac{1}{2}(1 - \mathrm{erf}(W_i))\right).$$

Once the default times are known, the pay-off can be calculated. For example, the contract can say that if 10 assets default before maturity, full refund of these will be given.

The difficulty, however, is that very few paths of a Monte Carlo method will lead to a non-zero pay-off: importance sampling must be used.

1.6. Conclusion

In the beginning the models were simple, but now some contain several correlated processes, no longer Gaussian, leading to integro-differential systems in high dimensions with inequalities and unusual boundary conditions.

Analytical solutions to the models are no longer available: numerical simulations are necessary.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

So far each *future* has been simulated separately but, the threat of systemic risk being real, it is expected that the next generation of models will involve multiple financial assets and derivatives. If so, massive computing power will be necessary.

PART TWO
SDE-based numerical methods

In this section we intend to give a pragmatic account of a few algorithms based on the stochastic formulation of the problems.

2.1. Principles of numerical simulation by Monte Carlo

Notice that it is easy to simulate (1.1.4) or (1.1.6) numerically. One way is to use the C-function `rand()` which, each time it is called, returns a different equiprobable integer between `0` and `RAND_MAX` (a machine-dependent large integer) and the Box–Muller formula to generate a normal Gaussian random variable,

$$\mathbf{N}_{0,1} = \sqrt{2\ln\frac{1}{x}}\cos 2\pi y, \tag{2.1.1}$$

where x and y are generated by `rand()/RAND_MAX`, then set

$$W_t - W_0 = \mathbf{N}_{0,1}\sqrt{t}, \quad \delta W_t = \mathbf{N}_{0,1}\sqrt{\delta t}, \tag{2.1.2}$$

and then use (1.1.6) directly or, if σ is not constant, use a finite difference approximation of (1.1.4):

$$S_{t+\delta t} = S_t(1 + r\delta t + \sigma_t \delta W_t). \tag{2.1.3}$$

Any mean quantity function of S_T such as $C = \mathrm{e}^{-rT}\mathbb{E}[(S_T - K)^+]$ can be computed by using the law of large numbers:

$$C \approx \frac{\mathrm{e}^{-rT}}{N}\sum_{n=1}^{N}\max\{S^n_{J\delta t} - K, 0\}, \tag{2.1.4}$$

where $\{S^n_t\}$ are N independent computations with (2.1.3). There are two sources of numerical error on C: one due to the Euler scheme which is first-order accurate only, and another due to the replacement of the mean with respect to the risk-neutral probability by a sample average. The following addresses the second source of errors.

Theorem 2.1.1. (Central Limit Theorem) Let x be a random variable with probability density p, expectation $\mathbb{E}[x]$ and variance

$$\mathrm{var}(x) = \mathbb{E}[(x - \mathbb{E}([x])^2].$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Let $\{x_i\}_1^N$ be N samples of x. The approximation

$$\mathbb{E}[x] := \int_{-\infty}^{+\infty} p(x)x\,\mathrm{d}x \approx \mathbb{E}_N[x] := \frac{1}{N}\sum_1^N x_i \tag{2.1.5}$$

satisfies, for all $c_1 < 0 < c_2$,

$$\lim_{N\to\infty} \mathcal{P}\left(\mathbb{E}[x]-\mathbb{E}_N[x] \in \left(c_1\sqrt{\frac{\mathrm{var}(x)}{N}}, c_2\sqrt{\frac{\mathrm{var}(x)}{N}}\right)\right) = \frac{1}{\sqrt{2\pi}}\int_{c_1}^{c_2} \mathrm{e}^{-\frac{x^2}{2}}\,\mathrm{d}x, \tag{2.1.6}$$

where $\mathcal{P}(y \in Y)$ stands for the probability that y belongs to Y.

The error due to the Euler scheme is as follows.

Theorem 2.1.2. Assume that r, σ are bounded and uniformly Lipschitz-continuous in time and space (implying at most linear growth at infinity). Then the $L^\infty(L^p)$-error, $p < 1$, is bounded by $c\sqrt{\delta t(1 - \ln \delta t)}$, for some constant c.

A proof is available in Pagès (2008), for instance.

Theorem 2.1.3. (Talay and Tubaro 1996) Let $f \in C^\infty(\mathbb{R})$. If r and σ are uniformly Lipschitz in time and Lipschitz in space, the error on $\mathbb{E}(f(S_T))$ when S_T is computed by (2.1.3) is of order $O(\delta t)$.

Remark 2.1.4. Note that $f : S_T \to (K - S_T)^+$ is not in $C^\infty(\mathbb{R})$. Consequently the precision of (2.1.4) is $O(\sqrt{\delta t}) + O\big(\frac{1}{\sqrt{N}}\big)$. However, if f were regular it would be $O(\delta t) + O\big(\frac{1}{\sqrt{N}}\big)$.

Remark 2.1.5. There exist schemes more precise than (2.1.3) (see Remark 2.1.6), but in the author's experience the improvement seems more theoretical than actual. On the other hand there are many extremely effective *variance reduction* techniques which reduce the constants multiplying the term $O\big(\frac{1}{\sqrt{N}}\big)$.

Most options can be simulated by the Monte Carlo method. For instance, a Monte Carlo simulation to price an Asian option (1.5.14) with arithmetic averaging is as follows.

Choose $N, \delta t$, set $C_0 = 0$.
For$(n = 0\,; n \le N;\ n = n + 1)$

S_0 given, set $A = 0$.
For$(t = 0\,; t \le T;\ t = t + \delta t)$

Call the random generator to simulate $W_{t+\delta t} - W_t$.
Compute $S_{t+\delta t} = S_t(1 + r\delta t + \sigma(W_{t+\delta t} - W_t))$.
Do $A = A + \frac{\delta t}{2t}(S_t + S_{t+\delta t})$.

Compute $\frac{\mathrm{e}^{-rT}}{N}(S_T - A_T)^+$ and add it to C_0.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Multi-level Monte Carlo

The most common method to speed up Monte Carlo is *variance reduction*, whereby one searches for an equivalent random process x with larger $\mathrm{var}(x)$, thus giving a better estimate in (2.1.6). Quasi-Monte Carlo is also popular but multi-level Monte Carlo achieves the same efficiency with less programming cost.

Following Giles (2008), a multi-level technique similar to multigrid is used. While standard Monte Carlo costs $O(\epsilon^{-3})$ operations (using the optimistic Theorem 2.1.2 with $\delta t = O(\epsilon)$ and $N = O(\epsilon^{-2})$) to reach a mean-square precision $O(\epsilon^2)$, multi-level Monte Carlo does the same in $O(\epsilon^{-2}\ln^2\epsilon)$.

So consider a multiple set of simulation with $\delta t = \frac{T}{2^m},\ m = 0, \dots, M$ to compute the pay-off P^m, and write

$$\mathbb{E}[P^M] = \mathbb{E}[P^0] + \sum_1^M \mathbb{E}[P^m - P^{m-1}].$$

The key point is to approximate $\mathbb{E}[P^m - P^{m-1}]$ by the law of large numbers with N_m simulations using the same Brownian path:

$$\mathbb{E}[P^m - P^{m-1}] \approx \frac{1}{N_m}\sum_{i=1}^{N_m}(P^{m,i} - P^{m-1,i}).$$

Remark 2.1.6. The theory suggests using the Milstein scheme, which is $O(\delta t)$ in the L^∞-norm, and gives a more robust error estimator and $O(\epsilon^2)$ operations:

$$S_{t+\delta t} = S_t\big(1 + r\delta t + \sigma(W_{t+\delta t} - W_t)\big) + \frac{S_t^2\sigma\partial_t\sigma}{2}\big((W_{t+\delta t} - W_t)^2 - \delta t\big).$$

2.2. The Longstaff–Schwartz algorithm for Bermudas

We follow the presentation of Clément, Lamberton and Pottier (2002). For clarity assume $r = 0$. After discretization on the discrete early exercise set times, $\{t_j\}_1^J$, (1.5.13) becomes

$$P_j = \max_{k\ge j}\mathbb{E}[(K - S_k)^+|S_j], \quad j = 1, \dots, J.$$

Following the usual dynamic programming paradigm, the same can be written as

$$P_J = (K - S_J)^+, \quad P_j = \max\{(K - S_j)^+, \mathbb{E}[P_{j+1}|S_j]\}.$$

It can also be rewritten so as to compute the exercise times τ_j:

$$\begin{aligned}
&\tau_J = T,\ P_J = (K - S_J)^+;\\
&\text{for } (\ j = J \text{ down to } j = 1)\\
&\qquad \text{if } (K - S_j)^+ \ge \mathbb{E}[P_{j+1}|S_j] \text{ do } \tau_j = t_j,\ P_j = (K - S_j)^+\\
&\qquad \text{else do } \tau_j = \tau_{j+1},\ P_j = \mathbb{E}[P_{j+1}|S_j]
\end{aligned} \tag{2.2.1}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Longstaff and Schwartz (1992) suggest using a least-squares projection on a polynomial basis to compute $\mathbb{E}[P_{j+1}|S_j]$.

For this, S_t is computed on a set of Monte Carlo paths indexed by i, giving S_j^i. Since $\mathbb{E}[P_{j+1}|S]$ is known from the previous iteration at the sample points $\{S = S_{j+1}^i\}_i$, then $\mathbb{E}[P_{j+1}|S = S_j]$ is computed by interpolation.

To project a function $y \to f(y)$ on $\{1, y, y^2, \ldots, y^n\}$ from a known set of values $f(y_i),\ i = 1, \ldots, I$, one solves

$$\min_{a_1,\ldots,a_n} \sum_{i=1}^{I} \left\| \sum_{m=1}^{n} a_m y_i^m - f(y_i) \right\|^2.$$

An implementation in C++ written by Tobias Lipp is available upon request by email to tobias.lipp@upmc.fr or olivier.pironneau@upmc.fr.

2.3. Methods using the process PDF for basket options

2.3.1. Basket options with constant volatilities

When the volatilities of the elements of a basket are constant, the computation of an option on the basket reduces to a multiple integral (1.5.20). For a basket with many elements ($d \gg 1$), we will show below how to use sparse grids to compute this integral efficiently.

2.3.2. Numerical simulation by Fourier transform

Take the case of a vanilla European call. $X_T = \ln(S_T)$ is a martingale so the PDF is $\rho(X_T - X_0)$. As above, the option is

$$C_0(X_0) = \mathrm{e}^{-rT} \int_{\mathbb{R}} (\mathrm{e}^{X_T} - K)^+ \rho(X_T - X_0)\, \mathrm{d}X_T.$$

As the right-hand side is a convolution, a Fourier transform changes the integral into a product. However the pay-off $(\mathrm{e}^x - \mathrm{e}^k)^+$ is not a k-integrable function on $\mathbb{R}$ so its Fourier transform is undefined. With an appropriate damping factor e^{ak}, Carr and Madan (1999) arrived at

$$\mathcal{F}[\mathrm{e}^{ak} C_0] = \frac{\mathrm{e}^{-rT} \Phi(z - (a+1)\mathrm{i})}{(a\mathrm{i} - z)(z - \mathrm{i}(1+a))}. \tag{2.3.1}$$

When X_t is driven by (1.1.5), with constant volatility, the characteristic function is

$$\Phi(z) = \mathrm{e}^{\mathrm{i}z(r - \frac{\sigma^2}{2})T - \frac{\sigma^2}{2} T z^2}.$$

However, this so-called 'CONV' method applies to more complex situations, whenever Φ is known for a monotone function of S_t, as is the case with Lévy processes (see Lord, Fang, Bervoets and Oosterlee (2007)). The computational cost is a single Fourier transform.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Basket options

The method also applies to basket options. With (4.3.6) the joint characteristic function is

$$\Phi(\vec{z}) := \mathbb{E}[\mathrm{e}^{i\vec{z}\cdot\vec{S}}|\vec{S}_0] = S_0\mathrm{e}^{\mathrm{i}T\sum_i(r-\frac{\sigma_i^2}{2}z_i)-\frac{T}{2}\sum_{i,j}z_iz_j\sigma_i\sigma_j\rho_{ij}}. \tag{2.3.2}$$

Damping factors do not seem to work, so the domain is truncated, $\mathbb{R}^d \approx D := (-L,L)^d$, and the following is computed approximately by using two FFTs:

$$C_0 = \mathrm{e}^{-rT}\mathcal{F}^{-1}\left[\mathcal{F}\left[\left(\sum_i \alpha_i\mathrm{e}^{y_i}-K\right)^+\mathbf{1}_D(\vec{y})\right]\Phi(-\vec{z})\right]. \tag{2.3.3}$$

Bermuda options

The same method is applied backwards in time on each interval $[t_i, t_{i+1}]$, where the contract cannot be exercised early. Then the pay-off is compared with the constraint ($(S-K)^+$ for a call) and the process is continued unless early exercise is better.

2.4. A binomial tree for American options

Trees are perhaps among the most popular tools for numerical finance. They are easy to understand and close to the probabilistic model, but their convergence is not so easy to establish. There are parallels between binomial trees and explicit finite difference methods. Because it is so well documented, only one example is given here, for the computation of an American option.

The time interval to maturity $(0,T)$ is divided into smaller intervals of size δt, days, for example. It is also assumed that the underlying asset is S_0 initially and can only take a finite set of values u^jS_0, indexed by $j = 1, 2, \ldots, J$. After n days, the underlying asset takes one of its possible values, S_j^n. Then it could either go up with probability p the next day to $S_{j+1}^{n+1} = uS_j^n$ or down to $S_{j-1}^{n+1} = \frac{1}{u}S_j^n$ with probability $1-p$. At time $T = N\delta t$, the probability p_j^N that $S_T = u^jS_0$ is known.

The American call is computed backwards in time from its value at T, knowing that S_T is u^jS_0, $C_j^N = (u^jS_0 - K)^+$:

$$C_j^n = \max\{\mathrm{e}^{-r\delta t}(pC_{j+1}^{n+1} + (1-p)C_{j-1}^{n+1}), (S_0u^j - K)^+\}. \tag{2.4.1}$$

Equivalence with a finite difference method

It will be shown later that C is also such that $C(T,x) = (\mathrm{e}^x - K)^+$ and

$$\min\left\{-\partial_tC - \frac{\sigma^2}{2}\partial_{xx}C - \left(r-\frac{\sigma^2}{2}\right)\partial_xC + rC, C - (\mathrm{e}^x-K)^+\right\} = 0,$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

where $x = \ln S$. Applying an explicit Euler scheme in time and central differences in x, the problem becomes

$$\min\left\{-\frac{1}{\delta t}(C_j^{n+1} - C_j^n) - \frac{\sigma^2}{2\delta x^2}(C_{j+1}^{n+1} - 2C_j^{n+1} + C_{j-1}^{n+1}) \right. \tag{2.4.2}$$
$$\left. - \left(r - \frac{\sigma^2}{2}\right)\frac{C_{j+1}^{n+1} - C_{j-1}^{n+1}}{2\delta x} + rC_j^{n+1}, C_j^{n+1} - (\mathrm{e}^{j\delta x} - K)^+\right\} = 0.$$

When $\delta x = \sigma\sqrt{\delta t}$ and $p = \frac{1}{2} + \frac{\sqrt{\delta t}}{2\sigma}(r - \frac{\sigma^2}{2})$, and u such that $S_0 u^j = j\delta x$, then both schemes are identical up to higher-order terms in δt. See Jiang and Dai (1999) for more details.

The website finance.bi.no/∼bernt/gcc_prog/recipes/recipes/ of Bernt Arne Ødegaard has more applications of binomial trees with the corresponding C++ codes.

PART THREE
Some partial differential equations in finance

3.1. From SDEs to PDEs and PIDEs

Numerical methods based on stochastic differential equations are usually easy to understand and very close to the probabilistic model. On the other hand they can be slow. The duality between the stochastic and deterministic formulations has been known at least since the discovery of molecular agitation as the underlying principle of temperature. Indeed, the heat equation is the macroscopic result of Brownian molecular agitation. The general rule is that if a deterministic quantity is a function of a Markov process, then there is a deterministic equation for it. Kolmogorov and Itô have laid the foundations and given the mathematical tools to understand and derive the (possibly integral) partial differential equations and inequalities in finance. Dupire has also contributed to the field, as we shall see.

3.2. Fokker–Planck and Kolmogorov equations

Recall that, almost by definition, the conditional probability density function $x \to \rho(x)$ of a random variable X, knowing that an event A has occurred, is related to the conditional expectation of $f(X_t)$, knowing A, by

$$\mathbb{E}(f(X)|A) = \int_{\mathbb{R}} f(x)\rho(x|A)\,\mathrm{d}x. \tag{3.2.1}$$

Consider the SDE

$$\mathrm{d}X_t = \mu_t\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t, \quad X_0 \text{ given.} \tag{3.2.2}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Kolmogorov (see Kolmogoroff 1931) showed that, when μ_t and σ_t are explicit functions of t and X_t, the PDF $X, T \to \rho(X_T|x,t)$ of X_τ, knowing that $X_t = x$, satisfies

$$\partial_T \rho - \partial_{XX}\left(\frac{\sigma(X,T)^2}{2}\rho\right) + \partial_X(\mu\rho(X,T)) = 0, \quad \rho|_t \text{ given}, \quad T > t. \tag{3.2.3}$$

This equation is known as the Kolmogorov forward or *Fokker–Planck equation* of X_T. It can be used to compute a European call, for instance,

$$C_0 = \mathrm{e}^{-(r-\frac{\sigma^2}{2})T}\int_{\mathbb{R}^+}\rho(X,T)(\mathrm{e}^X - K)^+\,\mathrm{d}X,$$

where ρ is the solution of (3.2.3) with initial condition $\rho(x,0) = \delta(\mathrm{e}^x - S_0)$, the Dirac mass at 0 (see (1.1.5)).

Conversely, if the statistics of X_T are known instead of those of X_t, the *Kolmogorov backward equation* may be used to compute $x, t \to \rho$:

$$\partial_t \rho + \frac{\sigma(x,t)^2}{2}\partial_{xx}\rho + \mu(x,t)\partial_x\rho = 0, \quad \rho|_T \text{ known}, \quad t < T. \tag{3.2.4}$$

From this the *Feynman–Kac formula* can be derived. Let u be the solution of

$$\partial_t u + \frac{\sigma(x,t)^2}{2}\partial_{xx}u + \mu(x,t)\partial_x u - ru = 0, \quad u|_T \text{ known}, \quad t < T. \tag{3.2.5}$$

Let X_t be a solution of the SDE (3.2.2). Then

$$u(x,t) = \mathbb{E}[\mathrm{e}^{-r(T-t)}u(X_T,T)|X_t = x]. \tag{3.2.6}$$

Corollary 3.2.1. The European call (1.5.1) can be computed by

$$C_t = u(\log(S_t), t),$$

where u is the solution of (3.2.5) with $u(x,T) = (\mathrm{e}^x - K)^+$ and $\mu = r - \frac{1}{2}\sigma^2$.

Proof. The Feynman–Kac formula can be applied to (1.1.5) with $u(x,t) = C_t|X_t = x$. It shows that if u satisfies (3.2.5) with $u(x,T) = (\mathrm{e}^x - K)^+$, then (3.2.6) holds and it is the definition of C_t. □

3.2.1. Itô's lemma

Let X_t be a solution of the SDE (3.2.2) and let $f(x,t)$ be twice differentiable in x and once in t. Then Itô's lemma says

$$\mathrm{d}f(X_t,t) = \frac{\partial f}{\partial x}(X_t,t)\,\mathrm{d}X_t + \frac{\partial f}{\partial t}(X_t,t)\,\mathrm{d}t + \frac{\sigma_t^2}{2}\frac{\partial^2 f}{\partial x^2}(X_t,t)\,\mathrm{d}t. \tag{3.2.7}$$

Note that when f is non-random, *i.e.*, $f = \bar{f} := \mathbb{E}[f(X_\tau,\tau|X_t = x]$, then by taking the expected value of (3.2.7) and (3.2.2),

$$\mathrm{d}\bar{f}(X_t,t) = \frac{\partial \bar{f}}{\partial x}(X_t,t)\mu\,\mathrm{d}t + \frac{\partial \bar{f}}{\partial t}(X_t,t)\,\mathrm{d}t + \frac{\mathbb{E}[\sigma_t^2]}{2}\frac{\partial^2 \bar{f}}{\partial x^2}(X_t,t)\,\mathrm{d}t. \tag{3.2.8}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

For more details, see for example Øksendal (1998).

3.2.2. Application to European options

Let us apply (3.2.7) to

$$C(S,t) := \mathbb{E}[\mathrm{e}^{-r(T-t)}(S_T - K)^+ | S_t = S], \quad \text{with } \mathrm{d}S_t = S_t(r\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t).$$

Then

$$\begin{aligned}\mathrm{d}C &= \frac{\partial C}{\partial S}(S_t,t)\,\mathrm{d}S_t + \frac{\partial C}{\partial t}(S_t,t)\,\mathrm{d}t + \frac{S_t^2\sigma_t^2}{2}\frac{\partial^2 C}{\partial S^2}(S_t,t)\,\mathrm{d}t \\ &= \left(rS_t\frac{\partial C}{\partial S} + \frac{\partial C}{\partial t} + \frac{S_t^2\sigma_t^2}{2}\frac{\partial^2 C}{\partial S^2}\right)\mathrm{d}t + \frac{\partial C}{\partial S}S_t\sigma_t\,\mathrm{d}W_t + o(\mathrm{d}t). \qquad (3.2.9)\end{aligned}$$

Note that while C_t is not random, $\mathrm{d}C$ is.

3.2.3. Application to hedging

Is it possible to build a riskless portfolio π_t made of κ parts of S_t plus one C_t? If so, no arbitrage requires that it evolves like a riskless asset, hence $\mathrm{d}\pi_t = r\pi_t\,\mathrm{d}t$. On the other hand

$$\begin{aligned}\mathrm{d}\pi_t = \kappa\,\mathrm{d}S_t + \mathrm{d}C_t &= \kappa S_t(r\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t) + \frac{\partial C}{\partial S}S_t\sigma_t\,\mathrm{d}W_t \\ &\quad + \left(rS_t\frac{\partial C}{\partial S} + \frac{\partial C}{\partial t} + \frac{S_t^2\sigma_t^2}{2}\frac{\partial^2 C}{\partial S^2}\right)\mathrm{d}t. \qquad (3.2.10)\end{aligned}$$

Accordingly π_t is non-random only if the factor of $\mathrm{d}W_t$ vanishes, namely $\kappa = -S\partial_S C$. Furthermore, the above simplifies to

$$r\pi_t\,\mathrm{d}t = r(\kappa S_t + C_t)\,\mathrm{d}t = \kappa S_t r\,\mathrm{d}t + \left(rS_t\frac{\partial C}{\partial S} + \frac{\partial C}{\partial t} + \frac{S_t^2\sigma_t^2}{2}\frac{\partial^2 C}{\partial S^2}\right)\mathrm{d}t.$$

Consequently we have the following result.

Theorem 3.2.2. A European call C_t on S, knowing S_t, can be priced by solving for $C(S,t)$ the backwards-in-time parabolic PDE

$$\begin{aligned}\partial_t C + \frac{S^2\sigma^2}{2}\partial_{SS}C + rS\partial_S C - rC = 0, \quad \forall S,t \in \mathbb{R}^+ \times (0,T), \\ C(S,T) = (S-K)^+, \quad \forall S \in \mathbb{R}^+, \qquad (3.2.11)\end{aligned}$$

and then set $C_t = C(S_t,t)$. Furthermore the portfolio $\kappa S_t + C_t$ is riskless when $\kappa = -S\partial_S C(S_t,t)$.

Remark 3.2.3. **(1)** Equation (3.2.11) is known as the Black–Scholes PDE.

(2) By a change of coordinate $x = \log(S)$, it is seen that (3.2.11) and (3.2.5) are the same.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

(3) Hedging is not necessary to derive (3.2.11); it is not an additional hypothesis to the Black–Scholes model. Since it leads to an extra result on riskless portfolios, it is generally preferred over the Feynman–Kac formula to derive the PDEs of finance.

(4) A similar equation can be derived for the European put, either with a riskless portfolio made of a put and the asset, or by using the put–call parity on (3.2.11).

(5) More generally, if the premium is $\phi(S_T)$ instead of $(S_T - K)^+$ and/or if r, σ depend on time and/or S_t, the same equation (3.2.11) holds but with $C(S,T) = \phi(S)$.

3.2.4. Application to basket options

Recall (1.5.15) written under the risk-neutral probability law, namely that the pay-off ϕ of the option is a function of

$$\mathrm{d}S_{it} = S_{it}(r\,\mathrm{d}t + \sigma_i\,\mathrm{d}W_{it}), \quad \text{with } \rho_{i,j}t := \mathbb{E}[W_{it}W_{jt}]. \tag{3.2.12}$$

No arbitrage and Itô calculus say that

$$\begin{aligned}
\mathrm{e}^{r\,\mathrm{d}t}\phi =\ & \mathrm{d}\phi \\
&= \mathbb{E}\left[\partial_t\phi\,\mathrm{d}t + \sum_i \partial_{S_i}\phi\,\mathrm{d}S_{it} + \frac{1}{2}\sum_{ij}\partial_{S_iS_j}\phi S_iS_j\,\mathrm{d}W_{it}\,\mathrm{d}W_{jt}|S_{i0}\right] \\
&= \left[\partial_t\phi + \sum_i rS_i\partial_{S_i}\phi + \frac{1}{2}\sum_{ij} S_iS_j\sigma_{it}\sigma_{jt}\rho_{ij}\partial_{S_iS_j}\phi\right]\mathrm{d}t.
\end{aligned} \tag{3.2.13}$$

With $\tau = T - t$, this equation can be written in compact form as

$$\partial_\tau\phi - r\vec{S}\cdot\nabla\phi - \Xi : \nabla\nabla\phi + r\phi = 0, \quad \phi(\vec{S},0) \text{ given}, \tag{3.2.14}$$

with $\Xi_{ij} = S_iS_j\sigma_{it}\sigma_{jt}\rho_{ij}/2$ and the notation $A : B = \sum_{ij} A_{ij}B_{ij}$ as the trace of the matrix AB^T.

3.2.5. Application to Asian options

With $A_t = \frac{1}{t}\int_0^t S_\tau\,\mathrm{d}\tau$ and $\mathrm{d}S_t = S_t(r\,\mathrm{d}t + \sigma_t\,\mathrm{d}W_t)$, $S(0) = S_0$, an Asian call $C_t = \mathrm{e}^{-r(T-t)}\mathbb{E}[(A_T - K)^+|S_t]$ satisfies

$$\begin{aligned}
\mathrm{e}^{r\,\mathrm{d}t}C_t =\ \mathrm{d}C_t &= \partial_tC\,\mathrm{d}t + \partial_AC\,\mathrm{d}A_t + \frac{1}{2}\partial_{AA}C(\mathrm{d}A)^2 + \partial_SC\,\mathrm{d}S_t \\
&\quad + \frac{1}{2}\partial_{SS}C(\mathrm{d}S)^2 + \partial_{AS}C\,\mathrm{d}S\,\mathrm{d}A \\
&= \partial_tC\,\mathrm{d}t + \partial_AC(S - A)\frac{\mathrm{d}t}{t} + \partial_SC(Sr\,\mathrm{d}t + S\sigma\,\mathrm{d}W_t) \\
&\quad + \frac{1}{2}\partial_{SS}C(\sigma S)^2\,\mathrm{d}t + o(\mathrm{d}t),
\end{aligned} \tag{3.2.15}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

because $\mathrm{d}(tA_t) = t\,\mathrm{d}A_t + A_t\,\mathrm{d}t = S_t\,\mathrm{d}t$.

As for vanilla European calls, a portfolio $\pi_t = \kappa S_t + C_t$ with variation $\kappa\,\mathrm{d}S_t + \mathrm{d}C_t$ will be riskless only if $\kappa = -S\partial_S C(S_t, t)$, and so by saying that $\mathrm{d}\pi = r\pi\,\mathrm{d}t$, the following is found for all $S \in \mathbb{R}^+, A \in \mathbb{R}^+, t \in (0, T)$:

$$\partial_t C + \frac{S - A}{t}\partial_A C + rS\partial_S C + \frac{\sigma^2 S^2}{2}\partial_{SS} C - rC = 0,$$
$$C(S, A, T) = (A - K)^+. \tag{3.2.16}$$

Let us set $C(S, A, t) = u(\mathrm{e}^{r(T-t)}S, tA, T - t)$. Then, in terms of $x = \mathrm{e}^{r(T-t)}S, y = tA, \tau = T - t$, the PDE for u is

$$\partial_\tau u - x\mathrm{e}^{-r\tau}\partial_y u - \frac{\sigma^2 x^2}{2}\partial_{xx} u + ru = 0,$$
$$u(x, y, 0) = \left(\frac{y}{T} - K\right)^+, \quad \forall x, y, \tau \in \mathbb{R}^+ \times \mathbb{R}^+ \times (0, T). \tag{3.2.17}$$

Additional boundary conditions are needed, such as

$$u(x, 0, \tau) = 0, \quad u(0, y, \tau) = \mathrm{e}^{-r\tau}\left(\frac{y}{T} - K\right)^+, \quad \forall x, y, \tau \in \mathbb{R}^+ \times \mathbb{R}^+ \times (0, T). \tag{3.2.18}$$

An asymptotic condition can be found if $yT > K$ and $x \to \infty$. Then

$$u \sim \mathrm{e}^{-r\tau}\left[y + \frac{1 - \mathrm{e}^{-r\tau}}{r}x - KT\right]/T.$$

3.2.6. From jump processes to partial integro-differential equations

The *generalized Black–Scholes model* uses a jump-diffusion process X_t, given by (1.4.3) instead of the Wiener process W_t. Hence consider the following model for the underlying asset S_t:

$$\mathrm{d}S_t = S_t^-(r\,\mathrm{d}t + \mathrm{d}X_t), \quad S_0 \text{ given, } \textit{i.e.}, S_t = S_0\mathrm{e}^{rt + X_t}. \tag{3.2.19}$$

A corollary to (1.4.6) is that $\mathrm{e}^{rt}S_t$ is a martingale if and only if

$$\gamma + \frac{\sigma^2}{2} + \int_{\mathbb{R}}(\mathrm{e}^x - 1 - x\mathbf{1}_{|x|<1})\nu(\mathrm{d}x) = 0. \tag{3.2.20}$$

Now, because (1.4.5) holds, it can be proved (Tankov and Voltchkova 2009) that a call option on S_t with pay-off $(S_t - K)^+$, namely

$$C_t = \mathrm{e}^{-r(T-t)}\mathbb{E}[(S_T - K)^+ | S_t = S),$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

is also given by $C(S,t)$, where C is the solution of

$$\partial_t C + \frac{\sigma^2 S^2}{2}\partial_{SS}C + r\partial_S C - rC + \int_{\mathbb{R}} [C(S\mathrm{e}^y,t) - C(S,t) - S(\mathrm{e}^y - 1)\partial_S C(S,t)]\nu(\mathrm{d}y) = 0,$$
$$C(S,T) = (S-K)^+. \tag{3.2.21}$$

Remark 3.2.4. With $x = \ln S$, $\tau = T - t$ and $u(x,\tau) = C(\mathrm{e}^x, T-\tau)$ then (3.2.22) becomes

$$\partial_\tau u - \frac{\sigma^2}{2}\partial_{xx}u - \left(r - \frac{\sigma^2}{2}\right)\partial_x u + ru - \int_{\mathbb{R}} [u(x+y,\tau) - u(x,\tau) - (\mathrm{e}^y - 1)\partial_x u(x,\tau)]\nu(\mathrm{d}y) = 0,$$
$$u(x,0) = (\mathrm{e}^x - K)^+. \tag{3.2.22}$$

Theorem 3.2.5. Assume $\sigma > 0$ and, for some $\beta \in (0,2)$,

$$\int_{|y|>1} \mathrm{e}^{2y}\nu(\mathrm{d}y) < \infty, \quad \liminf_{\epsilon\to 0} \epsilon^{-\beta}\int_{-\epsilon}^{\epsilon} |y|^2\nu(\mathrm{d}y) > 0. \tag{3.2.23}$$

Then $C(S,t) = \mathrm{e}^{-r(T-t)}\mathbb{E}[(S_T - K)^+ | S_t = S]$ is continuous on $\mathbb{R}^+ \times [0,T]$, C^1 in t and C^2 in S on $\mathbb{R}^+ \times (0,T)$ and satisfies (3.2.22).

Proof. The leading idea is to apply Itô's formula to the martingale $\hat{C}_t = \mathrm{e}^{r\tau}C(S_t,t)$ with $\tau = T - t$.

The second condition in (3.2.23) implies that X_t has a smooth C^2 density and so $C(S,t)$ has the regularity mentioned in the theorem.

The fact that $\int_{|y|>1} \mathrm{e}^{2y}\nu(\mathrm{d}y) < \infty$ implies that there exists J_X such that

$$\mathrm{d}S_t = S_t\left(\sigma\,\mathrm{d}W_t + \int_{\mathbb{R}} (\mathrm{e}^x - 1)J_X(\mathrm{d}x\,\mathrm{d}t)\right).$$

Skipping the details (see Cont and Voltchkova (2004), Cont and Tankov (2003)),

$$\mathrm{d}\hat{C}_t = a\,\mathrm{d}t + \mathrm{d}M_t, \tag{3.2.24}$$

with

$$\mathrm{e}^{-r\tau}a = -rC + \partial_t C + \frac{\sigma^2 S^2}{2}\partial_{SS}C + rS\partial_S C + \int_{\mathbb{R}} (C(S\mathrm{e}^x,t) - C(S,t) - S(\mathrm{e}^x - 1)\partial_S C(S,t))\nu(\mathrm{d}x),$$
$$\mathrm{e}^{-r\tau}\,\mathrm{d}M_t = \partial_S C\sigma S\,\mathrm{d}W_t + \int_{\mathbb{R}} (C(S\mathrm{e}^x,t) - C(S,t)J_X(\mathrm{d}t\,\mathrm{d}x). \tag{3.2.25}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Then one shows that M_t is a square-integrable martingale and so is $\hat{C}_t - M_t$. Thus, by taking the expected value of (3.2.24), we obtain that $a = 0$. □

Popular choices for ν are $\nu(\mathrm{d}y) = k(y)\,\mathrm{d}y$ with the *variance gamma model*,

$$k(y) = \frac{\omega}{|y|}\mathrm{e}^{-y^+\eta^+ - y^-\eta^-}, \tag{3.2.26}$$

with $\eta^\pm > 2$, the *CGMY model* (Carr, Geman, Madan and Yor 2002),

$$k(y) = \frac{\omega}{|y|^{1+Y}}\mathrm{e}^{-y^+\eta^+ - y^-\eta^-}, \tag{3.2.27}$$

with $0 < Y < 2,\ 0 < \eta^-,\ 2 < \eta^+$, and *Merton's model*,

$$k(x) = \frac{\lambda}{\sqrt{2\pi}\delta}\exp\left(-\frac{(y-\mu)^2}{2\delta^2}\right). \tag{3.2.28}$$

Theorem 3.2.6. In the case of Merton's model C, the call on S at $t = 0$, can be computed by

$$C = \mathrm{e}^{-rT}\sum_0^\infty \frac{\mathrm{e}^{-\lambda T}(\lambda T)^n}{n!}\mathrm{e}^{r_n T}C_{BS}(S_0\mathrm{e}^{n\delta^2/T}, K, T, \sigma_n, r_n),$$

where C_{BS} refers to the Black–Scholes formula and

$$r_n = r - \lambda(\mathrm{e}^{\mu+\delta^2/2} - 1) + \mu n T, \quad \sigma_n = \sqrt{\sigma^2 + n\frac{\delta^2}{2}}.$$

Remark 3.2.7. Note that the variance gamma model and the CGMY model do not satisfy the second hypothesis in (3.2.23). Another proof must be made, using viscosity solutions, for instance.

3.2.7. Lévy copula

An extension to basket options can be found in Farkas *et al.* (2006). The following PIDE is found, in variational form (see below):

$$\int_{\mathbb{R}^d}\partial_\tau u(x,\tau)v(x)\,\mathrm{d}x + \int_{\mathbb{R}^d}\nabla v^T\boldsymbol{\Xi}\nabla u - \int_{\mathbb{R}^d} v\mathbf{b}\cdot\nabla u = J(u,v), \quad \forall v \in V, \tag{3.2.29}$$

where

$$J(u,v) = \frac{1}{2}\int_{x\in\mathbb{R}^d}\int_{y\in\mathbb{R}^d}(u(x+y)-u(x))(v(x+y)-v(x))\tilde{\nu}(\mathrm{d}y)\,\mathrm{d}x.$$

Here V is a Sobolev space with anisotropic weight, $\boldsymbol{\Xi}$ is the variance-correlation matrix (see (3.2.14)) and $\mathbf{b}$ is the drift in log price variables,

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

made up of the usual drift proportional to r, plus the term proportional to $\nabla\Xi$, plus the part of the integro-differential term corresponding to $(e^y - 1)\partial_x u$ in the scalar case. The symmetry of J corresponds to a special case in the general theory.

3.2.8. From constrained SDEs to PDIs

American option

Let P_t be an American put on S_t with strike K and maturity T. We have seen above that unless $P_t \geq (K - S_t)^+$ the contract ceases to exist. As for a European call, the put $u(S_t, t) = e^{-(T-t)}\mathbb{E}[(K - S_T)^+|S_t]$ can be estimated by computing $\mathbb{E}[\,\mathrm{d}u]$ with Itô's lemma. Intuitively, no arbitrage implies that that $\mathbb{E}\,\mathrm{d}u \leq ru\,\mathrm{d}t$, so in addition to $P(S, T) = (K - S_T)^+$, we have

$$\partial_t P + \frac{\sigma^2 S^2}{2}\partial_{SS} P + rS\partial_S P - rP \leq 0, \quad \text{or} \quad P(S, t) \geq (K - S)^+, \tag{3.2.30}$$

and at every S, t, one of the two, at least, is an equality. This can be summarized by the requirement that, for all $S > 0, t \in (0, T)$,

$$\min\left\{-\partial_t P - \frac{\sigma^2 S^2}{2}\partial_{SS} P - rS\partial_S P + rP, P - (K - S)^+\right\} = 0. \tag{3.2.31}$$

Convertible bond

Suppose the bond has a probability of default p with a fraction of return R. Let V denote the convertible, B the bond component, q the dividend and S the share. Then Hull (2003) suggests the following:

$$\begin{aligned}
&\partial_t V + \frac{\sigma^2 S^2}{2}\partial_{SS} V + (r - q)S\partial_S V - rV = sB, \quad V_T = \max\{B_T, \kappa S_T\},\\
&\partial_t B + \frac{\sigma^2 S^2}{2}\partial_{SS} B + rS\partial_S B - (r + s)B = 0, \quad B_T \text{ given},
\end{aligned} \tag{3.2.32}$$

where

$$\text{if } B_c < \kappa S, \text{ then } V = \kappa S, \text{ else } \max\{B_p, \kappa S_t\} \leq V_t \leq B_c, \tag{3.2.33}$$

where the spread s is related to p and R by $s = p(1 - R)$. Assuming B_c constant in time, for simplicity, the ambiguity between the constraints and the PDE is resolved by:

$$\begin{aligned}
&\text{if } B_c < \kappa S \text{ then } V = \kappa S \text{ else}\\
&\max\left\{\max\{B_p, \kappa S\} - V, \partial_t V + \frac{\sigma^2 S^2}{2}\partial_{SS} V + (r - q)S\partial_S V - rV + sB\right\}\\
&\quad = 0.
\end{aligned} \tag{3.2.34}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

3.2.9. PDEs for interest rates and swaps

Options on interest rates

Letting $\bar{r}(t)$ denote the mean interest rate at T and coming back to (1.5.23) and (1.5.22), an option on S_t with pay-off

$$C(S,r,\sigma,t) = \mathbb{E}\big[\mathrm{e}^{-\frac{1}{T}\int_t^T r(s)\,\mathrm{d}s}\phi(S_T)|S_t = S, r_t = r, \sigma_t = \sigma\big]$$

will satisfy, by Itô's lemma and the no arbitrage rule,

$$\begin{aligned} C(S,r,\sigma,t)r\,\mathrm{d}t = \ & \mathrm{d}C = \partial_t C\,\mathrm{d}t + \partial_S C\mathbb{E}[\mathrm{d}S] + \partial_r C\mathbb{E}[\mathrm{d}r] + \partial_\sigma C\mathbb{E}[\mathrm{d}\sigma] \\ & + \frac{1}{2}\partial_{SS}C\mathbb{E}[\mathrm{d}S^2] + \frac{1}{2}\partial_{rr}C\mathbb{E}[\mathrm{d}r^2] + \frac{1}{2}\partial_{\sigma\sigma}C\mathbb{E}[\mathrm{d}\sigma^2] \\ & + \partial_{Sr}C\mathbb{E}[\mathrm{d}S\,\mathrm{d}r] + \partial_{S\sigma}C\mathbb{E}[\mathrm{d}S\,\mathrm{d}\sigma] + \partial_{r\sigma}C\mathbb{E}[\mathrm{d}S\,\mathrm{d}\sigma] \\ = \ & \partial_t C\,\mathrm{d}t + Sr\,\mathrm{d}t\partial_S C + \lambda(m-r)\,\mathrm{d}t\partial_r C \\ & + \frac{1}{2}\sigma^2 S^{2\beta}\rho_{22}\,\mathrm{d}t\partial_{SS}C + \frac{1}{2}\eta^2\rho_{11}\,\mathrm{d}t\partial_{rr}C + \frac{1}{2}\gamma^2\rho_{33}\,\mathrm{d}t\partial_{\sigma\sigma}C \\ & + \sigma S^\beta\eta\rho_{21}\,\mathrm{d}t\partial_{Sr}C + \gamma\sigma S^\beta\rho_{23}\,\mathrm{d}t\partial_{S\sigma}C + \gamma\eta\rho_{31}\,\mathrm{d}t\partial_{r\sigma}C. \end{aligned} \tag{3.2.35}$$

So the PDE of the problem is in $\mathbb{R}^{+3}$: find C with $C(S,r,\sigma,T) = \phi(S)$ and

$$\begin{aligned} & \partial_t C - rC + Sr\partial_S C + \lambda(m-r)\partial_r C \\ & \quad + \frac{1}{2}\sigma^2 S^{2\beta}\rho_{22}\partial_{SS}C + \frac{1}{2}\eta^2\rho_{11}\partial_{rr}C + \frac{1}{2}\gamma^2\rho_{33}\partial_{\sigma\sigma}C \\ & \quad + \sigma S^\beta\eta\rho_{21}\partial_{Sr}C + \gamma\sigma S^\beta\rho_{23}\partial_{S\sigma}C + \gamma\eta\rho_{31}\partial_{r\sigma}C = 0. \end{aligned} \tag{3.2.36}$$

Credit default swaps

We have seen above that the spread s paid by the buyer is a deterministic function of the probability of default, which is itself a linear function of the two SDEs (1.5.28). Itô calculus gives

$$\begin{aligned} \mathrm{d}s = \ & \partial_t s\,\mathrm{d}t + \partial_\xi s\mathbb{E}[\mathrm{d}\xi] \\ & + \partial_\lambda s\mathbb{E}[\mathrm{d}\lambda] + \frac{1}{2}\partial_{\xi\xi}s\mathbb{E}[\mathrm{d}\xi^2] + \partial_{\xi\lambda}s\mathbb{E}[\mathrm{d}\xi\,\mathrm{d}\lambda] + \frac{1}{2}\partial_{\lambda\lambda}s\mathbb{E}[\mathrm{d}\lambda^2] \\ = \ & \mathrm{d}t\bigg(\partial_t s + a(\xi_m - \xi)\partial_\xi s + \alpha(\lambda_m - \lambda)\partial_\lambda s \\ & \quad + \frac{c^2\xi}{2}\partial_{\xi\xi}s + c\sigma\sqrt{\xi\lambda}\partial_{\xi\lambda}s + \frac{\sigma^2\lambda}{2}\partial_{\lambda\lambda}s\bigg). \end{aligned} \tag{3.2.37}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

If the buyer's spread evolves with the current interest rate then the following PDE holds for s:

$$\begin{aligned}&\partial_t s + a(\xi_m - \xi)\partial_\xi s + \alpha(\lambda_m - \lambda)\partial_\lambda s\\ &\qquad + \frac{c^2\xi}{2}\partial_{\xi\xi} s + c\sigma\sqrt{\xi\lambda}\partial_{\xi\lambda} s + \frac{\sigma^2\lambda}{2}\partial_{\lambda\lambda} s = rs,\\ &s(\xi,\lambda,T) = \phi(\xi,\lambda) \quad \text{given by (1.5.29).}\end{aligned} \tag{3.2.38}$$

3.3. Some existence and regularity results

3.3.1. Existence results for the Black–Scholes PDE

By Theorem 3.2.2 a plain vanilla put P can be computed by solving

$$\partial_t P + \frac{\sigma^2 S^2}{2}\partial_{SS} P + rS\partial_S P - rP = 0, \text{ or } P(S,t) \geq (K-S)^+, \tag{3.3.1}$$

for all $S, t \in \mathbb{R}^+ \times (0,T)$. For clarity we assume r constant (but not σ).

Change of variables
Let

$\tau := T - t$, the time to maturity,

$y := \mathrm{e}^{r(T-t)}\dfrac{S}{K}$, known as the *forward moneyness*,

$x := \ln y$ (*i.e.*, $S = K\mathrm{e}^{x-(r-d)\tau}$), the *log forward moneyness variable*.

Proposition 3.3.1. Let $P(S,t)$ be a solution of (3.3.1), and let

$$v(y,\tau) := \frac{\mathrm{e}^{r\tau}}{K}P(yK\mathrm{e}^{-r\tau}, T-\tau), \quad u(x,\tau) := \frac{\mathrm{e}^{r\tau}}{K}P(K\mathrm{e}^{x-r\tau}, T-\tau). \tag{3.3.2}$$

Then

$$\partial_\tau v - \frac{\sigma^2 y^2}{2}\partial_{yy} v = 0 \quad \text{in } \mathbb{R}^+ \times (0,T), \quad v(y,0) = (1-y)^+, \tag{3.3.3}$$

$$\partial_\tau u + \frac{\sigma^2}{2}\partial_x u - \frac{\sigma^2}{2}\partial_{xx} u = 0 \quad \text{in } \mathbb{R} \times (0,T), \qquad u(x,0) = (1-\mathrm{e}^x)^+. \tag{3.3.4}$$

Proof. Note that $(S-K)^+ = K(\frac{S}{K}-1)^+$ and $S^2\partial_{SS} = (\frac{S}{K})^2\partial_{\frac{S}{K}\frac{S}{K}}$. Finally note that

$$\begin{aligned}K\partial_\tau v(y,\tau) &= \partial_\tau\big(\mathrm{e}^{r\tau}P(yK\mathrm{e}^{-r\tau}, T-\tau)\big)\\ &= r\mathrm{e}^{r\tau}P(S,t) - \mathrm{e}^{r\tau}\partial_t P(S,t) - \mathrm{e}^{r\tau}rKy\mathrm{e}^{-r\tau}\partial_S P(S,t)\\ &= \mathrm{e}^{r\tau}(rP - \partial_t P - S\partial_S P).\end{aligned} \tag{3.3.5}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

To obtain the equation for u, one makes the change of variable $x = \ln y$ in (3.3.3):

$$\partial_x = \partial_x(y)\partial_y = \partial_x(\mathrm{e}^x)\partial_y = y\partial_y,$$
$$\partial_y(\partial_y) = \mathrm{e}^{-x}\partial_x(\mathrm{e}^{-x}\partial_x) = (\mathrm{e}^{-x})^2(\partial_{xx} - \partial_x). \qquad \square$$

A direct proof of existence of solution for (3.3.3) or (3.3.4) is possible but hard to generalize to the multi-dimensional case, so we prefer to use the variational framework, at the cost of imposing some regularity on σ.

Variational formulation

Consider

$$H = \{w \in L^2(\mathbb{R}^+) : y\partial_y w \in L^2(\mathbb{R}^+)\}. \tag{3.3.6}$$

It can be shown (Achdou and Pironneau 2005) that H is a Hilbert space with the scalar product and norm

$$\langle a, b\rangle = \int_{\mathbb{R}^+} [ab + y^2\partial_y a\partial_y b]\,\mathrm{d}y, \quad \|a\| = \left(\int_{\mathbb{R}^+} [a^2 + y^2(\partial_y a)^2]\,\mathrm{d}y\right)^{\frac{1}{2}},$$

for $a, b \in H$. The variational formulation of (3.3.3) is obtained by multiplying it by $w \in H$, integrating it on $\mathbb{R}^+$ and integrating the second derivative in y by parts: find $v \in L^2(0,T;H)$ with $v(y,0) = (1-y)^+$, such that, for all $w \in H$,

$$\int_{\mathbb{R}^+} \left[w(y)\partial_\tau v(y,\tau) + \partial_y\left(w(y)\frac{\sigma^2 y^2}{2}\right)\partial_y v(y,\tau)\right]\mathrm{d}y = 0, \text{ a.e. in } (0,T). \tag{3.3.7}$$

We recall the result established in Achdou and Pironneau (2005).

Theorem 3.3.2. If there exist three positive constants $\sigma_m, \sigma_M, C_\sigma$ such that

$$\sigma_m \le \sigma(S,t) \le \sigma_M, \quad |S\partial_S\sigma(S,t)| \le C_\sigma, \tag{3.3.8}$$

then (3.3.7) has a solution in $L^2(0,T;H)$ and the solution is unique. Furthermore $S \to P(S,t)$ is positive bounded by $K\mathrm{e}^{-r(T-t)}$ and convex almost everywhere.

Proof. The proof relies on an extension of the usual theory of parabolic equations, where the coercivity of the bilinear form is established with Gårding's inequality (see Achdou and Pironneau (2005)) for the bilinear form of the PDE:

$$a(u,u) \ge \alpha\|u\|^2 - \beta\|u\|^2_{L^2(\mathbb{R}^+)}.$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Indeed, by replacing the time derivative by a time difference of size $\delta\tau$ and renaming $v^m(y) := v(y, m\delta\tau)$, (3.3.7) becomes

$$\int_{\mathbb{R}^+}\left[w\frac{v^m - v^{m-1}}{\delta t} + \partial_y\left(w\frac{\sigma^2y^2}{2}\right)\partial_y v^m\right]\mathrm{d}y = 0.$$

It is a problem of the following type: find $v^m \in H$ such that

$$a(v^m, w) = (f, w) \quad \forall w \in H,$$

where

$$a(v,w) = \int_{\mathbb{R}^+}\left[w\frac{v}{\delta t} + \partial_y\left(w\frac{\sigma^2y^2}{2}\right)\partial_y v\right]\mathrm{d}y \quad \text{and} \quad f = \frac{v^{m-1}}{\delta\tau}.$$

The fundamental result of the variational framework is that if a is bilinear continuous and satisfies Gårding's inequality for some strictly positive α, β and if $f \in H$, then the solution v exists and is unique. Here

$$\begin{aligned} a(v,v) &\geq \int_{\mathbb{R}^+}\left[\frac{v^2}{\delta\tau} + \frac{\sigma_m^2}{2}y^2(\partial_y v)^2\right] + \int_{\mathbb{R}^+}\left(\partial_y\frac{y\sigma^2}{2}\right)v(y\partial_y v) \\ &\geq \min\left\{\frac{1}{\delta\tau}, \frac{\sigma_m^2}{2}\right\}\|v\|^2 - \left(\max_y \partial_y\frac{y\sigma^2}{2}\right)\|v\|\|v\|_0 \\ &\geq \min\left\{\frac{1}{\delta\tau}, \frac{\sigma_m^2}{2}\right\}\|v\|^2 - \left(\frac{\sigma_m^2}{4}\|v\|^2 + \lambda\|v\|_0^2\right) \end{aligned} \tag{3.3.9}$$

for some bounded λ as long as $\partial_y(y\sigma^2)$ is bounded too, pointwise. □

The following measures the effect of *localization*, *i.e.*, the error due to the replacement of $\mathbb{R}^+$ in (3.3.1) by $(0, S_M)$. Naturally a boundary condition is needed: $P(S_M, t) = 0$ for all $t \in (0, T)$.

Proposition 3.3.3. With the hypotheses of Theorem 3.3.2, the $L^\infty(\mathbb{R}^+ \times (0,T))$-error between the solution of (3.3.7) and the solution of (3.3.7) in which $\mathbb{R}^+$ and H are replaced by $(0, S_M)$ and $\{w \in H : w(S_M) = 0\}$, decays faster than any exponential $\mathrm{e}^{-\eta S_M}$ as $S_M \to \infty$.

Proof. See Achdou and Pironneau (2005). □

A further change of variables

Assume σ constant. Note that

$$\frac{\mathrm{d}}{\mathrm{d}\tau}u\left(\tau, x + \frac{\sigma^2}{2}\tau\right) = \partial_\tau u + \frac{\sigma^2}{2}\partial_x u.$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Let $w(x,\tau) = u\big(\tau, x + \frac{\sigma^2}{2}\tau\big)$, where u is the solution of (3.3.4). Then

$$\partial_\tau w - \frac{\sigma^2}{2}\partial_{xx} w = \partial_\tau u + \frac{\sigma^2}{2}\partial_x u - \frac{\sigma^2}{2}\partial_{xx} u = 0,$$

and $w(x,0) = (1 - \mathrm{e}^x)^+$. Finally note that changing $\tau \to \frac{\tau\sqrt{2}}{\sqrt{\sigma}}$ transforms the equation into the canonical heat equation. The Green's function of the one-dimensional heat equation is known,

$$\partial_\tau G - \partial_{xx} G = 0, \quad G(x,0) = \delta(x - x_0) \;\Rightarrow\; G(x,t) = \frac{1}{\sqrt{4\pi t}} \mathrm{e}^{-\frac{|x-x_0|^2}{4t}},$$

so $w = \int_{\mathbb{R}^+} G(x - x')(1 - \mathrm{e}^{x'})^+ \,\mathrm{d}x'$. Hence we have a closed-form solution for (3.3.4). This leads to the Black–Scholes formula for v. It can also be written in terms of a more C-friendly form with erf functions, as follows.

Black–Scholes formula for a vanilla put:

$$\begin{aligned} v_\sigma(y,\tau) = {} & \frac{y}{2}\left(1 + \operatorname{erf}\left(\frac{\ln y}{\sigma\sqrt{2\tau}} + \sigma\sqrt{\frac{\tau}{8}}\right)\right) \\ & - \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{\ln y}{\sigma\sqrt{2\tau}} - \sigma\sqrt{\frac{\tau}{8}}\right)\right) + 1 - y \\ & \text{and} \quad \operatorname{erf}(y) = \frac{2}{\sqrt{\pi}} \int_0^y \mathrm{e}^{-x^2} \,\mathrm{d}x. \end{aligned} \tag{3.3.10}$$

3.3.2. Existence results for basket options

The above results extend without difficulty to basket option. Using the moneyness prices $y_i = \mathrm{e}^{r\tau} S_i / K$, (3.2.14) becomes

$$\begin{aligned} & \partial_\tau u - \Xi' : \nabla\nabla u = 0, \quad u(y,0) = \phi'(y), \\ & \text{with} \quad \Xi'_{ij} = \frac{y_i y_j}{2} \sigma_i \sigma_j \rho_{ij}, \quad i,j = 1,\dots,d. \end{aligned} \tag{3.3.11}$$

The variational formulation is as follows: find

$$u \in V := \{v \in L^2(\mathbb{R}^{+d}) : y_i \partial_i u \in L^2(\mathbb{R}^{+d})\}$$

such that

$$\int_{\mathbb{R}^{+d}} \left[w \partial_\tau u + \sum_{ij} \partial_i (w \Xi'_{ij}) \partial_j u \right] = 0 \quad \forall w \in V. \tag{3.3.12}$$

Theorem 3.3.4. If $\{\sigma_i \sigma_j \rho_{ij}\}$ is a positive definite matrix and if there exists $\sigma_m, \sigma_M, C_\sigma$ with $0 < \sigma_m \le \sigma_i \le \sigma_M$, $|y_i \sigma_i| < C_\sigma$, for all $i, y_i > 0$, then (3.3.12) has exactly one solution.

The proof is established exactly as in the one-dimensional case.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

3.3.3. Existence result for an Asian option

Consider (3.2.16) and the following change of variable:

$$C(S,A,t) = Su(z,t)|_{z=\frac{K}{S}-\frac{t}{T}\frac{A}{S}}.$$

We have

$$\partial_A C = -\frac{t}{T}\partial_z u, \quad \partial_S C = u - \frac{1}{S}\left(K - \frac{At}{T}\right)\partial_z u, \quad \partial_t C = S\partial_t u - \frac{A}{T}\partial_z u,$$

$$\partial_{SS} C = \frac{1}{S^3}\left(K - A\frac{t}{T}\right)^2 \partial_{zz} u.$$

This leads to

$$\partial_t u + \frac{z^2\sigma^2}{2}\partial_{zz} u - \left(\frac{1}{T} + rz\right)\partial_z u = 0, \quad u(z,T) = z^+, \quad \forall z,t \in \mathbb{R}\times(0,T). \tag{3.3.13}$$

Appropriate behaviour for $z \gg 1$ is required: $u(z) \sim \mathrm{e}^{rt} z$.

Note that the drift term gives a contribution $b(u,u)$ to the bilinear form of the variational formulation, which is compatible with Gårding's inequality:

$$b(u,u) = \int_{\mathbb{R}} \left(\frac{1}{T} + rz\right)\partial_z u^2 = -\int_{\mathbb{R}} ru^2 \geq -r\|u\|_0^2.$$

Existence and uniqueness hold for this problem too. However, there is a technical difficulty due to the fact that the second-order term vanishes at $z = 0$, which is in the middle of the domain of z. Existence can be established by using the concept of viscosity solutions of the PDE (see Barles (1997)).

3.3.4. Analysis of the PDE for stochastic volatilities

It is shown in Achdou and Pironneau (2005, 2009) that the PDE for a European put P_t on S driven by the stochastic volatility model (1.3.1), (1.3.2) is

$$\begin{aligned}\partial_\tau u - \frac{f(y)^2 S^2}{2}\partial_{SS} u - \rho\beta S f(y)\partial_{Sy} u - \frac{\beta^2}{2}\partial_{yy} u - rS\partial_S u \\ + \left(\beta[\rho\frac{\mu - r}{f(y)} + \gamma\sqrt{1-\rho^2}] - \alpha(m-y)\right)\partial_y u + ru = 0,\end{aligned}$$

$$u(S,y,0) = (K-S)^+. \tag{3.3.14}$$

As usual, $\tau = T - t$ and $u(S,y,\tau) = P|_{T-\tau,S_0=S,Y_0=y}$. The return on volatility at risk γ is a function of S,y,τ, which is part of the model and which comes from the fact that the PDE derived by Itô calculus and hedging from (1.3.1), (1.3.2) is not unique.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

With $x = \ln y,\ v(x, y, \tau) = u(\mathrm{e}^x, y, \tau)$, the PDE becomes

$$\begin{aligned}
&\partial_\tau v - \frac{f(y)^2}{2}\partial_{xx} v - \rho\beta f(y)\partial_{xy} v - \frac{\beta^2}{2}\partial_{yy} v - (r - \frac{f(y)^2}{2})\partial_x v \\
&\qquad + \left(\beta[\rho\frac{\mu - r}{f(y)} + \gamma\sqrt{1-\rho^2}] - \alpha(m - y)\right)\partial_y v + rv = 0, \\
&v(x, y, 0) = (K - \mathrm{e}^x)^+. \qquad (3.3.15)
\end{aligned}$$

The characteristic polynomial $\pi(X, Y)$ of the second-order terms satisfies

$$2\pi(X, Y) = f(y)^2 X^2 + 2\rho\beta XY + \beta^2 Y^2.$$

The discriminant $D = \beta^2(\rho^2 - f^2)$ must be negative to secure a strongly parabolic PDE (3.3.15):

$$|\rho| < |f|. \qquad (3.3.16)$$

This condition is violated by the Cox–Ingersoll–Ross model because $f(y) = \sqrt{y}$; numerical difficulties are expected (change of type from elliptic to hyperbolic could cause discontinuous solutions and non-uniqueness). On the other hand, when the condition is satisfied, the problem can be studied in the variational space

$$V = \left\{w : w\sqrt{1 + y^2},\ \partial_y w,\ S|y|\partial_S w \in L^2(\mathbb{R}^+ \times \mathbb{R})\right\}.$$

Let $\Omega = \mathbb{R}^+ \times \mathbb{R}$. Find $u \in L^2(0, T; V)$ such that, for all $w \in V$,

$$\begin{aligned}
&\int_\Omega (w\partial_\tau u + ruw) + \int_\Omega \left[\frac{f(y)^2}{2}\partial_S(wS^2)\partial_S u + \partial_y\left(w\frac{\beta^2}{2}\right)\partial_y u\right] \qquad (3.3.17) \\
&\quad + \frac{1}{2}\int_\Omega \left[\beta f(y)\partial_S(w\rho S)\partial_y u + S\partial_y(w\rho\beta f(y))\partial_S u\right] \\
&\quad + \int_\Omega \left[-rS\partial_S u + \left(\beta[\rho\frac{\mu - r}{f(y)} + \gamma\sqrt{1-\rho^2}] - \alpha(m - y)\right)\partial_y u\right] w = 0.
\end{aligned}$$

3.3.5. Partial integro-differential variational equations

Consider a basket $S_{1t}, S_{2t}, \ldots, S_{dt}$ described by d SDEs with Lévy processes. Let $X_{it} = \ln S_{it}$ and let $g(X_t)$ be the pay-off of the option. By an argument close to that made for Theorem 3.2.5, we show that

$$\begin{aligned}
&\partial_\tau C + \Xi_{ij}\partial_{x_i x_j} C + r_i\partial_{x_i} C - rC \\
&\qquad + \int_{\mathbb{R}} [C(x + y, t) - C(x, t) - (\mathrm{e}^{y_i} - 1)\partial_{x_i} C(x, t)]\nu(\mathrm{d}y) = 0, \\
&C(x, T) = g(x), \qquad (3.3.18)
\end{aligned}$$

where r_i are the tendencies of the S_i (there may not be a risk-neutral d-dimensional measure that brings all r_i to the value r) and Ξ is computed from the volatilities σ_i and the correlation ρ_{ij} of the Brownian parts. For

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

instance, if σ_1, σ_2 are the volatilities of S_1 and S_2 and ρ is the correlation of the Brownian motions, then

$$\Xi = \begin{pmatrix} \sigma_1^2 & \frac{2\rho\sigma_1\sigma_2}{1+\rho^2} \\ \frac{2\rho\sigma_1\sigma_2}{1+\rho^2} & \sigma_2^2 \end{pmatrix}. \tag{3.3.19}$$

In variational form, for $u(x,\tau) = C(T-\tau, \mathrm{e}^x)$, the PDE reads as follows: for all $v \in V$, the set of square-integrable functions with square-integrable derivatives in $\mathbb{R}^d$, we have

$$\int_{\mathbb{R}^d} \big(v\partial_\tau u + \partial_{x_i}(\Xi_{ij} v)\partial_{x_j} u - r_i v \partial_{x_i} u + rvu\big) = \tag{3.3.20}$$
$$\int_{x,y\in\mathbb{R}^d} \big[(u(x+y)-u(x))v(x) - (\mathrm{e}^{y_i}-1)\partial_{x_i} u\ v(x)\big]\nu(\mathrm{d}y)\,\mathrm{d}x.$$

As mentioned before, this vectorial PIDE can be written as in (3.2.29), and when the integral is symmetric with respect to u and v, existence and uniqueness have been obtained by Farkas *et al.* (2006) (see also Tankov and Voltchkova (2009)).

PART FOUR
Numerical methods

4.1. A finite difference method for the Black–Scholes PDE

One of the oldest numerical methods to integrate a PDE is to replace the partial derivatives by finite differences on a grid: namely, given $x, t \to f(x,t)$ and a step size δx and a time step δt, f_i^m is the computed approximation of $f(j\delta x, m\delta t)$. For instance (3.3.3) for a vanilla put, written with the moneyness variable y and time to maturity τ, is approximated on the lattice $y_j = j\delta y$ by

$$\frac{P_j^{m+1} - P_j^m}{\delta\tau} - \frac{y_j^2 \sigma_j^{m+\frac12\,2}}{2\delta y^2}\big(P_{j+1}^{m+\frac12} - 2P_j^{m+\frac12} + P_{j-1}^{m+\frac12}\big) = 0, \quad j = 1,\dots,J-1,$$
$$P_j^0 = (1-y_j)^+, \quad j = 0,\dots,J,$$
$$P_0^m = 1, \quad P_J^m = 0, \quad m > 0, \tag{4.1.1}$$

with the convention that $f^{m+\frac12} := \frac12(f^{m+1} + f^m)$. This is certainly the best among the easily implemented schemes because at each stage m it involves a tridiagonal system, which is very fast to solve by Gauss factorization.

When σ is not time-dependent, stability of this Crank–Nicolson scheme is easy to establish.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Proposition 4.1.1. When σ does not depend on t, scheme (4.1.1) is unconditionally stable and second-order in space and time.

Proof. The proof requires us to replace the non-homogeneous boundary conditions by zero at the cost of adding a non-zero right-hand side. So let $w(y) = a(y-b)(J\delta y - y)^2$ when $y > \frac{1}{2}$ and $w(y) = 1 - y$ otherwise, with a, b such that $w(\frac{1}{2}) = \frac{1}{2}$ and $w'(\frac{1}{2}) = -1$. The scheme for $u = P - w$ is

$$\begin{aligned} &\frac{u_j^{m+1} - u_j^m}{\delta\tau} - \frac{y_j^2 {\sigma_j}^2}{2\delta y^2}\big(u_{j+1}^{m+\frac{1}{2}} - 2u_j^{m+\frac{1}{2}} + u_{j-1}^{m+\frac{1}{2}}\big) = y_j^2\sigma_j f_j, \quad j = 1, \dots, J-1, \\ &\qquad u_j^0 = (1-y_j)^+ - w(y_j), \quad j = 0, \dots, J, \\ &\qquad u_0^m = 0, \quad u_J^m = 0, \quad m > 0, \end{aligned} \tag{4.1.2}$$

where $f_j = \frac{1}{2}(w_{j+1} - 2w_j + w_{j-1})/\delta y^2$ is the centred finite difference approximation of $\frac{1}{2}w''$. For what follows it is important to notice that $u_j^0 = 0$ when $j\delta y < \frac{1}{2}$.

Let (4.1.2) be multiplied by $u_j^{m+\frac{1}{2}}\delta\tau/(y_j^2\sigma_j^2)$ and summed in j:

$$\begin{aligned} &\sum_{j=1}^{J-1} \frac{u_j^{m+1\,2} - u_j^{m\,2}}{y_j^2\sigma_j^2} \\ &\quad = \frac{\delta\tau}{2\delta y^2}\sum_{j=1}^{J-1} u_j^{m+\frac{1}{2}}(u_{j+1}^{m+\frac{1}{2}} - 2u_j^{m+\frac{1}{2}} + u_{j-1}^{m+\frac{1}{2}}) + \sum_{j=1}^{J-1}\delta\tau f_j u_{j+\frac{1}{2}}^m \\ &\quad = -\frac{\delta\tau}{2\delta y^2}\left(\sum_{j=0}^{J-2}(u_{j+1}^{m+\frac{1}{2}} - u_j^{m+\frac{1}{2}})^2 + u_{J-1}^{m+\frac{1}{2}\,2} + (u_1^{m+\frac{1}{2}} - u_0^{m+\frac{1}{2}})u_0^{m+\frac{1}{2}}\right) \\ &\qquad + \delta\tau\sum_{j=1}^{J-1} f_j u_j^{m+\frac{1}{2}} \\ &\quad \le \delta\tau\sum_{j=1}^{J-1} f_j u_j^{m+\frac{1}{2}} \le \delta\tau\left(\sum_{j=1}^{J-1} y_j^2\sigma_j^2 {f_j}^2\right)^{\frac{1}{2}}\left(\sum_{j=1}^{J-1}\frac{u_j^{m+\frac{1}{2}\,2}}{y_j^2\sigma_j^2}\right)^{\frac{1}{2}}. \end{aligned} \tag{4.1.3}$$

This is an inequality of the type $A^{m+1} - A^m \le \delta\tau F\sqrt{A^{m+\frac{1}{2}}}$, which leads to the boundedness of A^M because

$$A^{m+1} - A^m = (\sqrt{A^{m+1}} + \sqrt{A^m})(\sqrt{A^{m+1}} - \sqrt{A^m})$$
$$\text{and} \quad \sqrt{A^{m+1}} + \sqrt{A^m} \ge \sqrt{2}\sqrt{A^{m+\frac{1}{2}}}.$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

So finally, after a summation in m,

$$\sum_{j=1}^{J-1} \frac{u_j^{M\,2}}{y_j^2\sigma_j^2} \le \sum_{j>\frac{1}{2\delta y}}^{J-1} \frac{u_j^{0\,2}}{y_j^2\sigma_j^2} + 2\left[\sum_{m=1}^{M-1} \delta\tau \left(\sum_{j=1}^{J-1} y_j^2\sigma_j^2 f_j^2\right)^{\frac{1}{2}}\right]^2.$$

The scheme is shown to be second-order accurate by a Taylor expansion in time and space of $u(j\delta y, m\delta\tau)$ in the usual way (Richtmyer and Morton 1967). □

Remark 4.1.2. In a different context, Rannacher (1984) showed that for problems with non-smooth initial conditions (which is the case here since the second derivative of u^0 is a Dirac mass), the Crank–Nicolson scheme is not second-order and may have oscillations near $\tau = 0$. It can be cured by starting with four smaller time steps of implicit Euler. More precisely, to integrate $\dot{u} + Au = 0$, one should use

$$\left(I + \frac{\delta\tau}{2}A\right)u^{m+\frac{1}{2}} = u^m, \qquad m = 0, \frac{1}{2}, 1, \frac{3}{2} \quad \text{(Euler)}, \tag{4.1.4}$$

$$\left(I + \frac{\delta\tau}{2}A\right)u^{m+1} = \left(I - \frac{\delta\tau}{2}A\right)u^m, \quad m = 2, 3, \ldots \quad \text{(Crank–Nicolson)}.$$

4.1.1. Alternative directions for basket options

Consider a basket option, with pay-off $\phi = (K - S_{1T} - S_{2T})^+$, on S_1, S_2. By a change to the moneyness variables $y = y_1, y_2$ and time to maturity, the PDE (3.2.14) is written in $\mathbb{R}^+ \times \mathbb{R}^+ \times (0, T)$ as

$$\partial_\tau u - \Xi : \nabla\nabla u = 0, \quad u(y, 0) = \phi(y). \tag{4.1.5}$$

Assuming $\Xi_{12} = \Xi_{21}$, Crank–Nicolson centred differences would lead to

$$\begin{aligned}
&\frac{u_{ij}^{m+1} - u_{ij}^m}{\delta\tau} - \frac{\Xi_{11ij}^{m+\frac{1}{2}}}{\delta y_1^2}\left(u_{i+1\,j}^{m+\frac{1}{2}} - 2u_{ij}^{m+\frac{1}{2}} + u_{i-1\,j}^{m+\frac{1}{2}}\right) \\
&\quad - \frac{\Xi_{12ij}^{m+\frac{1}{2}}}{2\delta y_1\delta y_2}\left(u_{i+1\,j+1}^{m+\frac{1}{2}} - u_{i+1\,j-1}^{m+\frac{1}{2}} - u_{i-1\,j+1}^{m+\frac{1}{2}} + u_{i-1\,j-1}^{m+\frac{1}{2}}\right) \\
&\quad - \frac{\Xi_{22ij}^{m+\frac{1}{2}}}{\delta y_2^2}\left(u_{i\,j+1}^{m+\frac{1}{2}} - 2u_{ij}^{m+\frac{1}{2}} + u_{i\,j+1}^{m+\frac{1}{2}}\right) = 0.
\end{aligned} \tag{4.1.6}$$

The linear system generated by this scheme is pentadiagonal, and so an efficient numerical solution requires the use of a linear algebra library and/or writing a complex computer program. An alternative is to use a fractional step method.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Following Ikonen and Toivanen (2009), if the cross-derivative is approximated by

$$(\partial_{y_1y_2}u)_{ij} \approx \frac{1}{2\delta y_1 \delta y_2}\left[2u_{ij} - u_{i-1\ j+1} - u_{i+1\ j-1} - \sum_{k=1,2} \delta y_k^2 (\partial_{y_k y_k} u)_{ij}\right] \tag{4.1.7}$$

and centred finite differences for $\partial_{y_k y_k} u$, then, when $\Xi_{12} \leq 0$, the matrices of the linear systems are M-matrices, and the scheme is second-order and unconditionally stable. The scheme is not superior to (4.1.6) but it is appropriate to AD (alternate directions).

With an implicit Euler time scheme, (4.1.7) leads to

$$\frac{u^{m+1} - u^m}{\delta\tau} + A_1 u^{m+1} + A_c u^{m+1} + A_2 u^{m+1} = 0, \tag{4.1.8}$$

where A_i contains the approximation of $\partial_{y_i y_i} u$ and A_c the rest of the cross-derivative $\partial_{y_1 y_2} u$.

AD is a fractional time step method (where $m + \frac{1}{3}$ is no longer the value at $(m + \frac{1}{3})\delta t$):

$$\begin{aligned}
\frac{u^{m+\frac{1}{3}} - u^m}{\delta\tau} + A_1 u^{m+\frac{1}{3}} + A_c u^m + A_2 u^m &= 0,\\
\frac{u^{m+\frac{2}{3}} - u^m}{\delta\tau} + A_1 u^{m+\frac{1}{3}} + A_c u^{m+\frac{2}{3}} + A_2 u^{m+\frac{1}{3}} &= 0,\\
\frac{u^{m+1} - u^m}{\delta\tau} + A_1 u^{m+\frac{2}{3}} + A_c u^{m+\frac{2}{3}} + A_2 u^{m+1} &= 0.
\end{aligned} \tag{4.1.9}$$

Each equation is, as it were, one-dimensional, so yields a tridiagonal linear system. The first one involves a horizontal stencil, the second one a diagonal stencil and the third a vertical stencil.

The same idea can used with Crank–Nicolson time discretization. Strang symmetrization can be added to obtain a fully second-order scheme.

4.1.2. Numerical method for PIDEs

The finite difference method applies equally well to PIDEs such as (3.2.21), but to avoid solving a full linear system it is wise to use a semi-implicit time scheme whereby the integro-differential term is discretized explicitly.

For instance, in the case of a finite difference scheme for (3.2.21), at each time step the right-hand side of the linear system is $P^{m+1}\delta t + v^{m+1}\delta t$, and

$$\begin{aligned}
v &= \int_{\mathbb{R}} (P(S\mathrm{e}^y) - P(S) - S(\mathrm{e}^y - 1)\partial_S P(S))\nu(\mathrm{d}y)\\
&\approx \sum_{j=0}^{N} \nu\left(\log\frac{S_j}{S_i}\right)[P_j - P_i - (P_{i+1} - P_i)]\frac{S_j - S_i}{(j+\frac{1}{2})\delta x}.
\end{aligned} \tag{4.1.10}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Furthermore, the sums can be restricted to $|i-j| < M$, M large, when $\nu(y)$ decays exponentially with y.

Theorem 4.1.3. The scheme is stable if $\delta t \le C\frac{\delta x}{\lambda+\delta x}$.

For a proof the reader is sent to Cont and Voltchkova (2005).

4.2. Spectral methods

As shown by Jackson, Jaimungal and Surkov (2008), Fourier transform of the PIDE is very powerful for pricing options on assets modelled with a jump process when σ is constant. Consider (3.2.22) written with $x = \ln S$, $\tau = T - t$ and $v(x,\tau) = \mathrm{e}^{r\tau}C(\mathrm{e}^x, T-\tau)$:

$$\begin{aligned}&\partial_\tau v - r\partial_x v - \frac{\sigma^2}{2}\partial_{xx}v\\ &\qquad - \int_{\mathbb{R}}\big[v(x+y,\tau) - v(x,\tau) - y\mathbf{1}_{|y|<1}\partial_x v(x,\tau)\big]\nu(\mathrm{d}y) = 0,\\ &v(x,0) = v^0 := (\mathrm{e}^x - K)^+. \end{aligned} \tag{4.2.1}$$

Let

$$\mathcal{F}(u) = \int_{\mathbb{R}} \mathrm{e}^{-\mathrm{i}\omega x}u(x)\,\mathrm{d}x \quad \text{and} \quad \mathcal{F}^{-1}(\hat{u}) = \frac{1}{2\pi}\int_{\mathbb{R}} \mathrm{e}^{\mathrm{i}\omega x}\hat{u}(\omega)\,\mathrm{d}\omega.$$

Applying the operator $\mathcal{F}$ to (4.2.1) gives

$$\partial_\tau \hat{v} - \Psi\hat{v} = 0 \text{ in } \mathbb{R}, \quad \hat{v}(\omega,0) = \mathcal{F}(\mathrm{e}^x - K)^+,$$

where Ψ is

$$\Psi(\omega) = \mathrm{i}r\omega - \frac{\sigma^2}{2}\omega^2 + \int_{\mathbb{R}}(\mathrm{e}^{\mathrm{i}\omega y} - 1 - \mathrm{i}y\omega\mathbf{1}_{|y|<1})\nu(\mathrm{d}y).$$

So the solution is

$$v^T(x) = \mathcal{F}^{-1}[\{\mathcal{F}v^0\}(\omega)\mathrm{e}^{\Psi(\omega)T}].$$

It is numerically approximated by using discrete fast Fourier transforms, so the solution is computed with only two of those.

The problem must be localized into (x_m, x_M) and a grid is chosen with $\delta x = \frac{1}{N}(x_M - x_m)$, $x_k = x_m + k\delta x$. Similarly a mesh in the frequency domain is chosen, but to avoid aliasing errors it is best to use the Nyquist critical frequency so that $\omega_M(x_M - x_m) = \frac{N}{2}$, $\delta\omega = \frac{2}{N}\omega_M$, $\omega_j = j\delta\omega$. Then $\mathcal{F}v$ is approximated by

$$\hat{v}_j = \mathcal{F}v(\omega_j) \approx \sum_{k=0}^{N-1} \mathrm{e}^{-\mathrm{i}\omega_j x_k}v_k\delta x = \mathrm{e}^{\mathrm{i}\omega_j x_m}\mathtt{FFT}(v), \quad v_k = v(x_k), \tag{4.2.2}$$

where $\mathtt{FFT}(v)$ is a numerical fast Fourier transform algorithm. Finally

$$v^T = \mathtt{FFT}^{-1}\big(\mathrm{e}^{\Psi T}\mathtt{FFT}(v^0)\big).$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

4.3. Finite element methods

Finite difference methods have one limitation: they are not easy to adapt to arbitrary meshes. Yet for a given precision, the gain in computing time obtained by using a mesh adapted to the problem is tremendous. While it is not impossible to use arbitrary meshes with finite difference methods, the formulae are rapidly inextricable, so it is better to use a finite element method.

Let us consider a system of equations such as (3.3.14) for stochastic volatility models, which we write symbolically as

$$\begin{aligned}
&\partial_\tau u - \Xi : \nabla\nabla u + b \cdot \nabla u + ru = 0,\\
&u(x,\tau) = u_\Gamma \quad \forall x \in \Gamma_1, \quad n^T \Xi \nabla u(x,t) = 0 \quad \forall x \in \partial\Omega \backslash \Gamma_1,\\
&u(x,0) = \phi(x) \quad \forall x \in \Omega, \quad \forall \tau \in (0,T).
\end{aligned} \tag{4.3.1}$$

For (3.3.14),

$$\Xi_{11} = f(x_2)x_1^2, \quad \Xi_{12} = \Xi_{21} = \frac{1}{2}\rho\beta x_1 f(x_2), \quad \Xi_{22} = \beta^2/2,$$

$$b_1 = rx_1, \quad b_2 = \beta\left[\rho\frac{\mu - r}{f(x_2)} + \gamma\sqrt{1-\rho^2}\right] - \alpha(m - x_2).$$

The domain Ω is a d-dimensional Cartesian product of $\mathbb{R}$ or $\mathbb{R}^+$. In the case of (3.3.14), $d = 2$ and $\Omega = \mathbb{R}^+ \times \mathbb{R}$.

Note that this formalism contains a wide variety of PDEs in finance. For example, a plain vanilla European put with local volatility corresponds to

$$d = 1, \quad \Omega = \mathbb{R}^+, \quad \Xi = \sigma^2(x,\tau)x^2/2, \quad b = -rx, \quad \phi(x) = (K - x)^+.$$

The variational formulation of (4.3.1) consists in seeking $u - \tilde{u}_\Gamma \in V_0$ such that

$$\begin{aligned}
&\int_\Omega \left[w\partial_\tau u + \nabla \cdot (w\Xi\nabla u) + wb \cdot \nabla u + rwu\right] \mathrm{d}x = 0 \quad \forall w \in V_0,\\
&\qquad \text{where} \quad u|_{\tau=0} = \phi.
\end{aligned} \tag{4.3.2}$$

As before, V is a space of square-integrable functions with weighted square-integrable derivatives. In the case of (3.3.14),

$$V = \left\{v : v\sqrt{1 + f(y)^2}, \frac{\partial v}{\partial y}, x|f(y)|\frac{\partial v}{\partial x} \in L^2(\Omega \times (0,T))\right\}, \tag{4.3.3}$$

where V_0 is the subspace of V of functions which are zero on Γ_1. Finally $\tilde{u}_\Gamma$ is an extension of u_Γ in V. We write (4.3.2) formally as follows: find

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

$u - \tilde{u}_\Gamma \in V_0$ such that $u|_{\tau=0} = \phi$ and

$$\frac{\mathrm{d}}{\mathrm{d}\tau}(w,u) + a_\tau(w,u) = 0 \quad \forall w \in V_0. \tag{4.3.4}$$

It goes without saying that

$$(w,u) = \int_\Omega wu\,\mathrm{d}x, \quad a_\tau(w,u) = \int_\Omega [\nabla\cdot(w\Xi\nabla u) + wb\cdot\nabla u + rwu]\,\mathrm{d}x.$$

4.3.1. Galerkin method

The Galerkin method obtains a discrete version of (4.3.4) by replacing V with a finite-dimensional space $V_h \subset V$. The following is needed for V_h:

- given $\epsilon > 0$ and $v \in V$, there exists $v_h \in V_h$ such that $\|v_h - v\| \le \epsilon$,
- accumulation points of sequences $v_h \in V_h,\ h \to 0$ belong to V.

So consider the problem of finding $u_h \in V_h$ with $u_h - u_{\Gamma h} \in V_{0h}$ and

$$\frac{\mathrm{d}}{\mathrm{d}\tau}(w_h,u_h) + a_\tau(w_h,u_h) = 0. \tag{4.3.5}$$

If $\{w^i\}_1^N$ is a basis of V_h, then (4.3.5) is equivalent to

$$u_h(x,\tau) = u_{\Gamma h}(x) + \sum_1^N u_j(\tau)w^j(x),$$

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\left(w^i, \sum_1^N u_j w^j\right) + a_\tau\left(w^i, \sum_1^N u_j w^j\right) = 0, \quad i = 1,\dots,N, \tag{4.3.6}$$

which is a system of differential equations

$$M\dot{U} + A(\tau)U = 0, \tag{4.3.7}$$

with $M_{ij} = (w^j, w^i),\ A_{ij}(\tau) = a_\tau(w^j, w^i)$. A discrete time-stepping scheme still has to be applied to (4.3.7), for instance the Crank–Nicolson scheme

$$M\frac{U^{m+1} - U^m}{\delta t_m} + A^{m+\frac{1}{2}}U^{m+\frac{1}{2}} = 0, \tag{4.3.8}$$

where $A^m = A(t_m)$ and $A^{m+\frac{1}{2}} = A(\frac{1}{2}(\tau_m + \tau_{m+1}))$.

4.3.2. Finite elements in 1D

Consider a plain vanilla put with local volatility and 'localize' the problem by approximating $\mathbb{R}^+$ by $\Omega := (0,L)$, L large. Then take

$$V_0 = \{v \in L^2(0,1) : x\partial_x v \in L^2(0,1)\}, \quad V_0 = \{v \in V : v(L) = 0\}.$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The easiest way is to choose a mesh $\cup_1^I [x_{i-1}, x_i] = [0, L]$ and take V_h to be the space of piecewise linear functions,

$$V_{0h} = \{v_h \text{ continuous, linear on each subinterval } [x_{i-1}, x_i],\ v_h(L) = 0\}.$$

The canonical basis for V_{0h} is the set of $\{w^i\}_0^{I-1}$, uniquely defined by

$$w^i(x_j) = \delta_{ij}, \quad w^i \in V_{0h}, \quad i, j = 0, \ldots, I-1.$$

With this basis, called the nodal basis (or the hat functions), the integrals M_{ij} and A_{ij} can be computed exactly, or up to quadrature error due to σ if not piecewise polynomial on the intervals $[x_{i-1}, x_i]$ (see Achdou and Pironneau (2005)).

It is easy to show that the matrices A and M are tridiagonal, so a resolution of the linear system (4.3.8) at each time step is best done with an LU factorization. In the end the computational complexity is the same as for a finite difference method. In fact both methods are identical when the mesh is uniform.

Convergence in the V-norm can be proved to be of the order $h^2 + \delta t^2$ in the L^2-norm and of order $h + \delta t^2$ in the V-norm

Most interesting is the following *a posteriori* estimate.

Proposition 4.3.1. (Achdou and Pironneau 2005) We have the following *a posteriori* estimate:

$$[[u - u_{h,\delta t}]](t_n) \le c(u_0)\delta t + c\frac{\mu}{\sigma_m^2}$$
$$\times \left(\sum_{m=1}^{n} {\eta_m}^2 + (1+\rho)^2 \max(2, 1+\rho) \sum_{m=1}^{n} \frac{\delta t_m}{\sigma_m^2} \prod_{i=1}^{m-1} (1 - 2\lambda\delta t_i) \sum_{\omega \in \mathcal{T}_{mh}} {\eta_{m,\omega}}^2 \right)^{\frac{1}{2}}$$

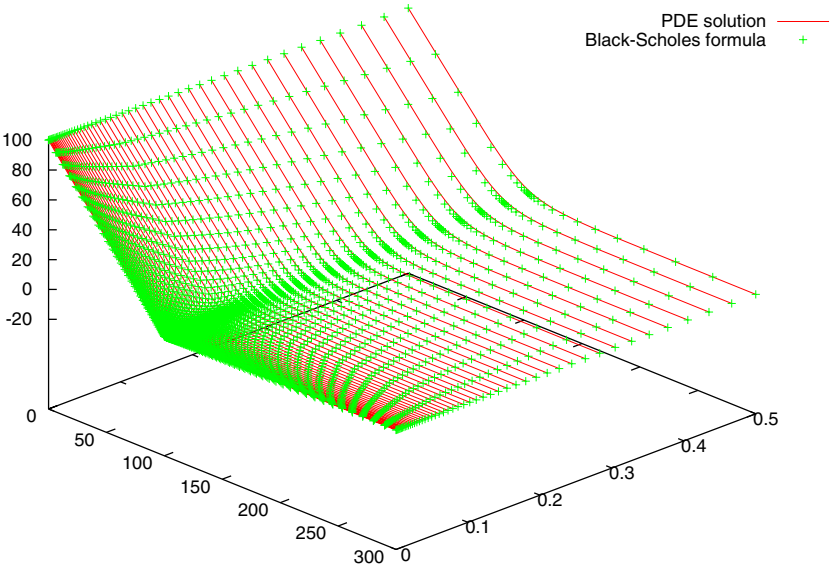
where μ is the continuity constant of a_t, $\rho = \max_{2 \le n \le N} \frac{\delta t_n}{\delta t_{n-1}}$, $\mathcal{T}_{mh}$ is the partition of $(0, L)$ in small intervals ω at time t_m, and

$$\eta_m^2 = \delta t_m \mathrm{e}^{-2\lambda t_{m-1}} \frac{\sigma_m^2}{2} |u_h^m - u_h^{m-1}|_V^2,$$
$$\eta_{m,\omega} = \frac{h_\omega}{x_{\max}(\omega)} \left\| \frac{u_h^m - u_h^{m-1}}{\delta t_m} - rx\frac{\partial u_h^m}{\partial x} + r u_h^m \right\|_{L^2(\omega)},$$

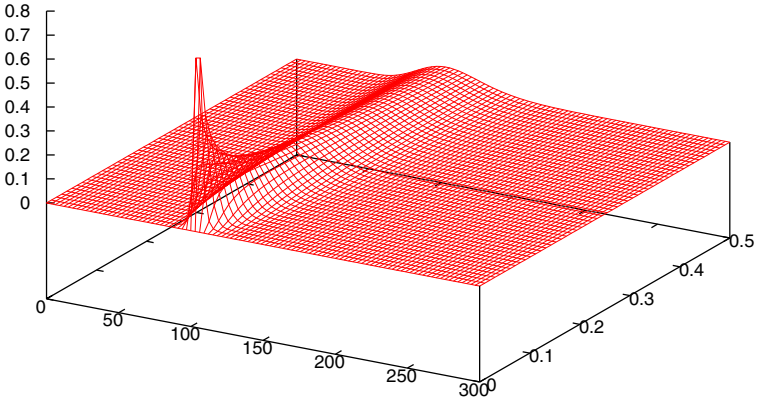
and h_ω is the size of ω.

Note that everything is known after the computation of u_h. The mesh can then be adapted by using the error indicator η_m to adjust δt and $\eta_{m,\omega}$ to adjust h. Figure 4.3.1 shows the computed *a posteriori* error and the actual error for a vanilla put with constant volatility and comparison with the Black–Scholes analytic formula.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press



(a)



(b)

Figure 4.3.1. (a) The computed solution on an adapted mesh (solid lines) and one computed with the Black–Scholes formula (crosses). Graph (b) shows $\eta_{n,\omega}$ as a function of x and t and indicates where the adapted mesh needs to be refined further in x. The parameters are $K = 100$, $T = 0.5$, $r = 0$, $\sigma = 0.3$, 50 time steps and 100 mesh points.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

4.4. Variational methods for American options

Referring to (3.2.31), an American put on S can be valued by solving

$$\partial_t P + \frac{\sigma^2 S^2}{2}\partial_{SS} P + rS\partial_S P - rP \le 0,$$
$$P - (K-S)^+ \ge 0 \quad \forall S \in \mathbb{R}^+, \quad t \in (0,T), \tag{4.4.1}$$

with, at all (S,t), at least one of the two inequalities satisfied as an equation. Let $(u,v) := \int_0^\infty uv\,\mathrm{d}S$ and $a, b : V \times V \to R$ be the two bilinear forms

$$a(u,v) := \int_{\mathbb{R}^+} \left(\frac{\sigma^2 S^2}{2}\partial_S u \partial_S v + rvu \right),$$
$$b(u,v) := \int_{\mathbb{R}^+} \left(\partial_S \left(\frac{\sigma^2 S^2}{2} \right) - rS \right) v \partial_S u, \tag{4.4.2}$$

where V is the Sobolev space

$$V = \{ v \in L^2(\mathbb{R}^+) : S\partial_S v \in L^2(\mathbb{R}^+) \},$$

which has already been used for European options. The problem becomes: find $u \in L^2(0,T;V)$ with $u(S,T) = \phi(S) := (K-S)^+$ and a.e. in $(0,T)$, $u \ge \phi$ and

$$(\partial_t u, v - \phi) - a(u, v-\phi) - b(u, v-\phi) \le 0,$$
$$\forall v \in V^\phi = \{ v \in V : v \ge \phi \}. \tag{4.4.3}$$

Let us call

$$V^+ := \{ v \in V : v(S) \ge 0, \text{ a.e. in } \mathbb{R}^+ \}.$$

Let $w = v - \phi$, $\tau = T - t$, to prove the following.

Proposition 4.4.1. The variational inequality is as follows: find $u \in L^2(0,T;V)$ with $u(S,T) = \phi(S) := (K-S)^+$, and a.e. in $(0,T)$

$$u - \phi \in V^+, \quad \partial_\tau (u,w) + a(u,w) + b(u,w) \ge 0 \quad \forall w \in V^+ \tag{4.4.4}$$

has exactly one solution. Further, for regular σ, u is a solution of (4.4.1).

Proof. The complete proof can be found in Achdou and Pironneau (2005). Let us give only a simple argument. Let (4.4.3) be discretized in time by

$$\left(\frac{u^{m+1} - u^m}{\delta t}, v - \phi \right) - a(u^m, v-\phi) - b(u^{m+1}, v-\phi) \le 0 \quad \forall v \in V^\phi \tag{4.4.5}$$

with $M\delta t = T$. As u^M is known, (4.4.5) is solved backwards in time. With

$$A(u,v) = \left(\frac{u}{\delta t}, v \right) + a(u,v) \quad \text{and} \quad L(v) := \left(\frac{u^{m+1}}{\delta t}, v \right) - b(u^{m+1}, v),$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

the problem becomes: find $u^m \in V^\phi$ such that

$$A(u^m, v-\phi) \geq L(v-\phi) \quad \forall v \in V^\phi. \tag{4.4.6}$$

Let us show that it is equivalent to u^m being the solution of

$$\min_{u\in V^\phi} \frac{1}{2}A(u,u) - L(u). \tag{4.4.7}$$

The minimum exists and is unique because the functional is strictly convex and V^ϕ is a closed cone in V. By definition, if u^m is the solution,

$$\frac{1}{2}A(u^m+w,u^m+w)-L(u^m+w) \geq \frac{1}{2}A(u^m,u^m)-L(u^m)\ \forall w:\ u^m+w\in V^\phi.$$

Take $w=\lambda v$, $\lambda>0$ and let $\lambda\to 0$. At the limit

$$A(u^m,v)\geq L(v) \quad \forall v\in V, \quad \text{such that } \lim_{\lambda\to 0} u^m+\lambda v \geq \phi.$$

If u^m is smooth, by integration by parts, this is also

$$\int_\Omega \left(\frac{u^m-u^{m+1}}{\delta t}+ru^m-\partial_S\left(\frac{\sigma^2S^2}{2}\partial_S u^m\right)-\left(rS-\partial_S\frac{\sigma^2S^2}{2}\right)\partial_S u^{m+1}\right)v\geq 0,$$

which implies that the integrand is positive everywhere. Consider an S where $u^m(S)>\phi(S)$. Then, for any $v\in V$, there exists an interval (S_1,S_2) containing S where $\lim_{\lambda\to 0} u^m+\lambda v \geq \phi$ holds if the support of v is in (S_1,S_2). This in turn implies that at this S,t either $u^m(S)=\phi(S)$ or

$$\frac{u^m-u^{m+1}}{\delta t}+ru^m-\partial_S\left(\frac{\sigma^2S^2}{2}\partial_S u^m\right)-\left(rS-\partial_S\frac{\sigma^2S^2}{2}\right)\partial_S u^{m+1}=0.$$

To complete the proof one must show that the finite difference scheme converges when $\delta t\to 0$. Note that there will be a stability condition because the first-order terms have been discretized explicitly in time (*i.e.*, at t^{m+1} instead of t^m or $t^{m+\frac{1}{2}}$). □

4.4.1. Discretization by FEM

Variational methods lead naturally to Galerkin and finite element methods. First the problem is localized: $\mathbb{R}^+ \approx (0,L)$ with L large, typically 3 times K. Then V is approximated by the piecewise linear continuous finite element space

$$V_{0h}=\{v_h\in C^0(0,L): v_h|_{T_k}\in P^1, \quad v_h(L)=0\}, \tag{4.4.8}$$

and V^ϕ by

$$V^\phi_{0h}=\{v_h\in V_{0h}: v_h\geq \phi_h\}, \tag{4.4.9}$$

where ϕ_h is the continuous piecewise linear interpolate of ϕ on the triangulation, uniquely defined by

$$\phi_h\in V_{0h}, \quad \phi_h(q^j)=\phi(q^j), \quad \text{for all vertices } q^j \text{ of the triangulation.}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The discretized problem is as follows: find $\{u_i^m\}_{i=1}^I$ such that

$$\left(\frac{u_h^{m+1}-u_h^m}{\delta t}, v_h-\phi_h\right) - a(u_h^m, v_h-\phi_h) - b(u_h^m, v_h-\phi_h) \le 0 \quad \forall v_h \in V_{0h}^{\phi},$$

$$\text{with } u_h^m(x) = \sum_{i=1}^{I} u_i^m w^i(x). \tag{4.4.10}$$

Note that $w^i(x) \ge 0\ \forall x$, so

$$v_h \ge \phi_h \Leftrightarrow v_j \ge \phi_j \quad \text{for all vertices } q^j.$$

Thus, and because $w^j(x) \ge 0$ for all j, (4.4.10) is also

$$\left(\frac{u_h^{m+1}-u_h^m}{\delta t} w^j\right) - a(u_h^m, w^j) - b(u_h^m, w^j) \le 0, \quad j=1,\dots,I,$$

which we write as

$$\tilde{a}(u_h, w^j) \ge (f, w^j), \quad j=1,\dots,I, \tag{4.4.11}$$

$$u_h(x) = \sum_{1}^{I} u_i w^i(x) \ge \phi_h(x),$$

$$\text{where } \tilde{a}(u,v) = \frac{1}{\delta t}(u,v) + a(,v) + b(u,v), \quad f = \frac{u_h^{m+1}}{\delta t}. \tag{4.4.12}$$

Ito and Kunisch (2006) suggested reformulating (4.4.11) as follows.

Proposition 4.4.2. For any given constant $c > 0$, problem (4.4.11) is equivalent to

$$\begin{aligned} &\tilde{a}(u_h, w_h) + (\lambda, w_h) = (f, w_h), \quad \forall w_h \in V_{0h}, \\ &\lambda(x) - \min\{0, \lambda(x) + c(u_h(x) - \phi_h(x))\} = 0, \quad \forall x. \end{aligned} \tag{4.4.13}$$

Proof. The last equality is equivalent to

$$\lambda \le 0, \quad \lambda \le \lambda + c(u_h - \phi_h),$$
$$i.e., \quad u_h \ge \phi_h, \quad \lambda \le 0, \quad \lambda(x)(u_h(x) - \phi(x)) = 0, \quad \forall x.$$

The last statement means that for each x one of the two inequalities at least one is an equality. The rest of the proof is a classical argument using Lagrange multipliers:

$$\begin{aligned} &\tilde{a}(u_h, w_h) + (\lambda, w_h) = (f, w_h), \quad \lambda(x) \le 0, \quad w_h(x) \ge 0 \\ &\text{implies } \tilde{a}(u_h, w_h) \ge (f, w_h), \quad \forall w_h \ge 0. \end{aligned} \tag{4.4.14}$$

Recall that an equation such as $F(x) = 0$ (with $F : \mathbb{R}^n \to \mathbb{R}^n$) can be solved by Newton's method:

$$x_{k+1} = x_k - G(x_k)^{-1} F(x_k),$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

with $G = F'$ the Jacobian of F. Hintermüller, Kovtunenko and Kunisch (2004) observed that Newton's algorithm converges even if F is not differentiable, provided that there exists G such that

$$\text{for all } x, \quad \lim_{\|h\|\to 0} \|F(x+h) - F(x) - G(x+h)h\| = 0,$$

a property which is satisfied by $F(x) = \max\{0, x\}$, for instance with $G(x) = \max\{0, x\}/x$.

4.4.2. Semi-smooth Newton algorithm

Let $A_{ij} := \tilde{a}(w^i, w^j)$. Newton's algorithm applied to (4.4.13) gives the following.

(1) Choose $c > 0$, u_0, $\lambda_0 \le 0$, set $k = 0$.

(2) Determine

$$A_k := \{x : \lambda_k(x) + c(u_k(x) - \phi(x)) < 0\}.$$

(3) Set

$$u_{k+1} = \arg \min_{u \in H^1(\mathbb{R}^+)} \left\{ \frac{1}{2} a(u,u) - (f,u) : u = \phi \text{ on } A_k \right\}.$$

(4) Set

$$\lambda_{k+1} = f - Au_{k+1}.$$

Set $k = k + 1$ and go to (2).

The beauty of the method is that its implementation is almost painless from an implicit finite element or finite difference solver with the following. To compute the solution v_h of

$$\bar{a}(v_h, w^j) = (\bar{f}, w^j) \quad \forall j \notin K, \quad v_h(q^j) = \phi(q^j), \quad j \in K, \quad j = 1, \dots, I,$$

- Compute $A_{ij} = \tilde{a}(w^i i, w^j)$ and $f_j = (f, w^j)$, for all i, j.
- Reset $A_{ii} = 10^{30}$, $f_i = 10^{30}\phi(q^j)$, for all $i \in K$.
- Solve $Av = f$.

Although it looks as if the modification of A would ruin its condition number, in reality it does not, because it is essentially equivalent to the elimination of the rows and columns of indices in K. It would be equivalent to row and column elimination if, instead of 10^{30}, the largest integer number that the computer can store is used so that the addition of $\sum_{j\neq i} A_{ij}v_j$ to $A_{ii}v_i$ does not affect the significant digits.

Theorem 4.4.3. Algorithm 4.4.2 converges locally super-linearly provided A is an M-matrix.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The proof can be found in Hintermüller *et al.* (2004). When $b = 0$, A is an M-matrix if mass-lumping (a reduced integration formula with Gauss points at the vertices) is applied to the part of A which comes from $(w^i, w^j)/\delta t$, and provided the triangulation has no angle greater than $\pi/2$. If this is not the case, the method still converges provided u_0, λ_0 is not too far from the solution. Hintermüller *et al.* (2004) also showed that the semi-smooth Newton method converges if applied to the continuous problem, before discretization, but the precision is proportional to c^{-1}, so c must be large.

4.5. A stochastic volatility model solved by FEM

We now return to (4.3.1) for the stochastic volatility model (3.3.14). When $f(0) = 0$, the problem is complicated by the fact that the PDE degenerates at $y = 0$. In the Stein–Stein case $f(y) = |y|$, at least, it shown in Achdou and Tchou (2002) and Achdou and Pironneau (2009) that the solution exists and is unique if the data are smooth, and if

$$r(t) \geq r_0 > 0, \quad \alpha^2 > 2\beta^2, \quad 2\alpha^2\eta(1-\eta) > \beta^2.$$

4.5.1. Finite elements for a 2D stochastic volatility model

As above we localize the problem in $\Omega := (0, L_x) \times (-L_y, L_y)$. No boundary condition is needed on the axis $x = 0$ since the PDE is degenerate there. Notice also that u is expected to tend to zero as x tends to infinity. For large y no arguments seem to give a boundary condition. However, if L_y is chosen such that

$$\alpha(L_y \pm m) \gg \beta\left[\rho\frac{\mu - r}{f(L_y)} + \gamma\sqrt{1-\rho^2}\right],$$

then for $y \sim \pm L_y$, the coefficient of the advection term in the y direction, and as the advective effects will dominate the diffusive ones, even if a wrong boundary condition is imposed, it will not propagate inside the domain. Hence we may apply a Neumann condition.

Now Ω is triangulated into a set of non-intersecting triangles $\{T_k\}_1^K$ with vertices q^j satisfying

$$\begin{aligned}&\Omega_h := \cup_{k=1}^K T_k \approx \overline{\Omega}, \quad T_k \cap T_l = \emptyset \quad \text{or a vertex, or an entire edge;}\\ &\text{vertices on } \Gamma_h = \partial\Omega_h \text{ are on } \Gamma\text{; corners of } \Gamma \text{ are vertices of } \Gamma_h. \end{aligned} \tag{4.5.1}$$

V_0 is approximated by V_{0h}, the space of continuous piecewise linear functions on the triangulation which vanish at $x = L_x$:

$$V_{0h} := \{v_h \in C^0(\Omega_h) : v_h|_{T_k} \in P^1(T_k)\ k = 1, \ldots, K,\ v_h(L_x, y) = 0\}.$$

The basis $\{w^i\}_1^I$ is uniquely defined by

$$w^i \in V_{0h}, \quad w^i(q^j) = \delta_{ij} \quad \forall j$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Table 4.5.1. Comparison of CPU time for the LU algorithm and SuperLU for a product put option on a uniform mesh and 200 time steps (computed by N. Lantos).

Mesh size	Gauss-LU (s)	Relative error	SuperLU (s)	Relative error
101×101	10.094	3.532 %	2.39	3.076 %
126×126	14.547	1.338 %	4.016	1.797 %
151×151	22.313	0.751 %	6.203	0.489 %
176×176	31.985	1.131 %	8.735	0.790 %
201×201	43.938	0.432 %	12.109	0.670 %

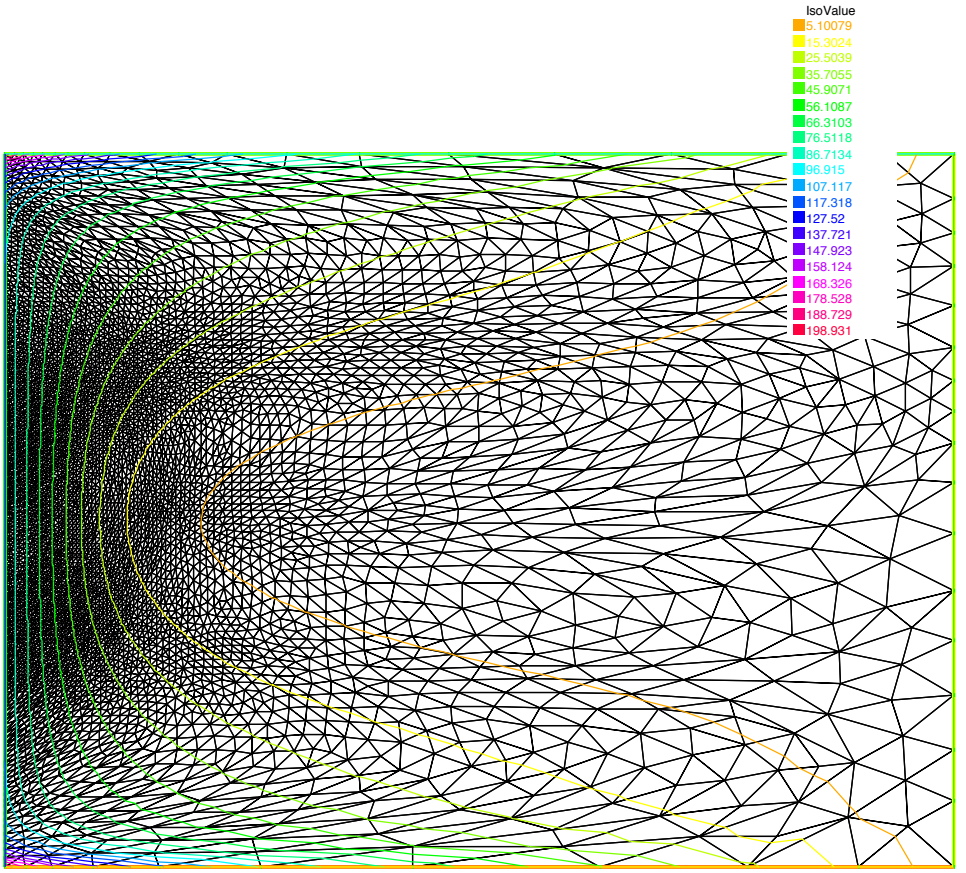


Figure 4.5.1. Solution of (4.3.1) for a put with maturity $T = 1$ and strike $K = 100$, with $r = 0.02$, $\beta = \alpha = 1$ and $m = 0.2$. We have used the automatic mesh adaptivity of FreeFem++; the contours and the triangulation are shown.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

for all index i such that the corresponding vertex q^i is not on the boundary $x = L_x$, *i.e.*, $q_1^i \neq L_x$.

As in the one-dimensional case, the variational formulation (4.3.4) is discretized by the Galerkin procedure (4.3.5), leading to (4.3.7) when

$$u_h^m(x,y) = \sum_1^I u_i^m w^i(x,y).$$

4.5.2. Numerical implementation

Integrals of polynomial expressions of x, y can be evaluated exactly on triangles (formula (4.19) in Achdou and Pironneau (2005)). The difficulty is in the strategy to solve the linear systems arising from (4.3.7). Multigrid seems to give the best results in term of speed (Ikonen and Toivanen 2008), but the computer program may not be so easy to write for an arbitrary triangulation. Since the number of vertices are not likely to exceed a few thousand, fast multifrontal libraries such as SuperLU (Xiaoye Li, Demmel and Gilbert 1999) can be used with minimal programming effort. Table 4.5.1 shows the gain in time over more classical algorithms.

Alternatively one could also use a preconditioned conjugate gradient iterative method. Another way is to use FreeFem++ (Hecht, Pironneau, Le Hyaric and Otsuka 2008), which is a general two-dimensional PDE solver particularly well suited to these option pricing models, where the coefficients are involved functions and one may want to try to change them often. Section 6.3, at the end of this article, is devoted to FreeFem++.

Results for the Stein–Stein model obtained with FreeFem++ are displayed in Figure 4.5.1.

Finite element solution of 3D problems

A basket option on three correlated assets has been computed by the same finite element method in 3D. The method and results are documented in Section 6.3.2.

4.6. Asian puts

Consider an Asian put with fixed strike whose pay-off is $P_0(S,y) = (K-y)^+$, where y is the average value of the asset in time, $y = \frac{1}{\tau}\int_0^\tau S(\tau)\,\mathrm{d}\tau$. The price of the option is found by solving, for all $\{x, y, \tau\} \in \mathbb{R}^+ \times \mathbb{R}^+ \times (0, T]$,

$$\begin{aligned} &\partial_\tau u - \partial_x\left(\frac{\sigma^2 x^2}{2}\partial_x u\right) - (\sigma^2 - r)x\partial_x u + \frac{y-x}{T-\tau}\partial_y u + ru = 0, \\ &u(x,y,0) = (K-y)^+,\ \partial_x u|_{+\infty,y,\tau} \approx \begin{cases} \frac{\tau}{T}\mathrm{e}^{-r\tau} & \text{if } y < \frac{KT}{T-\tau}, \\ \frac{1-r\mathrm{e}^{-r\tau}}{Tr} & \text{otherwise.} \end{cases} \end{aligned} \tag{4.6.1}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The problem is hard because the second-order part of the differential operator in y is missing, and because the convective velocity $v = \left((\sigma^2 - r)x, \frac{y-x}{T-t}\right)^T$ tends to infinity as $t \to T$.

Convective terms are not easy to discretize with finite element methods. Ideally, an equation such as

$$\partial_\tau u + v_1 \partial_x u + v_2 \partial_y u = 0, \quad u(x, y, 0) = u^0 \tag{4.6.2}$$

on an unstructured mesh ought to be discretized by a finite volume method rather than finite elements. There are ways to combine finite volume and finite element methods (Dervieux 1996), and more recently discontinuous Galerkin methods have also been shown to be well adapted, and to any order of accuracy, but numerically expensive. With triangular elements and polynomials of degree one, we can use streamline Galerkin least-squares or characteristic Galerkin for unwinding. Our preference is for the second scheme because there is no *ad hoc* parameter to tune.

The characteristic Galerkin finite element method is as follows:

$$\begin{aligned} &(\partial_\tau u + v \cdot \nabla u)|_{x,y,\tau^{m+1}} \approx \frac{1}{\delta\tau}(u^{m+1}(q) - u^m(Q)), \\ &\text{with} \quad Q = q - \delta\tau v\left(q - v^m(q)\frac{\delta\tau}{2}, \tau^{m+\frac{1}{2}}\right) \quad \text{and} \quad q = (x, y)^T. \end{aligned} \tag{4.6.3}$$

Boundary conditions for (4.6.2) ought to be u, given at time zero and on the part of the boundary where $v \cdot n < 0$, n being the outer normal to the boundary.

In the case of (4.6.1) the convective velocity is

$$v_1 = -(\sigma^2 - r)x, \quad v_2 = \frac{y - x}{T - \tau}.$$

After localization the domain of the PDE is $(0, L_x) \times (0, L_y)$. Hence $v \cdot n$ is

$$\begin{aligned} &v \cdot n|_{x=0} = -\frac{y}{T-\tau} < 0, \quad v \cdot n|_{x=L_x} = \frac{y - L_x}{T-\tau} < 0, \quad \text{if } L_y < L_x, \\ &v \cdot n|_{y=0} = (\sigma^2 - r)x > 0, \quad v \cdot n|_{y=L_y} = (\sigma^2 - r)x < 0, \quad \text{if } \sigma^2 > r. \end{aligned} \tag{4.6.4}$$

This shows that boundary conditions are needed everywhere except at $y = 0$. One possibility is to use a discounted value of the initial condition at $x = 0$ and a condition compatible with the asymptotic behaviour expected for x or y large:

$$u(x, y, \tau) = \mathrm{e}^{-r\tau} K \text{ at } x = 0, \quad u(L_x, y, \tau) = \mathrm{e}^{-r\tau}(K - y)^+, \quad u(x, L_y, \tau) = 0.$$

The discontinuous P^2 reconstruction with characteristic convection of the gradient and of the value at the centre of gravity is applied. The domain is localized to $(0, L) \times (0, L)$, with $L = 250$, *i.e.*, greater than twice the

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

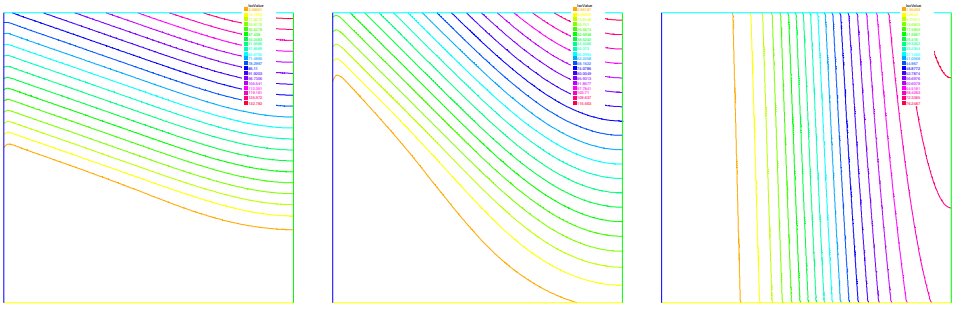


Figure 4.6.1. Solution of (4.6.1) for an Asian option of maturity $T = 4$ and strike $K = 100$ when the interest rate is 0.03 and the volatility 0.3. The three figures show the contours at times 0.96, 1.96 and 3.96.

contract price $K = 100$, in this example; the interest rate is $r = 3\,\%$ and the volatility $\sigma = 0.3$. The square domain is triangulated by a uniform 50×50 mesh and we have taken 50 time steps over the 4-year period of this example ($T = 4$). The results are shown in Figure 4.6.1 at times $t = 0.96$, 1.96 and 3.96.

4.7. Automatically adapted meshes

When an *a posteriori* error estimate is available, mesh adaptation based on this example is a sure way to reduce the error below a given threshold; when it is not available, local mesh adaptation by metric control is a powerful tool for building meshes well suited to the problem to be solved (George and Borouchaki 1998).

Delaunay criterion. A *Delaunay mesh* is one where all inner edges are such that the circle defined by one triangle having the edge as a side is such that the third vertex of the other triangle having the same edge as a side is outside the circle.

A quick way to make a Delaunay mesh is as follows.

4.7.1. Delaunay mesh generator

(1) Unmark all edges.
(2) Loop on the unmarked inner edges until all are marked.

- For each edge identify the two triangles which have this edge as a side. If the edge fails the Delaunay test above, then swap the edge with the other diagonal of the quadrilateral made by the two triangles, and update the triangulation by replacing the two old triangles by the two new ones.
- Mark the new edge and unmark the four edges of the quadrilateral.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

It can be shown that the loop stops in finite time because the minimum angle in the triangulation is strictly increased at each loop, and that the resulting mesh is Delaunay if the domain is convex.

If the domain is not convex, a complicated recovery of the boundary edges must be performed, but still the mesh has similar good qualities in most cases (see George and Borouchaki (1998)). In fact it can be shown that the Delaunay mesh is the nearest to a quasi-equilateral mesh, in the sense that the smallest angle in the triangulation is maximized by the edge swaps so as to satisfy the Delaunay criterion.

4.7.2. Anisotropic Delaunay mesh generator

Note that if the local metric is Euclidean, the mesh elements are isotropic in the sense that there is no preferred direction in the algorithm.

To obtain an anisotropic mesh one must replace circles by ellipses in the Delaunay criterion.

Given a positive definite matrix $M(x)$, we define a variable metric by $\|x-y\|^2 = (x-y)^T M(x)(x-y)$. M-circles, *i.e.*, circles with respect to the variable metric, are ellipses in the Euclidean space.

A given triangulation is said to satisfy the M-Delaunay criterion if, for all inner edges, the quadrangle made by the two adjacent triangles is such that the fourth vertex is outside the M-circle passing through the other three vertices.

The adaptive Delaunay mesh generator of George and Borouchaki (1998) is based on six steps.

(1) Discretize the boundary of Ω, the domain for which we seek a triangulation.
(2) Build an M-Delaunay triangulation of the convex hull of all the boundary nodes (no internal points).
(3) Add an internal point to all edges which are longer (in the variable metric) than the prescribed length.
(4) Rebuild the triangulation with internal nodes, using the Delaunay criterion and the variable metric.
(5) Remove the triangles which are outside Ω.
(6) Go to (3) until the elements have the required quality.

So anisotropy is controlled by M, the local metric.

4.7.3. Metric adapted to the solution of the PDE

As stated earlier, if we want the mesh to be adapted to the solution, we need to define the metric at each point of the domain and use it in the Delaunay algorithm above.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The definition of the metric is based on the Hessian of the state variables of the problem. Indeed, for a P^1 Lagrange discretization of a variable u, the interpolation error is bounded by

$$\mathcal{E} := |u - \Pi_h u|_0 \leq ch^2 |D^2 u|_0, \tag{4.7.1}$$

where h is the element size, $\Pi_h u$ the P^1 interpolation of u and $D^2 u$ its Hessian matrix. This matrix is symmetric and therefore diagonalizable:

$$D^2 u = \begin{pmatrix} \partial^2 u/\partial x^2 & \partial^2 u/\partial x \partial y \\ \partial^2 u/\partial x \partial y & \partial^2 u/\partial y^2 \end{pmatrix} = \mathcal{R} \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \mathcal{R}^{-1}. \tag{4.7.2}$$

Using this information, we introduce the following metric tensor $\mathcal{M}$:

$$\mathcal{M} = \mathcal{R} \begin{pmatrix} \tilde{\lambda}_1 & 0 \\ 0 & \tilde{\lambda}_2 \end{pmatrix} \mathcal{R}^{-1}, \tag{4.7.3}$$

where

$$\tilde{\lambda}_i = \min\left(\max\left(|\lambda_i|, \frac{1}{h_{\max}^2} \right), \frac{1}{h_{\min}^2} \right), \tag{4.7.4}$$

with $h_{\min}$ and $h_{\max}$ being the minimal and maximal edge lengths allowed in the mesh.

Now, if we generate, by a Delaunay procedure, an equilateral mesh with edges of length 1 in the metric $\mathcal{M}/(c\mathcal{E})$, the interpolation error $\mathcal{E}$ is equidistributed over the edges of length a_i if

$$\frac{1}{c\mathcal{E}} a_i^T \mathcal{M} a_i = 1. \tag{4.7.5}$$

4.8. Greeks

Probably due to the fact that mathematicians have used Greek letters to name the partial derivatives of option prices with respect to the relevant parameters, these are called the *Greeks*. Let P be the price of a vanilla European put.

- The *delta*, δ, is its derivative with respect to the stock price S: $\partial_S P$.
- The Θ, or *time-decay*, is its derivative with respect to time: $\partial_t P$.
- The *vega* κ is its derivative with respect to the volatility $\partial_\sigma P$.
- The *rho*, ρ, is its derivative with respect to the interest rate, $\partial_r P$.
- The *eta*, η, is its derivative with respect to the strike $\partial_K P$.
- The *gamma*, γ, is the rate of change of its delta: $\gamma = \partial_{SS} P = \partial_S \delta$.

Computing Greeks from the SDE of the asset can be challenging, but the same is very easily done with the PDE of the option. For an illustration, take the vanilla put on S_t written in moneyness y and time to maturity τ:

$$\partial_\tau u - \frac{y^2 \sigma^2}{2} \partial_{yy} u = 0, \quad u(y, 0) = (1 - y)^+.$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

A finite element or finite difference method numerical solution will give the δ with order-one precision. If more precision is needed, by differentiation of the PDE with respect to y, $v = \partial_y u$ satisfies

$$\partial_\tau v - \frac{y^2\sigma^2}{2}\partial_{yy} v = (y\sigma^2 + y^2\sigma\partial_y\sigma)\partial_{yy} u, \quad v(y,0) = H(y-1) - 1,$$

with $H(x)$ the Heaviside function, which is zero for $x < 0$ and one otherwise. Thus the delta is $\delta = Kv(\mathrm{e}^{-r(T-t)S/K}, T-t)$. Further,

$$\partial_\tau v - \frac{y^2\sigma^2}{2}\partial_{yy} v - (y\sigma^2 + y^2\sigma\partial_y\sigma)\partial_y v = 0, \quad v(y,0) = H(y-1) - 1.$$

The Θ can be computed by $\Theta = y^2\sigma^2\partial_y v/2$ or by differentiating the equation for v.

The vega satisfies

$$\partial_\tau \kappa - \frac{y^2\sigma^2}{2}\partial_{yy}\kappa = y^2\sigma\partial_{yy} u, \quad \kappa(y,0) = 0.$$

Alternatively, as explained in Achdou and Pironneau (2005, 2009), if the numerical solver for u is written in C++, all Greeks can be computed at no programming cost by automatic differentiation using operator overloading and template classes.

Greeks for American options

American options have been recast as variational inequalities such as (4.4.4):

$$u - \phi \in V^+ : \partial_\tau(u,w) + a(u,w) \geq (f,w) \quad \forall w \in V^+ = \{w \in V : w \geq 0\},$$

where $(\cdot,\cdot)$ is the scalar product of L^2, $a(\cdot,\cdot)$ is a continuous coercive bilinear form on the Hilbert space V and $f \in V$. Another form of the same is, for some operator A,

$$u - \phi \in V^+ : \partial_\tau u + Au \geq f \quad \forall w \in V^+ = \{w \in V : w \geq 0\}.$$

To compute the sensitivity of u with respect to a parameter α, it is best to return to the definition of the derivative and work out the equation/inequality for the increments. For instance,

$$u(\alpha) \geq \phi(\alpha), \quad u(\alpha + \delta\alpha) \geq \phi(\alpha + \delta\alpha) \;\Rightarrow\; u'_\alpha \geq \phi'_\alpha \quad \text{at } S, \tau,$$

where $u(\alpha) = \phi(\alpha)$. Similarly

$$\partial_\tau u'_\alpha + Au'_\alpha + \lambda' = f'_\alpha - A'_\alpha u,$$

with $\lambda' \leq 0$ at all S, τ where $\partial_\tau u + Au = f$ and $\lambda' = 0$ at all other points. But by (4.4.4) we know that, at the point where $\partial_\tau u + Au \neq f$, we have $u = \phi$, and consequently u'_α is a solution of

$$u'_\alpha \geq \phi'_\alpha, \quad \partial_\tau u'_\alpha + Au'_\alpha \geq f'_\alpha - A'_\alpha u,$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

or equivalently

$$u'_\alpha - \phi'_\alpha \in V^+ : \partial_\tau(u'_\alpha, w) + a(u'_\alpha, w) \geq (f'_\alpha, w) - a'_\alpha(u, w) \quad \forall w \in V^+.$$

The semi-smooth Newton algorithm can be used for u'_α exactly as for u.

4.9. Reduced-order modelling

Derivatives such as European calls and puts on one or many assets are computed a very large number of times every day. When a closed-form solution is not available, an alternative to Monte Carlo simulation is to solve the Black–Scholes partial differential equation.

A complement to the finite element or finite difference method for a numerical solution of the problem is to construct an appropriate basis, smaller in size but with larger support. Proper orthogonal decomposition (POD) is the usual tool for this.

Cont, Lantos and Pironneau (2011) showed that a set of rescaled calls with constant volatilities forms a reduced basis with a similar performance to POD.

If τ is the time to maturity, K is the strike, r the interest rate, and $y = \mathrm{e}^{r\tau}\frac{S}{K}$ the moneyness, a put P can be priced by solving

$$\partial_\tau v_\sigma - \frac{\sigma^2 y^2}{2}\partial_{yy} v_\sigma = 0, \quad v_\sigma(0) = (1-y)^+, \tag{4.9.1}$$

where $v_\sigma = P\mathrm{e}^{r\tau}$.

When σ is constant, the Black–Scholes formula provides the analytical solution

$$v_\sigma = \frac{y}{2}\left(1 + \operatorname{erf}\left(\frac{\ln y}{\sigma\sqrt{2\tau}} + \sigma\sqrt{\frac{\tau}{8}}\right)\right) - \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{\ln y}{\sigma\sqrt{2\tau}} - \sigma\sqrt{\frac{\tau}{8}}\right)\right) + 1 - y \tag{4.9.2}$$

where $\operatorname{erf}(y) = \frac{2}{\sqrt{\pi}}\int_0^y \mathrm{e}^{-x^2}\,\mathrm{d}x$. Note that only $\sigma\sqrt{\tau}$ appears, so *changing the volatility is equivalent to changing* τ. Hence a basis of snapshots in time for (4.9.1) can also be

$$v_\sigma(y,\tau) = v_\Sigma(y,\tau) + \sum_{i=1}^{I} a_i(\tau)\bar{w}_i(y) \quad \text{with } \bar{w}_i := v_{\sigma_i}(y,T) - v_\Sigma(y,T), \tag{4.9.3}$$

where $\{\sigma_i\}_{i\in I}$ is an appropriate set of constant volatilities and Σ is a chosen reference volatility. The PDE satisfied by $u := v_\sigma - v_\Sigma$ is

$$\partial_\tau u - \frac{\sigma^2 y^2}{2}\partial_{yy} u = -\left(\partial_\tau v_\Sigma - \frac{\sigma^2 y^2}{2}\partial_{yy} v_\Sigma\right), \quad u(0) = 0. \tag{4.9.4}$$

Applying Galerkin's method to (4.9.4) with (4.9.3) will give the $a_i(\tau)$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

However, it is numerically much easier to work with $\hat{w}_i := \partial_{yy} v_{\sigma_i}(y,T)$ instead of $\bar{w}_i = v_{\sigma_i}(y,T) - v_\Sigma(y,T)$, and since ∂_{yy} is a bijection with appropriate boundary conditions it should not make much of a difference!

As we shall see, a good choice is $\sigma_i = (2c\,T\,i)^{-1/2},\ i = 1,2,\dots$, where c is a chosen constant. A quick computation shows that $\hat{w}_i$ is proportional to

$$w_i := \sqrt{y}\mathrm{e}^{-\alpha_i \ln^2 y}, \quad \text{with } \alpha_i = c\ i.$$

This will be the basis chosen here, and $c = 1$ in the numerical tests.

The POD algorithm. In this framework POD entails the following.

(1) Choose $I \ll 100$.

(2) Apply the Gram–Schmidt algorithm to orthogonalize the $\{w_i\}_{i\in I}$ with respect to the $L^2(\mathbb{R}^+)$ or $H^1(\mathbb{R}^+)$ scalar product $(\cdot,\cdot)$. Let $\{p_i\}_{i\in I}$ be the result.

(3) Solve (4.9.4) on this basis by the Galerkin method.

(4) If the result is not precise enough, add a new element in I and go back to step (2).

The 'not precise enough' of step (4) needs in principle an *a posteriori* estimate to become mathematically sound. The selection of a 'good new basis vector' is also difficult in step (4), and one way is to order the vectors p_i so as to have a decreasing sequence of eigenvalues of the correlation matrix (w_i, w_j) (see Prud'homme *et al.* (2002) for details).

4.9.1. The Galerkin approximation

Mathematical result

Decomposition (4.9.3) gives a v_σ which satisfies the initial and boundary conditions of the Black–Scholes equations (4.9.1); it also has the right exponential behaviour when $y \to \infty$. Consider $y \to u(y)/\sqrt{y}$ and, as before, $u = v_\sigma - v_\Sigma$.

Note that $\mathrm{e}^{-ic\ln^2 y} = f(y)^i$ with $f(y) = \mathrm{e}^{-c\ln^2 y}$. By the Stone–Weierstrass theorem, $\{f(y)^n\}_{n=0,1,\dots}$ is a basis if $y \to f(y)$ separates points. However, $\mathrm{e}^{-c\ln^2 y}$ is a separating function only on the interval $(0,1)$ or on $(1,\infty)$ but not on $(0,\infty)$. Thus we need a symmetry assumption about 1, $u\big(\frac{1}{y}\big) = \frac{1}{y}u(y)$, for (4.9.3) to work. Fortunately, if $\sigma\big(\frac{1}{y},\tau\big) = \sigma(y,\tau)$ then it is easy to show that u has the required symmetry. If symmetry does not hold, then another set of vectors must be added to the basis, such as $\{\sqrt{y}w^i(y)\}_{i=1,2,\dots}$. More details can be found in Cont *et al.* (2011).

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Implementation
Note that

$$\partial_{yy} w^i = \frac{\mathrm{e}^{-\alpha_i \ln^2 y}}{y\sqrt{y}}\left(4\alpha_i^2 \ln^2 y - \frac{1}{4} - 2\alpha_i\right), \tag{4.9.5}$$

$$\partial_\tau v_\Sigma - \frac{\sigma^2 y^2}{2}\partial_{yy} v_\Sigma = \frac{(\Sigma^2-\sigma^2)y^2}{2}\partial_{yy} v_\Sigma = \frac{\mathrm{e}^{-\frac{\Sigma^2\tau}{8}}}{2\Sigma\sqrt{2\pi\tau}}(\Sigma^2-\sigma^2)\sqrt{y}\mathrm{e}^{-\frac{\ln^2 y}{2\Sigma^2\tau}}.$$

So (4.9.3) is

$$\sum_1^I \dot{a}_i\sqrt{y}\mathrm{e}^{-\alpha_i \ln^2 y} - \sum_1^I a_i \mathrm{e}^{-\alpha_i \ln^2 y}\sigma^2\sqrt{y}\left(2\alpha_i^2 \ln^2 y - \frac{1}{8} - \alpha_i\right)$$
$$= -\frac{\mathrm{e}^{-\frac{\Sigma^2\tau}{8}}}{2\Sigma\sqrt{2\pi\tau}}(\Sigma^2-\sigma^2)\sqrt{y}\mathrm{e}^{-\frac{\ln^2 y}{2\Sigma^2\tau}}. \tag{4.9.6}$$

For the Galerkin method we multiply this equation by $\gamma(y)\sqrt{y}\mathrm{e}^{-\alpha_j \ln^2 y}$ (the extra factor $\gamma(y)$ is a weight) and integrate over $\mathbb{R}^+$. We obtain an ODE system for $a = (a_1(\tau), \ldots, a_I(\tau))^T$:

$$M\dot{a} + Ba = F, \tag{4.9.7}$$

with

$$M_{ij} := \int_0^\infty \mathrm{e}^{-(\alpha_i+\alpha_j)\ln^2 y}\gamma(y) y \,\mathrm{d}y,$$
$$B_{ij} := -\int_0^\infty \mathrm{e}^{-(\alpha_i+\alpha_j)\ln^2 y}\sigma^2\left(2\alpha_i^2 \ln^2 y - \frac{1}{8} - \alpha_i\right)\gamma(y) y \,\mathrm{d}y,$$
$$F_j := -\frac{\mathrm{e}^{-\frac{\Sigma^2\tau}{8}}}{2\Sigma\sqrt{2\pi\tau}}\int_0^\infty \mathrm{e}^{-(\alpha_j+\frac{1}{2\Sigma^2\tau})\ln^2 y}(\Sigma^2-\sigma^2)\gamma(y) y \,\mathrm{d}y. \tag{4.9.8}$$

The case σ constant
When σ is constant and $\gamma(y) = y^{-2}$,

$$M_{ij} = \sqrt{\frac{\pi}{\alpha_i+\alpha_j}},$$
$$B_{ij} = \frac{\sqrt{\pi}\sigma^2}{(\alpha_i+\alpha_j)^{\frac{1}{2}}}\left(\frac{\alpha_i\alpha_j}{\alpha_i+\alpha_j} + \frac{1}{8}\right),$$
$$F_j = -\frac{\mathrm{e}^{-\frac{\Sigma^2\tau}{8}}(\Sigma^2-\sigma^2)}{2\sqrt{2\alpha_j\Sigma^2\tau+1}}. \tag{4.9.9}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

The non-constant case

If σ is a function of y and τ, with $\sigma\left(\frac{1}{y},\tau\right)=\sigma(y,\tau)$, it is best to express it using an exponential basis, such as

$$\sigma(x,t)=\sigma_0+\sum_1^J \sigma_j(t)\mathrm{e}^{-\alpha_j \ln^2 y}, \tag{4.9.10}$$

because then all integrals can be computed analytically.

Another choice for γ

Later on we will change variables to $z=\mathrm{e}^{-\ln^2 y}$. Then, in view of the fact that $\mathrm{d}z=\frac{2}{y}\mathrm{e}^{-\ln^2 y}\ln\frac{1}{y}\,\mathrm{d}y$, we will need to choose

$$\gamma(y)=\frac{2}{y^2}|\ln y|,$$

so as to have

$$\mathrm{e}^{-\ln^2 y}y\gamma(y)\,\mathrm{d}y=\mathrm{e}^{-\ln^2 y}\frac{2}{y}|\ln y|\,\mathrm{d}y=\mathrm{d}z.$$

Then, with σ constant,

$$\begin{aligned}
M_{ij}&=\frac{1}{\alpha_i+\alpha_j},\\
B_{ij}&=\frac{\sigma^2}{\alpha_i+\alpha_j}\left[\frac{1}{8}+\alpha_i-\frac{2\alpha_i^2}{\alpha_i+\alpha_j}\right],\\
F_j&=-(\Sigma^2-\sigma^2)\frac{\mathrm{e}^{-\frac{\Sigma^2\tau}{8}}\sqrt{2\Sigma^2\tau}}{2\sqrt{\pi}(1+2\Sigma^2\tau\alpha_j)}.
\end{aligned} \tag{4.9.11}$$

The computation of the integrals relies on the following:

$$\begin{aligned}
\int_0^\infty \mathrm{e}^{-k\ln^2 y}\gamma(y)y\,\mathrm{d}y&=2\int_0^1 z^k\,\mathrm{d}z=\frac{2}{k+1},\\
\int_0^\infty \mathrm{e}^{-k\ln^2 y}\ln^2\frac{1}{y}\gamma(y)y\,\mathrm{d}y&=2\int_0^1 z^k\ln\frac{1}{z}\,\mathrm{d}z=\frac{2}{(k+1)^2}.
\end{aligned} \tag{4.9.12}$$

4.9.2. Numerical tests

The method was thoroughly tested by Cont *et al.* (2011), but since we are concerned with a minor variation of the original, we retested the method for the computation of a vanilla call of volatility $\sigma=\sqrt{0.9}$, using $\Sigma=\sqrt{0.5}$ and I ranging from 5 to 30 (using multi-precision arithmetic when $I>20$). The maturity is $T=2$ and 10 time steps are used to integrate (4.9.7); with 20 time steps the precision is not significantly improved.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

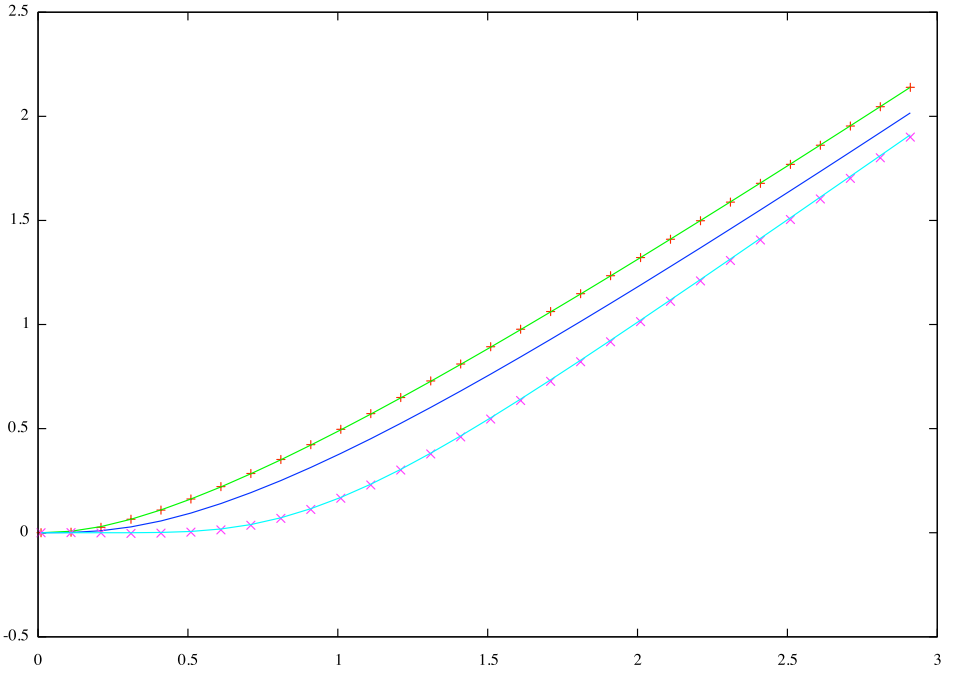
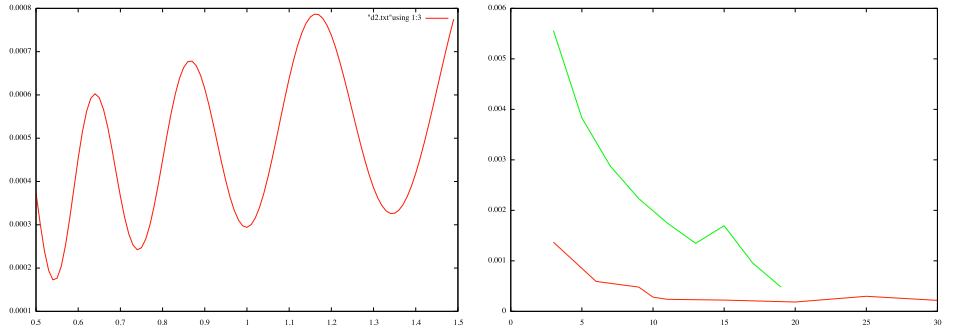


Figure 4.9.1. Here $\Sigma = \sqrt{0.5}$, and the call for this volatility is the middle curve. The call computed by the reduced basis method overlaps perfectly, even with $I = 5$ here. The call is computed analytically for both $\sigma = \sqrt{0.9}$ (the highest curve and the plus marks) and $\sigma = \sqrt{0.3}$ (the lowest curve and the cross marks).



(a) (b)

Figure 4.9.2. (a) Errors with $I = 10$ (difference between the analytical Black–Scholes formula and this method) in the interval $y \in (0.5, 1.5)$ are shown to be below the 0.1 % accepted standard. (b) The upper curve is the L^1-error versus ten times the time to maturity τ. The lower curve is the L^1-error at $\tau = T$ versus the number of basis functions I.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Figures 4.9.1 and 4.9.2 show that the precision of the method is well within the three digits required by banks. Calls are computed from puts by the put–call parity relation. The linear system is solved with the SVD module of Press, Teukolsky, Vettering and Flannery (1992). However, ill-conditioning is such that even SVD fails when I is larger than 20 or so. An analytical orthogonalization of the basis is necessary, as used in spectral methods when one replaces the natural polynomial basis z^n by the Legendre polynomials.

4.9.3. Sparse grids

To compute one price with a d-dimensional SDE, Monte Carlo algorithms are not cursed by the dimension d in the sense that their computational complexity is proportional to d^2. With PDEs, even storing the solution costs $O(N^d)$ operations on a uniform grid of size N in each direction. Sparse grids can potentially deal with this dimensional curse.

The idea that underlies sparse grids is due to Smolyak (1963). It is possible to compute $\int_\Omega f$, $\Omega \subset \mathbb{R}^d$, by a quadrature formula, where the quadrature points are on a manifold of dimension $\mathbb{R}^{d-1}$. Smolyak's method achieves a precision proportional to $N^{-r}|\ln N|^{(d-1)(r+1)}$ with a grid of size N when $f \in C^r(\Omega)$.

This is a recursive argument. Indeed, if the quadrature points were on the boundary, for instance, then the sum which approximates the integral could be seen as an approximation of another integral on the $\partial\Omega$, which itself could be approximated by quadrature with points on a manifold of dimension $d-2$, *etc.*

Let $Q_i^{(1)}$ be I quadrature rules for $g : \mathbb{R} \to \mathbb{R}$ with quadrature points x_k^i and weights w_k^i:

$$Q_i^{(1)} g = \sum_{k=0}^{n_i-1} w_k^i g(x_k^i), \quad i = 1, \dots, I, \quad Q_0^{(1)} g = 0.$$

For example, for a quadrature on $(-L, +L)$,

$$n_i = 2^i, \quad w_k^i = \frac{2L}{n_i}, \quad x_k^i = -L + \frac{2(k+\frac{1}{2})L}{n_i}, \quad k = 0, \dots, n_i - 1.$$

Smolyak computes $Q_I^{(d)} f$ by

$$Q_i^{(p)} f = \sum_{j=1}^{i} (Q_j^{(1)} - Q_{j-1}^{(1)}) \otimes Q_{i-j+1}^{(p-1)} f, \quad i = 1, \dots, I, \quad p = 2, \dots, d,$$
$$Q_0^{(p)} f = 0. \tag{4.9.13}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Note that if $x = (x^1, x^2, \ldots, x^d)^T$, then $Q_i^{(p)}$ is an approximation of the p-multiple integral of f with respect to $x^1, \ldots, x^p$, so it is a function of $x^{p+1} \cdots x^d$, and $Q_j^{(1)} \otimes$ is an approximate integration with respect to the first free variable, namely x^{p+1}. More details can be found in Gerstner and Griebel (1998).

Remark 4.9.1. Equivalently

$$Q_i^{(p)} f = \sum D_{j_1} \otimes \cdots \otimes D_{j_p} f \quad \text{with } D_k = Q_k^{(1)} - Q_{k-1}^{(1)}. \tag{4.9.14}$$

The method is easy to apply to the computation of a basket option by (1.5.20). Even though the regularity is $r = 0$, the method gives reasonably good results. The following program implements (4.9.14). Results are shown in Figure 4.9.3. Improvement can be obtained by using recursively embedded grids and computing directly $Q_k - Q_{k-1}$ instead of Q_k and Q_{k-1} separately.

```
#include <iostream>
#include <cmath>
#include <ctime>
using namespace std;
#define d 5          // dimension of the integration space
const double L=2*atan(1.);   // size of the cube (-L,L)^d;
const int m=4;       // max number of points of quadrature in each dim
int *y, ik[d];       // container of a distribution of integers with sum<m+d
double *z;           // will be all the quadrature points in R^d, each in turn

void explore(int p, int s); // finds y[] s.t. sum of y<=s,
double f(double *y);        // the function to integrate
void QId(int* y);           // the 1d integration procedure

int *ni,*kd; // working array
double somme=0, L2=2*L, pi=4*atan(1.0), xpi=sqrt(pow(2*pi,d));

double f(double *x){ double aux=1;
for (int i=0;i<d;i++) aux*=cos(x[i]/2); return aux;}
/*  double f(double *z){ double aux=0, aux1=0, pid2=1;
        for (int i=0;i<d;i++){ aux+= z[i]; aux1+=z[i]*z[i];pid2*=2*pi;}
        return max(1-exp(aux),0.)*exp(-aux1/2)/sqrt(pid2);
} */ // change to this f for finance
void explore(int p, int s){ // p ends recursion
    if(p>0)
        for(y[p]=0;y[p]<=s;y[p]++)
                explore(p-1,s-y[p]);
    else {
            y[p]=s;// here p==0
        QId(y);
    }
}
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

```
void QId(int* y){
    double dd;
    for( ik[0]=0;ik[0]<=1;ik[0]++) // to add dimensions add loops below
      for( kd[0]=0;kd[0]<ni[y[0]-ik[0]];kd[0]++)
#if(d>1)
    for( ik[1]=0;ik[1]<=1;ik[1]++) for( kd[1]=0;kd[1]<ni[y[1]-ik[1]];kd[1]++)
#endif
#if(d>2)
    for( ik[2]=0;ik[2]<=1;ik[2]++) for( kd[2]=0;kd[2]<ni[y[2]-ik[2]];kd[2]++)
#endif
#if(d>3)
    for( ik[3]=0;ik[3]<=1;ik[3]++) for( kd[3]=0;kd[3]<ni[y[3]-ik[3]];kd[3]++)
#endif
#if(d>4)
    for( ik[4]=0;ik[4]<=1;ik[4]++) for( kd[4]=0;kd[4]<ni[y[4]-ik[4]];kd[4]++)
#endif
#if(d>5)
    for( ik[5]=0;ik[5]<=1;ik[5]++) for( kd[5]=0;kd[5]<ni[y[5]-ik[5]];kd[5]++)
#endif
#if(d>6)
    for( ik[6]=0;ik[6]<=1;ik[6]++) for( kd[6]=0;kd[6]<ni[y[6]-ik[6]];kd[6]++)
#endif
      {
        dd=1;
        for(int k=0;k<d;k++){
            int iy=y[k]-ik[k], nk = ni[iy];
            if(iy>=1){
                z[k] = -L + (L2*kd[k]+L)/nk;
                dd *= ik[k]? -L2/nk : L2/nk;
            } else dd=0;
            if(dd==0)break;
          }
          if(dd!=0)   somme += f(z)*dd;
      }
 }
int main(){
    y = new int[d];
    ni = new int[m+1];
    kd = new int[d];
    z = new double[d];
    ni[0]=1; for(int k=1;k<=m;k++) ni[k] =2*ni[k-1];  // ni[k]=2^k;
    time_t time0=clock();
    for(int n=1;n<m+d;n++) explore(d-1,n); // y[] with sum_0^{d-1} y[j]=n
    cout <<"{\sc cpu}=\t"<<double(clock()-time0)/CLOCKS_PER_SEC
         <<"\t relative error = \t"<<fabs(1-pow(4*sin(L/2),d)/somme)<<"
         "<<somme<<endl;
    return 0;
}
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

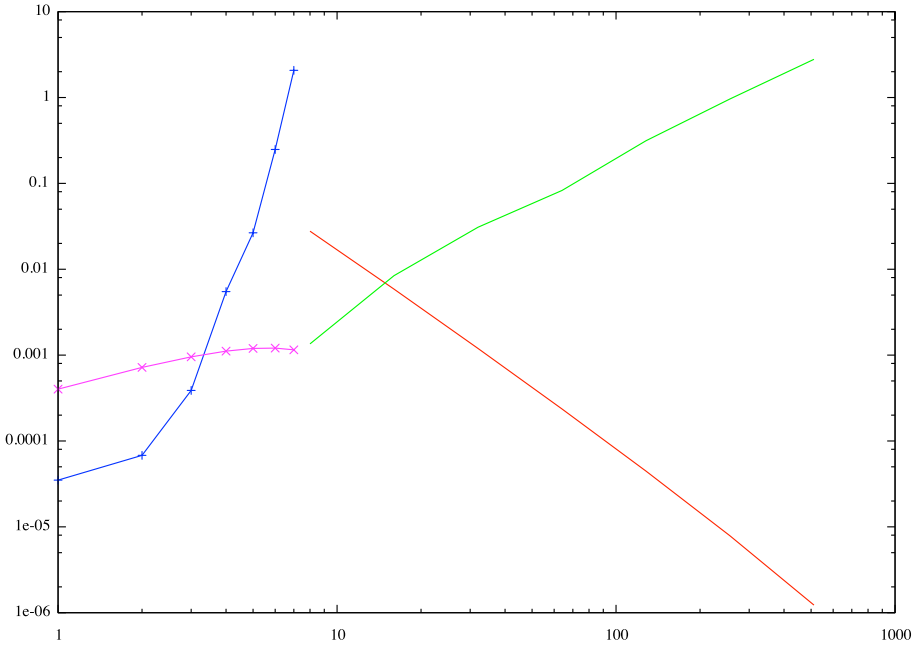


Figure 4.9.3. Left steeply ascending curve: log-log CPU time (in seconds) versus d, the dimension, at $I = 4$ obtained by the sparse grid implementation described above and applied to $f(x) = \Pi_1^d \cos(\frac{x_i}{2})$, $x \in [-\frac{\pi}{2}, \frac{\pi}{2}]^d$, $L = \frac{\pi}{2}$. Left slightly ascending curve (log-log): precision versus d at $I = 4$, showing only a mild degradation of the precision with dimension for a fixed N in each direction. Right descending curve (log-log): precision versus N at $d = 5$. Right ascending curve (log-log): CPU time versus N at $d = 5$.

PART FIVE
Calibration

5.1. Introduction

Whatever the quality of the modelling, the simulations usually do not reproduce the market data. Besides, some parameters, like volatilities, are not easy to estimate. On the other hand many assets and options are observable, and available, albeit possibly at a cost (for example, Reuters provide such data, as do Bloomberg).

Consider the problem of building an accurate model for a vanilla European call on S_t. Today, at time $t = 0$, we can observe

- the underlying asset S_0,
- the values $(c_i)_{i\in I}$ of calls $(C_{K_i,T_i}(S_0, 0))_{i\in I}$ of maturity T_i and strike K_i.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

We wish to use the Black–Scholes PDE with a local (*i.e.*, non-constant) volatility, so we must find $x, t \to \sigma(x,t)$ such that the $\{C_i\}_{i\in I}$ computed by

$$\begin{aligned}&\partial_t C_i + \frac{\sigma^2 S^2}{2}\partial^2_{SS} C_i + rS\partial_S C_i - rC_i = 0,\\ &C_i(S,T_i) = (S-K_i)^+, \quad \text{for all } t\in[0,T_i[,\ x\in\mathbb{R}^+,\end{aligned} \tag{5.1.1}$$

satisfy $C_i(S_0,0) = c_i$

The most natural way to proceed is to solve this *inverse problem* by least-squares:

$$\min_{\sigma\in\mathcal{S}} \sum |C_i(S_0,0) - c_i|^2 \quad \text{subject to (5.1.1).}$$

This problem has three difficulties.

(1) It is ill-posed unless $\mathcal{S}$ is severely constrained or a *Tikhonov regularization* is added to the cost function (see Crépey (2003)).

(2) It has infinitely many solutions, as we shall see.

(3) It is numerically expensive to solve.

Curse number (3) is problematic. One reason is that it is an optimal control problem with card(I) PDEs. Working with the moneyness variables

$$\tau = T - t, \quad y = \mathrm{e}^{r\tau}\frac{S}{K}, \quad Ku(y,\tau) = \mathrm{e}^{r\tau}C(K\mathrm{e}^{-r\tau}y, T-\tau)$$

reduces the number of PDEs to one, and the problem becomes

$$\begin{aligned}&\min_{\sigma\in\mathcal{S}} \sum |\mathrm{e}^{rT_i}K_i u\left(\mathrm{e}^{rT_i}\frac{S_0}{K_i}, T_i\right) - c_i|^2\\ &\text{subject to} \quad \partial_\tau u - \frac{\sigma^2 y^2}{2}\partial_{yy}u = 0, \quad u(y,0) = (y-1)^+,\end{aligned} \tag{5.1.2}$$

where $\mathcal{S} = \{\sigma(y,\tau) : \sigma_m \le \sigma(y,\tau) \le \sigma_M\}$. If a Tikhonov regularization of the form

$$\|\sigma\|_0^2 + \|\partial_y\sigma\|_0^2 + \|\partial_\tau\sigma\|_0^2 + \|\partial_{y\tau}\sigma\|_0^2$$

is added to the cost function, then the problem is well posed and solvable numerically via a conjugate gradient method, for example (see Achdou and Pironneau (2005)).

However, the explicit dependence of C on K is specific to this problem and the method is not general. Dupire had a better idea!

5.2. Dupire's equation

Consider a call priced by (3.2.11). To stress explicitly the dependence on the parameters K, T we write the solution as $C_{K,T}(S,t)$.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Dupire (1997) observed that an equation for $C_{S,t}(K,T)$ can be derived, where S,t are now parameters:

$$\begin{aligned}&\partial_\tau C - \frac{1}{2}\sigma^2 K^2 \partial_{KK} C + rK\partial_K C = 0,\\ &C(K,t_0) = (S-K)^+ \quad \forall K>0, \tau \geq t_0.\end{aligned} \tag{5.2.1}$$

Dupire's proof (Dupire 1997) uses the Green's function of the differential operator in (3.2.11). We shall give a proof which can be extended to any financial object governed by a *linear* system. The proof relies on a duality argument, such as the forward and backward Kolmogorov equation. The proof also shows that a double primitive of the dual of the Black–Scholes PDE is involved. Thus it is misleading to use K,T as variables in (5.2.1).

Proposition 5.2.1. (Dupire) Let v be a solution in $\mathbb{R}^+ \times (t_0,T)$ of

$$\partial_t v - \frac{1}{2}\sigma^2 S^2 \partial_{SS} v + rS\partial_S v = 0, \quad v(S,t_0) = (S_0 - S)^+. \tag{5.2.2}$$

Then $C_{K,T}(S_0,t_0) = v(K,T)$ and $p = \partial_{SS} v$ is a solution of the adjoint of (3.2.11):

$$\partial_t p - \partial_{SS}\left(\frac{\sigma^2 S^2}{2} p\right) + \partial_S(rSp) + rp = 0,\ p(t_0) = \delta(S - S_0). \tag{5.2.3}$$

Proof. An integration by parts in time and Green's formula in space applied to (3.2.11), multiplied by p and integrated over $\mathbb{R}^+ \times (t_0,T)$, yields

$$C(S_0,t_0) = \int_0^\infty \mathrm{d}S \left[C_T p(T) + \int_{t_0}^T \left(p\frac{\sigma^2 S^2}{2}\partial_S C - C\partial_S\left(\frac{\sigma^2 S^2}{2} p\right)\right)\mathrm{d}t\right], \tag{5.2.4}$$

where $C_T(S)$ is the option's premium. By the properties of C and p at zero and infinity the boundary term vanishes. Let v be the double primitive of p, *i.e.*, $\partial_{SS} v = p$. Then (5.2.3) integrated twice becomes (5.2.2) for an appropriated choice of the integration constants. □

5.2.1. Example of generalization: Dupire for binary call options

The premium is one monetary unit if $S_T > K$ and zero otherwise. It can be treated by the same method. The adjoint equation is integrated once only, and the result is

$$\begin{aligned}&u(S_0,t_0) = \int_0^\infty u_T \partial_S w \,\mathrm{d}S = -\int_0^\infty w\partial_S u_T \,\mathrm{d}S + [u_T w]_0^\infty = -w(K,T),\\ &\text{with} \quad \partial_t w - \partial_S\left(\frac{\sigma^2 S^2}{2}\partial_S w\right) + rS\partial_S w + rw = 0,\ w(t_0) = \mathbf{1}_{S>S_0} - 1.\end{aligned} \tag{5.2.5}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

5.2.2. Dupire's equation for barrier options

Consider a European barrier option which ends if $S_t \notin (S_m e^{rt}, S_M e^{rt})$. The change of variable $c(s,t) = e^{-rt}C(se^{rt},t)$ reduces (3.2.11) with barriers to

$$\partial_t c + \frac{1}{2}\sigma^2 s^2 \partial_{ss} c = 0, \quad c(s,T) = (s - Ke^{-rT})^+,$$
$$c(S_m,t) = c(S_M,t) = 0, \quad \forall s,t \in (S_m,S_M)\times(0,T). \tag{5.2.6}$$

Proposition 5.2.2. Assume that $S_m < Ke^{-rT} < S_M$. If c satisfies (5.2.6) then

$$c(s_0,t_0) = v(Ke^{-rT},T) + (S_M - Ke^{-rT})^+ \partial_S v|_{S_M,T}, \tag{5.2.7}$$

where v is the solution of

$$\partial_t v - \frac{1}{2}\sigma^2 s^2 \partial_{ss} v = 0, \quad v(s,t_0) = (s_0 - S)^+,$$
$$v(S_m,t) = (s_0 - S_m)^+, \quad v(S_M,t) = 0 \quad \forall t \in (t_0,T). \tag{5.2.8}$$

Proof. Now the boundary conditions in (5.2.3) with $r = 0$ are

$$p(S_m,t) = p(S_M,t) = 0$$

for all t. For v, the double primitive of p, it translates into $\partial_{ss} v = 0$, which in turn implies $\partial_t v = 0$, which means that v is constant at the barriers. Naturally $C(S_0,t_0)$ can be recovered from c by choosing $s_0 = S_0 e^{rt_0}$. □

5.2.3. Dupire's equation for options on Lévy-driven assets

If a Poisson–Lévy process is used in the Black–Scholes model, as in Cont and Tankov (2003), a term appears on the right-hand side of (3.2.11):

$$\int_{\mathbb{R}} \left(C(Se^y,t) - C(S,t) - S(e^y - 1)\frac{\partial C}{\partial S} \right) k(y)\,dy, \tag{5.2.9}$$

where k, the kernel of the process, is usually singular at the origin and decaying fast at infinity.

When multiplied by p and integrated in S, with $z = Se^{-y}$, this term can be transformed into an integral of $C(S,t)\chi(S,t)$ for an appropriate χ leading to the integro-differential equation for p. The resulting equation is again integrated twice and leads to the following result.

Proposition 5.2.3. A Poisson–Lévy-driven option $C_{K,T}(S,t)$, *i.e.*, a solution of (3.2.11) with (5.2.9) on the right-hand side, satisfies Dupire's identity

$$C_{K,T}(S_0,t_0) = v(K,T),$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

where v is the solution of

$$\begin{aligned}&\partial_t v - \frac{\sigma^2 S^2}{2}\partial_{SS} v + rS\partial_S v \\ &\quad - \int_{\mathbb{R}} \big(\mathrm{e}^y (v(S\mathrm{e}^{-y},t) - v(S,t)) + (\mathrm{e}^y - 1) S\partial_S v(S,t)\big) k(y)\,\mathrm{d}y = 0, \\ &v(S,t_0) = (S_0 - S)^+, \quad v(0,t) = S_0, \quad \lim_{S\to\infty} v(S,t) = 0 \quad \forall t \in (t_0, T).\end{aligned} \tag{5.2.10}$$

Theorem 5.2.4. Consider a call on S modelled by

$$\begin{aligned}&\partial_t C + \frac{\sigma^2 S^2}{2}\partial_{SS} v + rS\partial_S C - rC \\ &\quad + \int_{\mathbb{R}} \big(C(S\mathrm{e}^y,t) - C(S,t) - S(\mathrm{e}^y - 1)\partial_S C(S,t)\big)\nu(\mathrm{d}y) = 0, \\ &C(S,T) = (S - K)^+.\end{aligned} \tag{5.2.11}$$

Dupire's identity holds, *i.e.*, $C_{K,T}(S_0, 0) = v(K, T)$, when v is the solution of

$$\begin{aligned}&\partial_t v - \frac{\sigma^2 S^2}{2}\partial_{SS} v + rS\partial_S v \\ &\quad - \int_{\mathbb{R}} \big(\mathrm{e}^y (v(S\mathrm{e}^{-y},t) - v(S,t)) + (\mathrm{e}^y - 1) S\partial_S v(S,t)\big)\nu(\mathrm{d}y) = 0, \\ &v(S,0) = (S_0 - S)^+, \quad v(0,t) = S_0, \quad \lim_{S\to\infty} v(S,t) = 0 \quad \forall t \in (0, T).\end{aligned} \tag{5.2.12}$$

The proof has (5.2.11) multiplied by p and integrated in S by parts. At one stage a change of integration variable is necessary, $z = S\mathrm{e}^{-y}$. Then, as for the Black–Scholes equation, one introduces v such that $\partial_{SS} u = p$.

Now assuming that $C_i = C_{K_i,T_i}(S_0, 0)$ are market observations, then to calibrate the model one may solve

$$\min \sum_{i\in I} |v(K_i, T_i) - C_i|^2,$$

where the optimization variables are the parameters of the model, *i.e.*, $\lambda, \gamma, \sigma, \mu, \delta$ in the case of Merton's model, for example.

5.2.4. Dupire's identity in the discrete case

Extension to a more complex option is possible as long as the PDE is linear. For instance, Dupire's argument does not work with American options. Furthermore, it is not always possible to find a double primitive of the adjoint equation like (5.2.3) which, as it stands, is numerically very hard because of the singularity of the initial condition. On the other hand, at the discrete level the same difficulty is not present for the variational approach.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

For simplicity assume that the coefficients in (5.2.3) do not depend on time, and consider a time discretization with an Euler implicit scheme with time step δt, and a space discretization with piecewise linear finite elements in the variable S. The vertices of the grid are called q^i, $0 \le i \le I$, and we set $L = q^I$. The scheme can be written in matrix form,

$$(\mathbf{M} + \mathbf{A})\mathbf{u}^n - \mathbf{M}\mathbf{u}^{n+1} = 0, \tag{5.2.13}$$

where $\mathbf{u}^n$ is the vector of the nodal values of the piecewise linear function u_h^n which approximates $u(\cdot, n\delta t)$. We have

$$\begin{aligned}
\mathbf{M}_{ij} &= \frac{1}{\delta t}\int_0^L w^i w^j,\\
\mathbf{A}_{ij} &= \int_0^L \left(\partial_S\left(\frac{y^2\sigma^2}{2}w^j\right)\partial_S w^i + \frac{y^2\sigma^2}{2}w^j\partial_S w^i + r w^i w^j\right),
\end{aligned} \tag{5.2.14}$$

where the hat functions w^i are by definition piecewise linear continuous and take values 0 at q^j, $j \ne i$ and 1 at q^i. Given a vector $\mathbf{p}^0$, introduce the sequence of vectors $\mathbf{p}^n$ obtained by iterating

$$(\mathbf{A} + \mathbf{M})^T\mathbf{p}^{n+1} - \mathbf{M}^T\mathbf{p}^n = 0, \quad \mathbf{p}^0 \text{ given.} \tag{5.2.15}$$

Now notice that (5.2.13) multiplied by $(\mathbf{p}^{n+1})^T$ gives

$$\begin{aligned}
0 &= (\mathbf{p}^{n+1})^T(\mathbf{A} + \mathbf{M})\mathbf{u}^n - (\mathbf{p}^{n+1})^T\mathbf{M}\mathbf{u}^{n+1}\\
&= (\mathbf{p}^n)^T\mathbf{M}\mathbf{u}^n - (\mathbf{p}^{n+1})^T\mathbf{M}\mathbf{u}^{n+1},
\end{aligned} \tag{5.2.16}$$

where the last equality used (5.2.8). Summing up over all n gives

$$(\mathbf{p}^0)^T\mathbf{M}\mathbf{u}^0 = (\mathbf{p}^N)^T\mathbf{M}\mathbf{u}^N. \tag{5.2.17}$$

Let

$$\mathbf{p}_i^0 \text{ the solution of } \mathbf{M}^T\mathbf{p}_i^0 = \delta_{ij} \quad j = 0, \dots, I, \tag{5.2.18}$$

and let $\mathbf{p}_i^N$ be given by (5.2.15). Then (5.2.17) gives the discrete Dupire equation

$$u_i^0 = (\mathbf{p}_i^N)^T\mathbf{M}\mathbf{u}^N. \tag{5.2.19}$$

Remark 5.2.5. It is not necessary to solve for $\mathbf{p}^0$; the knowledge of $\mathbf{M}^T\mathbf{p}^0$ is sufficient to initialize (5.2.15).

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

5.3. Calibration of the local volatility

We return to the calibration of the local volatility, but now with Dupire's equation.

Knowing the spot price S_0 and some values $\{c_i\}_1^d$ of calls on S_t, we find the solution $\sigma(S,t)$ of

$$\min_{\sigma\in\mathcal{S}} \sum |u(K_i,T_i)-c_i|^2,$$

$$\text{where } \partial_t u - \frac{\sigma^2 x^2}{2}\partial_{xx}^2 u + rx\partial_x u = 0, \quad u(x,0) = (S_0-x)^+. \tag{5.3.1}$$

The problem has an obvious candidate solution. Let $\tilde{u}$ be a C^2 interpolation of $\{K_i,T_i,c_i\}_I$. Then

$$\sigma^2 = \frac{\partial_t\tilde{u} + rx\partial_x\tilde{u}}{\frac{1}{2}\partial_{xx}^2\tilde{u}} \tag{5.3.2}$$

is a solution.

Obviously these (infinitely many) solutions are not robust with respect to the data, in the sense that a small change in one c_i can cause a large change of σ. This can be fixed, however. For instance, Bézier splines are less sensitive to parameter change. Yet the main reason for not using this simple construction is that it is quite specific to the one-dimensional case. It would not work with a basket option, for instance, and it is usually not a good idea to invest numerically in a solution limited in scope.

Stabilization of (5.3.1) is achieved either by adding a Tikhonov regularization or by restricting the space $\mathcal{S}$. When the second solution is used and the optimization problem is solved by conjugate gradients, using automatic differentiation for the sensitivities (see Achdou and Pironneau (2005)), it is not that expensive if the dimension of $\mathcal{S}$ is small. For example, suppose $\sigma(x,t)$ is C^1, bilinear in an (x,t) parallelogram, grows large outside and is parametrized by eight values $\{z_k\}_{k=0}^7$, giving the following values:

$$S_{11} = z_0^2, \quad S_{21} = z_1^2, \quad S_{12} = S_{11} + z_2^2, \quad S_{22} = S_{21} + z_3^2,$$

$$\sigma_{1j} = \frac{z_{3+j}^2}{1+z_{3+j}^2}, \quad \sigma_{2j} = \frac{z_{5+j}^2}{1+z_{5+j}^2}, \quad j = 1,2. \tag{5.3.3}$$

Let $S_i,\ \sigma_i,\ i=1,2$ be the following linear functions of t:

$$S_i = S_{i1}\left(1-\frac{t}{T}\right) + S_{i2}\frac{t}{T}, \quad \sigma_i = \sigma_{i1}\left(1-\frac{t}{T}\right) + \sigma_{i2}\frac{t}{T}. \tag{5.3.4}$$

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

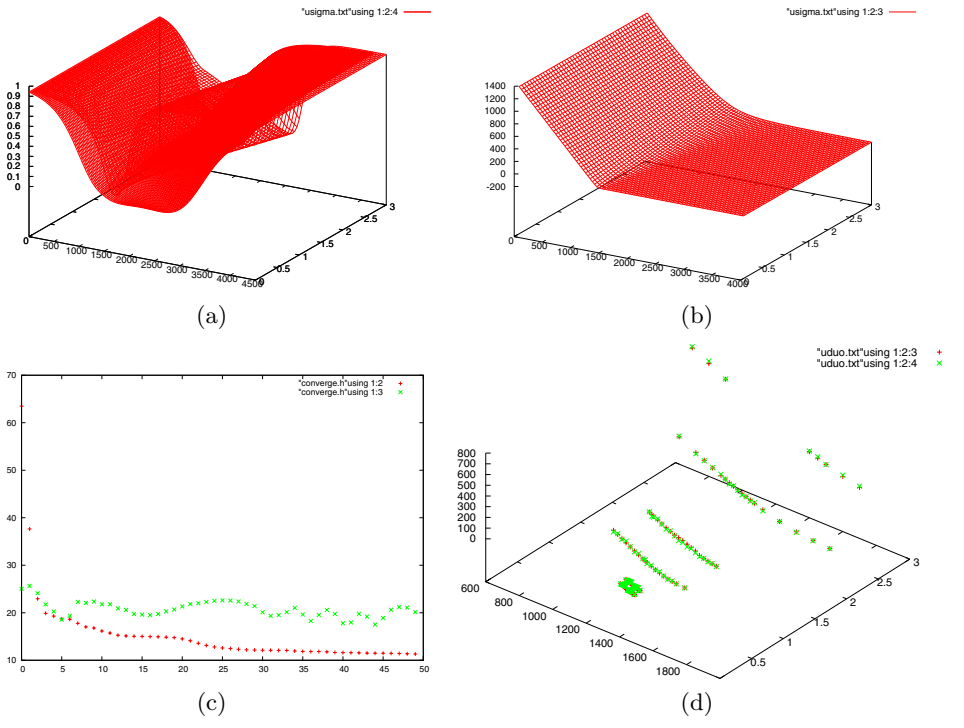


Figure 5.3.1. Calibration by least-squares and a spline parametrization of the local volatility on the SPX index of 21 Dec. 2006 at $S_0 = 1418.3, r = 3\,\%$. (a) The local volatility surface, and (b) the corresponding option price versus strike and maturity. (c) Convergence of the conjugate gradient algorithm. The functional (+) decays rapidly in less than five iterations and very slowly afterwards, while the norm of the functional gradient (×) decays erratically. (d) The data and the model's values at the same K_i, T_i points.

The volatility surface ($a = \sigma(0,t)$) is given by

$$\sigma(S,t) = \begin{cases} a + \left(2\frac{\sigma_1 - a}{S_1} - \frac{\sigma_2 - \sigma_1}{S_2 - S_1}\right)S + \left(\frac{\sigma_2 - \sigma_1}{S_2 - S_1} - \frac{\sigma_1 - a}{S_1}\right)\frac{S^2}{S_1} & \text{if } (S < S_1), \\ \sigma_2 \frac{S - S_1}{S_2 - S_1} + \sigma_1 \frac{S_2 - S}{S_2 - S_1} & \text{if } S_1 \le S \le S_2, \\ \sigma_2 + (S - S_2)\frac{\sigma_2 - \sigma_1}{S_2 - S_1} + \left((S - S_2)\frac{\sigma_2 - \sigma_1}{S_2 - S_1}\right)^2 & \text{if } S > S_2. \end{cases} \tag{5.3.5}$$

The parametrization is such that, for all values of $\{z_i\}_0^7$, σ_{ij} is positive and bounded from below and above and smooth, so no Tikhonov regularization is needed: the optimization is now done with respect to $\{z_i\}_0^7$, unconstrained.

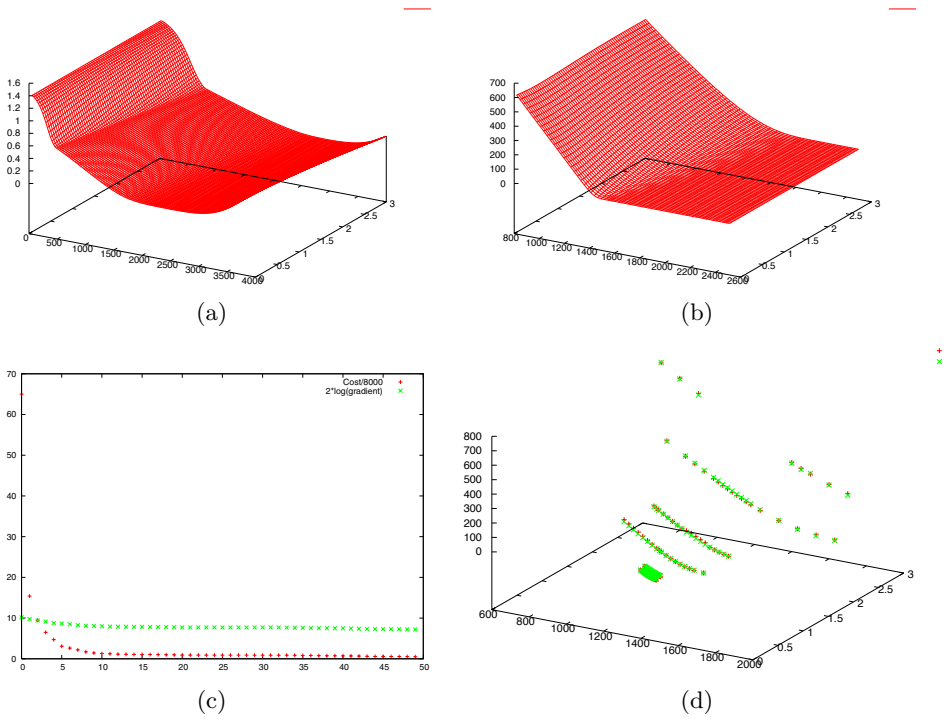
Results on the SPX index of 21 December 2006 at $S_0 = 1418.3, r = 3\,\%$ for some fifty observations are shown in Figure 5.3.1. The table of calls used is shown in Table 5.3.1.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

Table 5.3.1. The prices of a family of calls on the same asset (Eurostoxx50).

Strike	1 month	2 months	6 months	12 months	24 months	36 months
700						733
800						650.6
900						569.8
1000					467.8	
1100					385.3	
1150					345.4	
1175			265.2			
1200			242	266.1	306.6	
1215				253.4		
1225			219	245		
1250			196.6	224.2	269.2	
1275			174.5	203.9	251	
1300			152.9	184.1	233.2	
1325			131.9	164.9	215.8	
1350			111.7	146.3	198.9	
1365			100			
1375	50.6	60	92.5	139	182.6	
1380	46.1	55.8				
1385	41.8	51.8				
1390	37.5	47.9				
1395	33.4	44				
1400	29.4	40.3	74.5	128.4	166.7	215.9
1405	25.6	36.7				
1410	21.9	33.2				
1415	18.7	29.8				
1420	15.4	26.6				
1425	12.7	23.8	58	111.4	151.5	
1430	10	20.7				
1435	8	18.2				
1440	6.3	15.7				
1445	4.4	13.4				
1450	3.1	11.3	43.3	95.2	136.9	187.5
1455	2.05	9.6				
1460	1.45	7.9				
1475			30.6	80.2		
1500			20.3	54	109.6	160.8
1525			12.6	42.7		
1550			7.5	33		
1575				24.7		
1600			1.95	18.2	64.5	113.9
1700					32.7	75.7
1800					15.5	
1900					5.2	

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press



(a) (b) (c) (d)

Figure 5.4.1. The same as Figure 5.3.1, except that the implied volatility surface is the parameter of the calibration problem.

5.4. Calibration of implied volatility

Given a spot price S_0, and an option price u at $t_0 = 0$, the Black–Scholes formula can be solved for σ_I:

$$C_{BS}(S_0, r, K, T, \sigma_I) = u(K, T), \tag{5.4.1}$$

where K, T are the strike and maturity of u and r is the interest rate.

The solution is called the *implied volatility*. In practice a Newton algorithm inverts the formula in less than five iterations.

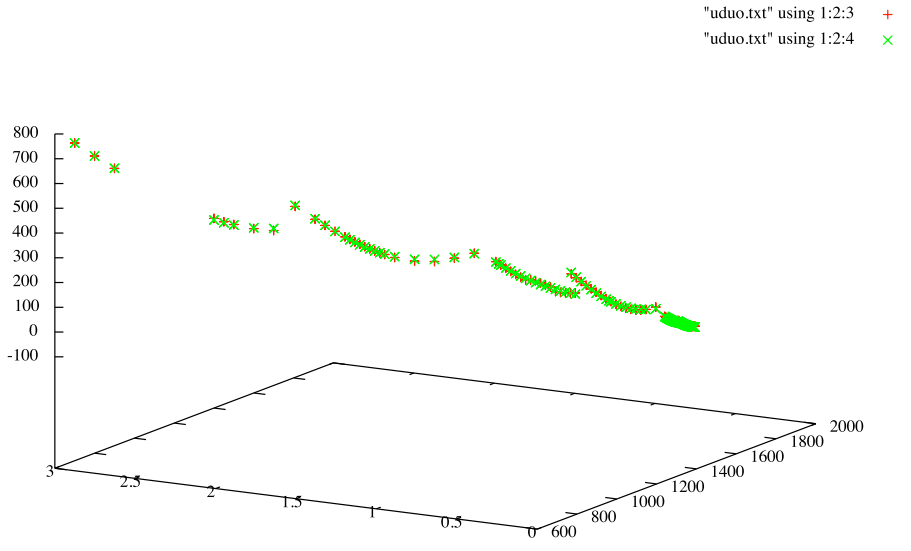
Now, to fit a set of discrete observations we solve

$$\min_{\sigma_I \in \mathcal{B}} \sum |C_{BS}(S_0, r, K_i, T_i, \sigma_I) - u_i|^2. \tag{5.4.2}$$

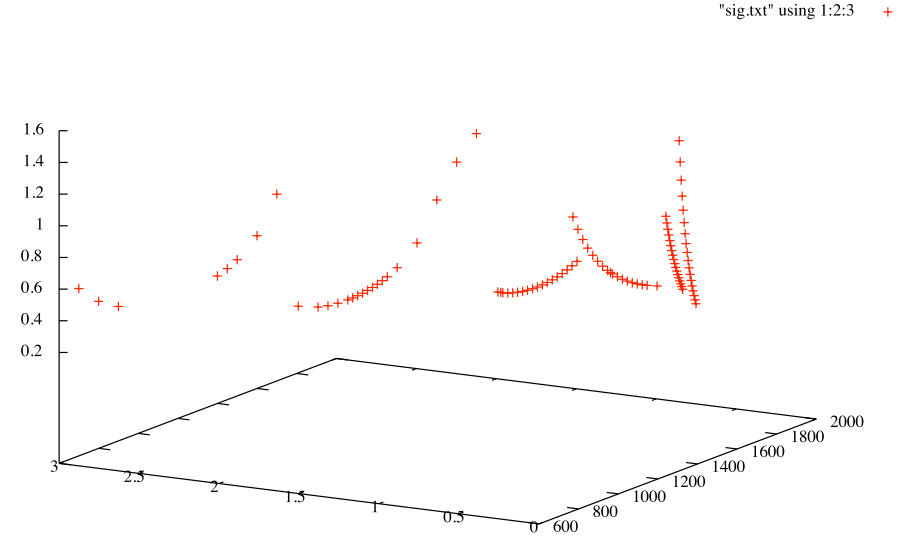
The local volatility σ_L is recovered from (5.3.2) in terms of the computed $u(K, T) := C_{BS}(S_0, r, K, T, \sigma_I)$.

The method is very fast because no PDE is solved, but it seems hard to extend to multi-dimensional cases. Results with this method are shown in Figure 5.4.1.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press



(a)



(b)

Figure 5.4.2. The same as in Figure 5.3.1 but calibrated on a reduced basis. (a) Observed prices and model predictions for the calls at the observation points; errors are hardly visible (between \$1 and \$9) and on average \$4 at each point. (b) The local volatility recovered from the Dupire equation at the observed points. For both graphs the horizontal axis is time, the vertical axis is the price of calls and the other axis is the strike.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

5.5. Calibration on a reduced basis

As before, one observes at $t = 0$ some calls $\{u_j\}_{j=1}^J$, all based on the same asset S; these have strikes K_j and maturity T_j.

Consider Dupire's equation,

$$\partial_T u_\sigma - \frac{\sigma^2}{2}\partial_{KK} u_\sigma + r\partial_K u_\sigma = 0, \quad u_\sigma(K,0) = (S-K)^+. \tag{5.5.1}$$

One wishes to choose the function $\sigma(.,.)$ to reproduce these calls. Formulated in terms of a least-squares optimization problem, we have

$$\sigma = \arg\min_\sigma \sum_{j=1}^J |u_\sigma(K_j,T_j) - u_j|^2. \tag{5.5.2}$$

When a decomposition of the general solution of (5.5.1) is used, but with K, T as variables, then (5.5.2) is a sum of independent problems at each time T_j. For each T' one solves

$$\alpha(T') = \arg\min_\alpha \sum_{j:T_j=T'} |u(K_j,T';\alpha) - u_j|^2, \quad \text{subject to}$$

$$u(K_j,T';\alpha) = u_\Sigma(K_j,T') + \sum_{i=1}^I \alpha_i \big[u_{\sigma_i}(K_j,T_M) - u_\Sigma(K_j,T_M)\big], \tag{5.5.3}$$

where $T_M = \max T_j$ is the reference time chosen to build the basis. The volatility surface is recovered from Dupire's equation and

$$u_\sigma(K,T) = u(K,T;\alpha).$$

The derivatives with respect to K are computed analytically.

The method is tested on the data of Table 5.3.1. Results are shown in Figure 5.4.2 for $r = 0.03$ and the spot price is $S_0 = 1418.3$.

PART SIX
Open-source tools

6.1. Premia

Premia (see www-rocq.inria.fr/mathfi/Premia/) is a platform and a consortium for pricing financial derivatives. Recently it has been integrated with NSP, a Matlab-like software by a member of INRIA. The C++ functions are written and maintained by the members of the consortium (including several banks and hedge funds) under the leadership of INRIA and the ENPC laboratory in France.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

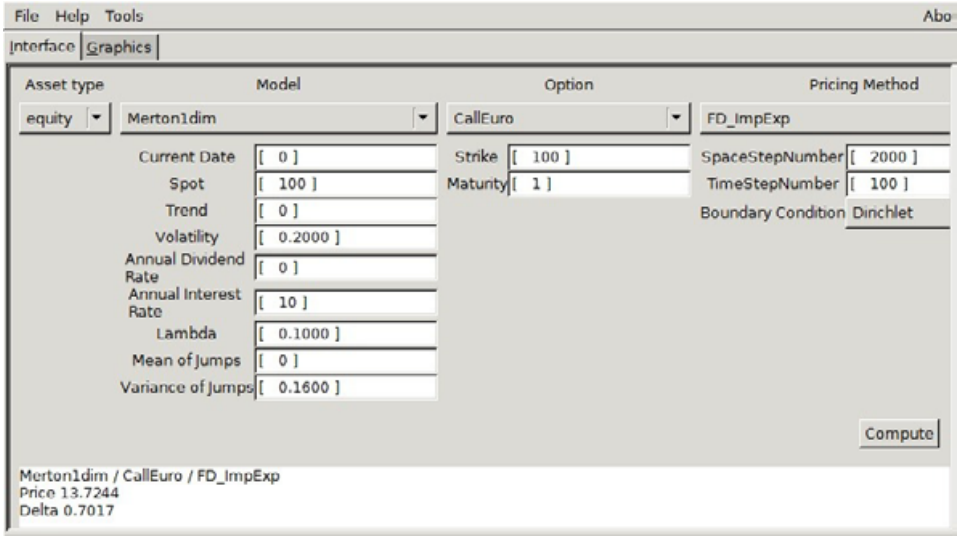


Figure 6.1.1. Under NSP, Premia is mouse/menu-driven, making the learning curve very short. This implementation runs under Windows, Linux and Mac OS X.

Premia solves the following.

Equity derivatives

- The Black–Scholes model (up to dimension 10), stochastic volatility models (Hull–White, Heston, Fouque–Papanicolaou–Sircar), models with jumps (Merton, Kou, tempered stable processes, variance gamma, normal inverse Gaussian), Bates model.
- For high-dimensional American options, Premia provides the most recent Monte Carlo algorithms: Longstaff–Schwartz, Andersen–Broadie, Barraquand–Martineau, Tsitsklis–Van Roy, Broadie–Glassermann, quantization methods, Malliavin calculus-based methods.
- Dynamic hedging for Black–Scholes and jump models is available.
- Calibration algorithms for some models with jumps, local volatility and stochastic volatility.

Interest rate derivatives

- Heath–Jarrow–Morton and LIBOR market models (LMM): affine models, Hull–White, CIR++, Black–Karasinski, squared Gaussian, Bhar–Chiarella, Li–Ritchken–Sankara–Subramanian, jump diffusion LMM, Markov functional LMM, LMM with stochastic volatility.
- Premia provides a calibration toolbox for LIBOR market models using implied volatilities for a database of swaptions and caps.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press



Figure 6.1.2. The main functions and algorithms of Premia (from www-rocq.inria.fr/mathfi/Premia).

Credit derivatives: CDSs, CDOs

- Reduced-form models and copula models are considered.
- Premia provides a toolbox for pricing CDOs using the most recent algorithms (Hull–White, Laurent–Gregory, El Karoui–Jiao, Yang–Zhang).

Hybrid products

- PDE solver for pricing derivatives on hybrid products, *e.g.*, options on inflation and interest or change rates.

Energy derivatives: swing options

- Mean reverting and jump models are considered.
- Premia provides a toolbox for pricing swing options using finite differences, the Monte Carlo Malliavin-based approach and quantization algorithms.

The main solution methods are listed in Figure 6.1.2.

6.2. QuantLib

QuantLib (see quantlib.org) is an open-source C++ library of functions and a class hierarchy to build complex finance solvers. It contains many functions to generate random numbers and variables of all types. There are also several functions for interest rate models, Black–Scholes pricers, Ornstein–Uhlenbeck, Heston, *etc.* It can be used as a base to build more complex solvers.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

6.3. FreeFem++

FreeFem++ (see www.freefem.org) is a general-purpose finite element package in the public domain to solve 2D and 3D partial differential equations. It has its own language/script, which in some sense is an extension of the C++ language. During execution the FreeFem script is interpreted and graphics are displayed on the fly; input/output can also be made using C++ keywords such as `cout<<`.

Most of the time in finance the domain of the PDE is a rectangle. It is generated by the script

```
real Lx=2, Ly=3;
mesh th = square(10,20,[Lx*x,Ly*y]);
```

It generates a uniform 10×20 triangulation (mesh) of the rectangle $(0, Lx)\times(0, Ly)$; `real,mesh,square` are keywords of the language and `Lx,Ly,th` are variables. Mesh adaptivity is built in and called by

```
mesh Th = adaptmesh(th, max(1.-x*y,0.), err=0.005);
plot(Th);
```

Here the new mesh `Th` is adapted to the function $\max(1 - x * y, 0)$ in the sense that there are more triangles in places where the Hessian of $(1-xy)^+$ is large, namely where $x * y < 1$. The optional parameter `err` controls the precision and the number of triangles generated. The result is displayed in the last line.

Piecewise polynomial finite element functions of degree one on `Th` are defined by

```
fespace Vh(Th,P1);
Vh u,v,uold=max(1.-x*y,0.);
```

Here `u`, `v` are declared only while `uold` is declared and assigned to the piecewise linear continuous interpolation of $(1-xy)^+$ on the vertices of `Th`; `u` will be used for u^{m+1} and `uold` for u^m, the values of the basket option at times to maturity $(m+1)\delta t$ and $m\delta t$.

The Black–Scholes PDE (4.3.1) for a basket with two assets x, y, discretized in time by a semi-implicit Euler scheme and P^1 continuous elements in space, is entered in variational form as

```
problem basket(u,v) = int2d(Th) ( u*v/dt
     + sigmax^2*x^2*dx(u)*dx(v)/2 + sigmay^2*y^2*dy(u)*dy(v)/2
     + sigmax*sigmay*x*y*rho*dx(u)*dy(v) )
     - int2d(Th)(uold*v/dt + sigmax*x*dx(uold)*v+sigmay*y*dy(uold)*v);
```

Here `problem,int2d,x,y,dx,dy` are keywords and `sigmax,sigmay,rho,dt` are parameters or functions which must be pre-defined.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

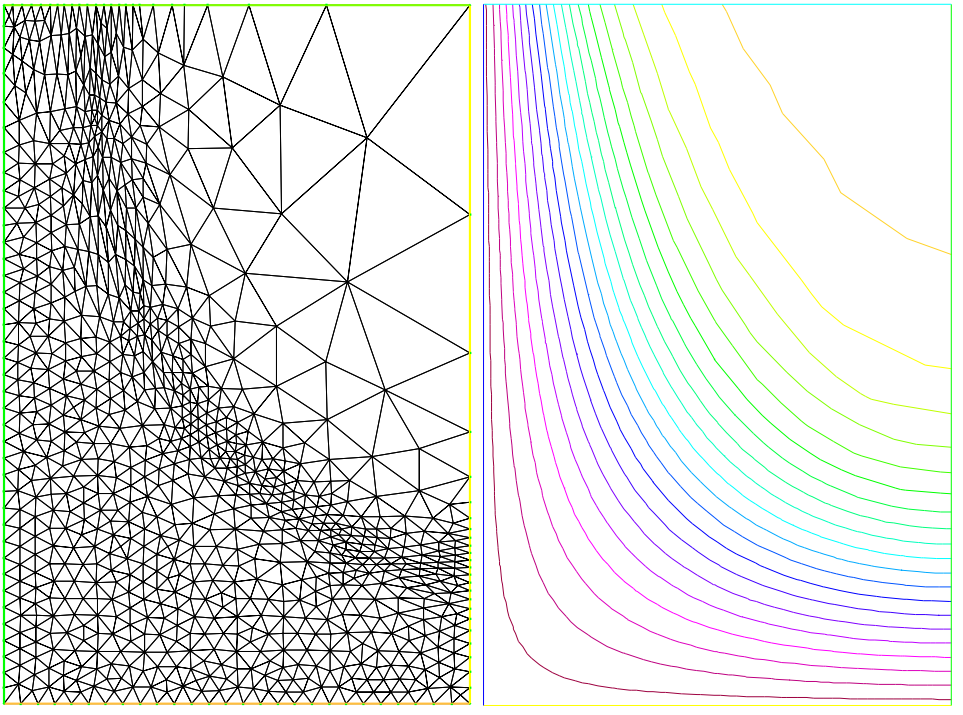


Figure 6.3.1. Result of a computation of a basket option using FreeFem++. The mesh is adapted to the pay-off $(1 - xy)^+$. The isovalues range from 1 to 0 with decrements 0.05. The parameters of the option can be read in the program below.

Finally, the time loop to compute u at maturity T is

```
for(real t=0;t<T;t+=dt){
   basket;  uold = u;
}
```

6.3.1. Two-dimensional Black–Scholes equation

The complete script and results are given below for a basket put option with multiplicative pay-off:

```
real Lx=2, Ly=3;
mesh th = square(10,20,[Lx*x,Ly*y]);
mesh Th = adaptmesh(th, max(1.-x*y,0.),err=0.005);
fespace Vh(Th,P1);
Vh v,u, uold=max(1.-x*y,0.);
plot(Th,wait=1,ps="Th.ps");
real dt=0.01, sigmax=0.5, sigmay=0.3, rho=-0.5, T=1;
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

```
problem basket(u,v) = int2d(Th) ( u*v/dt
     + sigmax^2*x^2*dx(u)*dx(v)/2 + sigmay^2*y^2*dy(u)*dy(v)/2
     + sigmax*sigmay*x*y*rho*dx(u)*dy(v) )
     - int2d(Th)(uold*v/dt - sigmax*x*dx(uold)*v - sigmay*y*dy(uold)*v);

for(real t=0;t<T;t+=dt){
   basket;  uold = u;
}
plot(u);
```

Remark 6.3.1. A large number of finite elements are built into FreeFem++, including discontinuous Galerkin methods.

FreeFem++ was not designed to solve one-dimensional problems, yet it can be done by making them 2D and invariant with respect to the added coordinate. The following is a script to compute a plain vanilla put and compare it with the Black–Scholes formula:

```
real Lx=4, Ly=0.1;
mesh Th = square(150,2,[Lx*x,Ly*y]);
fespace Vh(Th,P1);
Vh v,u, uold=max(1.-x,0.);

plot(Th,wait=1);

real dt=0.01, sigmax=0.5,  T=1;
problem basket(u,v) = int2d(Th) ( u*v/dt
     + sigmax^2*x^2*dx(u)*dx(v)/2  )
     - int2d(Th)(uold*v/dt - sigmax^2*x*dx(uold)*v) + on(2,u=0);

for(real t=0;t<T;t+=dt){
   basket;  uold = u;
}
plot(u);
ofstream fich("u.txt");
real x0=0,x1=Lx, y0=0;
for(real t=0;t<=1;t+=0.05){
real c=x0+t*x1;
real BS=0.5*c*(1+ erf(log(c)/(sigmax*sqrt(2*T)) + sigmax * sqrt(T/8.)))
      -0.5*(1+erf(log(c)/(sigmax*sqrt(2*T)) -sigmax * sqrt(T/8.)))+1-c;
    real error = u(c,y0)-BS;
fich<<c<<"   "<< u(c,y0)<<"   "<<BS<<"   "<<error<<endl;
}
// display "u.txt" by gnuplot: plot "u.txt"using 1:2 w l
// display the error by plot "error.txt"using 1:4 w l
```

Using FreeFem++ to compute a one-dimensional option requires about four times the CPU of a hand-written C++ program.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

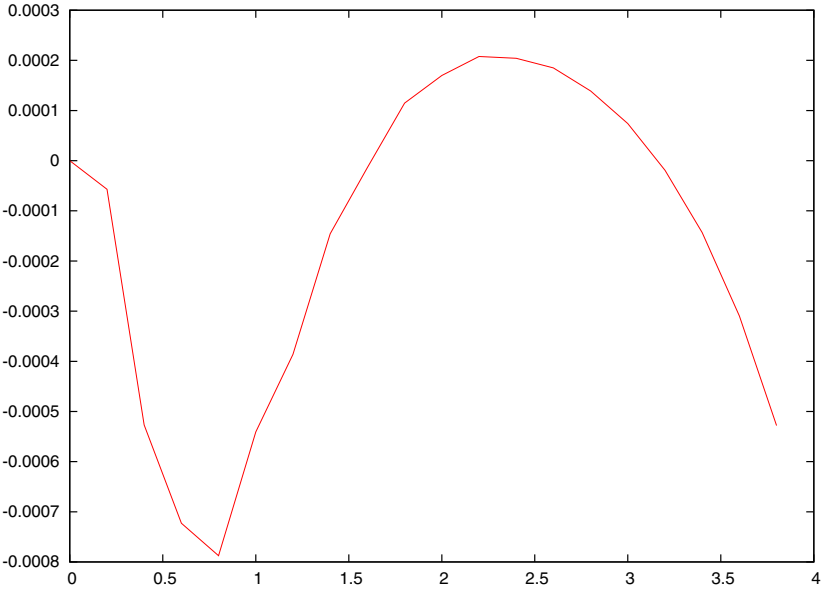


Figure 6.3.2. Result of a computation of a plain vanilla put option using FreeFem++. The plot displays the difference between the computation and the Black–Scholes analytical solution. The error is well below 0.1 %.

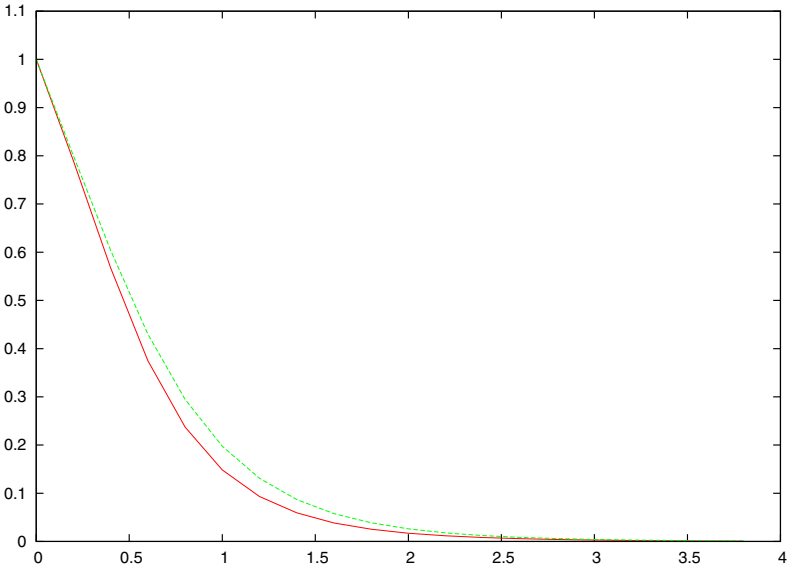


Figure 6.3.3. Computation of (one-dimensional) put with FreeFem++: comparison between a local volatility CEV with $\sigma y^{\frac{1}{4}}$, $\sigma = 0.5$ and a plain vanilla put with σ at maturity 1. The results are displayed in moneyness y. There are 150 y-points and 150 time steps.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

6.3.2. Three-dimensional basket

The same problem with three assets in the basket can be solved with FreeFem++. The script is given below for a put on a basket with linear payoff $(1 - x/2 - y/3 - z/4)^+$. A fully implicit Euler scheme is used combined with a characteristic Galerkin method (keyword `convect`). The mesh is built by extrusion of the mesh of a square. Results are shown in Figure 6.3.4.

```
load "msh3" load "tetgen" load "mshmet" load "medit"
// improve the search of a point in the mesh
//build initial mesh
int nn  = 6;
real Lx=2,Ly=3,Lz=4;
int[int] l1111=[1,2,2,1],l01=[0,1],l02=[0,2],l1122=[1,1,2,2];//   label numbering
mesh3 Th3=buildlayers(square(nn,nn,region=0,label=l1111,[x*Lx,y*Ly]),
      nn,  zbound=[0,Lz],  labelmid=l1122,   labelup = l02,  labeldown = l01);

fespace Vh(Th3,P1);
Vh u,v,uold,usol,h;

real dt = 0.02, sigmax=0.3,sigmay=0.2,sigmaz=0.1;
real sigmax2=sigmax^2/2,sigmay2=sigmay^2/2,sigmaz2=sigmaz^2/2;
real U=x*sigmax2, V=y*sigmay2, W=z*sigmaz2;

problem Poisson(u,v,solver=CG) = int3d(Th3)( u*v/dt
+ x^2*sigmax2*dx(u)*dx(v) + y^2*sigmay2*dy(u)*dy(v) + z^2*sigmaz2*dz(u)*dz(v) )
    - int3d(Th3)( convect([U,V,W],-dt,uold)*v/dt) + on(2,u=0);

real errm=1e-3;// level of error for mesh adaptivity
int iimax=2;
for(int ii=0; ii<=iimax; ii++){ // iterations of mesh adaption
   uold = max(0., 1.-x/2.-y/3.-z/4.);
for(int k=0;k<50;k++){
Poisson;
uold=u;
}
  cout <<" u min, max = " <<  u[].min << " "<< u[].max << endl;
  h=0. ;// for resizing h
  h[]=mshmet(Th3,u,normalization=1,aniso=0,nbregul=1,hmin=1e-3,hmax=0.3,err=errm);
  cout <<" nb of vertices " << Th3.nv << endl;
  plot(u,wait=0,nbiso=15);
  errm *= 0.5;// change the level of error
  cout << " Th3" << Th3.nv < " " << Th3.nt << endl;
  if(ii != iimax) Th3=tetgreconstruction(Th3,switch="raAQ",sizeofvolume=h*h*h/6.);
}
medit("U-adap-iso",Th3,u,wait=1); // for interactive graphics
ofstream fich("u.txt");          // for gnuplot display by plot"u.txt"using 1:2 w l
real x0=0, y0=0, z0=0, x1=Lx, y1=Ly,z1=Lz; // the cube diagonal
for(real t=0;t<=1;t+=0.05)
fich<<t<<"   "<< u(x0+t*x1,y0+t*y1,z0+t*z1)<<endl;

// CPU= 144" end with NUMBER OF GIVEN VERTICES      12156
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

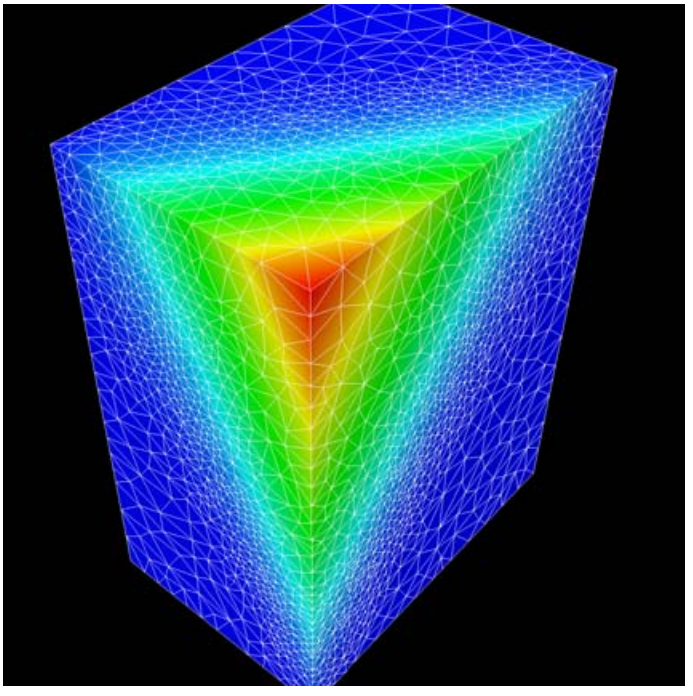


Figure 6.3.4. Put option on a basket of three assets computed via FreeFem++.

6.4. CUDA and other tools

Several programming environments provide tools such as Matlab and Mathematica for option pricing. The open-source project Sage will have several too (see www.sagemath.org/doc/reference/finance.html). An example with the Black–Scholes formula is displayed in Figure 6.4.1.

CUDA is an add-on language to C++, and is the property of NVIDIA corporation, but free: using CUDA, the programmer can make sure that part of the program runs on NVIDIA's GPU (graphic processor unit), when available (see also OpenCL). GPUs have many computing units, though less powerful than Intel's i7, for instance. In finance a speed up of 100 with GPUs is not rare. We give here a very simple example for illustration.

Given a random number generator to compute realizations (n) of a normal random variable X, the aim is to compute

$$C = \frac{\mathrm{e}^{-rT}}{N} \sum_n \max(S_T^{(n)} - K)^+, \quad \text{with } S_T^{(n)} = S_0 \mathrm{e}^{(r - \frac{\sigma^2}{2})T + \sigma\sqrt{T}X^{(n)}},$$

when N is large, and accelerate the computation with GPU.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

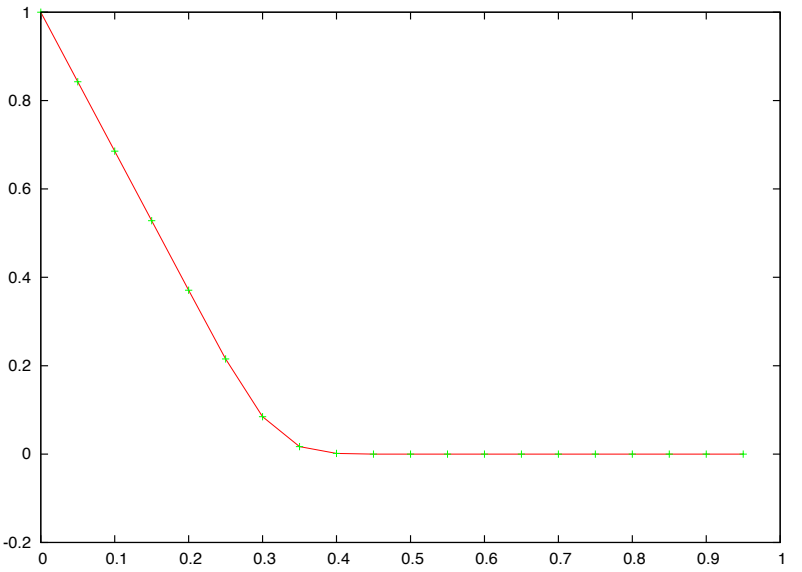


Figure 6.4.1. Put option computed by using Sage.

First we need a random number generator that can execute on the GPU (called __device__ in CUDA) and some initialization constants:

```
#define PI 3.14159265358979323846264338327950288f
#define IA 16807
#define IM 2147483647
#define AM (float)(1.0/IM)
#define IQ 127773
#define IR 2836
#define MASK 123459876

const int NbBlocs = 50000;
const int NbThreads = 500;
const long N  = NbBlocs*NbThreads;
const float  K = 220, S0=100, r=0.02f, sig=0.3f, T=1.0f;
const float R = (r-sig*sig/2)*T;
const float q = 0;

__device__
void ran0(long*idum, float *res)
{
   long k;
   float ans;
   *idum^=MASK;
   k=(*idum)/IQ;
   *idum=IA*(*idum-k*IQ)-IR*k;
   if(*idum<0)*idum+=IM;
   ans=AM*(*idum);
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

```
   *idum^=MASK;
   *res = ans;
}
```

Then we shall use the Box–Muller formula. The following function will be run on the GPU and the CPU for comparison:

```
__host__ __device__
void BS(float *x, float *y){
   float xx, yy;   // Box-Muller formula for normal r.v.
   float u1 = *x, u2 = *y;
   xx = sqrtf(-2.0f * logf(u1)) * cosf(2.0f * PI * (u2));
   *y = S0*expf(R+sig*sqrtf(T)*xx);   // S_T
   yy = sqrtf(-2.0f * logf(u1)) * sinf(2.0f * PI * (u2));
   *x = S0*expf(R+sig*sqrtf(T)*(yy + q * xx));  // S_T
}
```

On input it takes two independent random variables x and y, uniformly distributed on $[0,1]$; on output it returns two values of $S_T^{(n)}$. In standard C, given two arrays `a1,a2`, the task would be achieved by a call to `BScpu` and calls to the C-function `rand()`:

```
void BScpu(float *a1, float *a2, int I) {
    srandom(time(NULL));
   for ( int n = 0 ; n < N ; ++n ){            // fills a1,a2
     a1[n] = (rand() + 0.5f)/(RAND_MAX + 1.0f);// with random
     a2[n] = (rand() + 0.5f)/(RAND_MAX + 1.0f);// nb in (0,1)
   for ( int i = 0 ; i < I ; ++i )
     BS(a1+i, a2+i);   // same as BS(&a1[i],&a2[i])
}
```

The same can be done on the GPU by:

```
__global__
void BSgpu(float *a1, float *a2, long *idum, long N) {
   int i = blockDim.x*blockIdx.x + threadIdx.x;
   if ( i < N ) {
     idum[i] = -1-i;     // used as seeds for the random gen
     ran0(idum+i,a1+i);  // fills array a1 with r.v.
     ran0(idum+i,a2+i);  // fills array a2 with r.v.
     BS(a1+i, a2+i);
   }
}
```

However, arrays `a1,a2` must be allocated on the GPU (by `CUDAMALLOC`), and after the computations the results must be sent back to the CPU (by `cudaMemcpy`). Finally, the put is computed by a reduction into a sum of all the values in `a1,a2` (here A1 and A2):

```
   cudaMalloc( (void **) &B1, taille);   // allocate B in GPU RAM
   cudaMalloc( (void **) &B2, taille);   // allocate B in GPU RAM
   cudaMalloc( (void **) &idum, taillelong);
```

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

```
BSgpu<<<NbBlocs, NbThreads>>>(B1,B2,idum, N);
cudaMemcpy(A1, B1, taille, cudaMemcpyDeviceToHost); // B1 -> A1
cudaMemcpy(A2, B2, taille, cudaMemcpyDeviceToHost);
float gput=0;
for ( int n = 0 ; n < N ; ++n )
  gput += fmax(K-A1[n]-A2[n],0.0f);
```

Notice the peculiar syntax to call `BSgpu()`. The modifiers `NbBlocs` and `NbThreads` influence the computing time on the GPU because they specify how the data are arranged in blocks and the number of threads.

Conclusion

We hope to have convinced the reader that computing financial assets by deterministic methods, in particular by solving PDEs, PDIs and PIDEs, can have advantages: certainly precision, maybe speed, and a sound treatment of singularities if any. Yet many pricing problems, such as complex contracts with many discrete dates, are hard to formulate in terms of partial differential equations.

In computational fluid dynamics, it took many years to convince aeroplane manufacturers to use variational methods instead of finite difference methods. The difficulty was in training engineers in this new field but the advantage was that an entire aeroplane could be discretized by an arbitrary mesh instead of the painful task of matching uniform blocks, as needed by finite difference methods. Here in finance the domain of integration is simple, usually $(0, L_1) \times \cdots \times (0, L_d)$, and so the advantage of variational methods is not so much in the mesh but rather in the *optimal* mesh: a mesh where the points are where they are needed for precision and for resolving singularities, if any.

Parallelism continues to grow on the hardware side, and certainly it is in favour of Monte Carlo methods; variational methods can be parallelized as well, but are more complex and much harder to implement.

In the end, the method which provides the best ratio of speed over precision will win, and the result will only be known after several years of numerical experimentation. Meanwhile innovation will perfect all methods: it is just the beginning!

Acknowledgements

Because of the rules of the game, this article contains some of my own research. I would like to thank my research collaborators, Yves Achdou, Rama Cont, Nicolas Dicesare, Frédéric Hecht, Grégoire Loeper and Nicolas Lantos.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

REFERENCES[4]

Y. Achdou and O. Pironneau (2005), *Computational Methods for Option Pricing*, Vol. 30 of *Frontiers in Applied Mathematics*, SIAM.

Y. Achdou and O. Pironneau (2009), *Mathematical Modeling and Numerical Methods in Finance*, Vol. XV of *Handbook of Numerical Analysis* (P. G. Ciarlet *et al.*, eds), North-Holland.

Y. Achdou and N. Tchou (2002), 'Variational analysis for the Black and Scholes equation with stochastic volatility', *M2AN: Math. Model. Numer. Anal.* **36**, 373–395.

H. Albrecher, P. Mayer, W. Schoutens, and J. Tistaert (2007), 'The little Heston trap', *Wilmott Magazine*, January 2007, 83–91.

L. Andersen and J. Andreasen (2000), 'Jump diffusion models: Volatility smile fitting and numerical methods for pricing', *Review of Derivatives Research* **4**, 231–262.

A. Ang and F. Longstaff (2011), Systemic sovereign credit risk: Lessons from the U.S. and Europe. NBER Working Paper No. 16982, April 2011. Also available at www.scribd.com/doc/55649615/7/MODELING-CDS-SPREADS-WITH-SYSTEMIC-RISK.

E. Ayache, P. A. Forsyth and K. R. Vetzal (2003), 'The valuation of convertible bonds with credit risk', *J. Derivatives* **11**, 9–29.

L. Bachelier (1900), 'Théorie de la spéculation', *Annales Scientifiques de l'Ecole Normale Supérieure* **3**, #17.

G. Barles (1997), Convergence of numerical schemes for degenerate parabolic equations arising in finance. In *Numerical Methods in Finance* (L. C. G. Rogers and D. Talay, eds), Cambridge University Press, pp. 1–21.

A. Bentata and R. Cont (2012), 'Forward equations for option prices in semimartingale models', *Finance & Stochastics*, to appear.

J. Bertoin (1996), *Lévy Processes*, Cambridge University Press, Cambridge.

F. Bervoets (2006), Model risk for exotic equity options. Master's thesis, Delft University of Technology, Netherlands.

F. Black and M. Scholes (1973), 'The pricing of options and corporate liabilities', *J. Political Economy* **81**, 637–654.

D. Brigo and F. Mercurio (2002), 'Lognormal-mixture dynamics and calibration to market volatility smiles', *Internat. J. Theoret. Appl. Finance* **5** 427–446.

D. Brigo and F. Mercurio (2007), *Interest Rate Models: Theory and Practice. With Smile, Inflation and Credit*, second edition, Springer Finance.

P. Carr and D. B. Madan (1999), 'Option valuation using the fast Fourier transform', *J. Comput. Finance* **2**, 61–73.

P. Carr, H. Geman, D. Madan and M. Yor (2002), 'The fine structure of asset returns: An empirical investigation', *J. Business.* **75**, 305–332.

B. Chen, L. A. Grzelak and C. W. Oosterlee (2011), Calibration of a stochastic interest rate SAB. In *Proc. Quantitative Methods in Financial and Insurance Mathematics*, Lorentz Center, Leiden, the Netherlands. To appear.

[4] The URLs cited in this work were correct at the time of going to press, but the publisher and the authors make no undertaking that the citations remain live or are accurate or appropriate.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

E. Clément, D. Lamberton and P. Protter (2002), 'An analysis of a least square regression method for American option pricing', *Finance & Stochastics* **6**, 449–471.

R. Cont and P. Tankov (2003), *Financial Modelling with Jump Processes*, Chapman & Hall.

R. Cont and E. Voltchkova (2004), Integro-differential equations for option prices in exponential Lévy models. CMAPX report R.I.547, Ecole Polytechnique.

R. Cont and E. Voltchkova (2005), 'A finite difference scheme for option pricing in jump diffusion and exponential Lévy models', *SIAM J. Numer. Anal.* **43**, 1596–1626.

R. Cont, N. Lantos and O. Pironneau (2011), 'A reduced basis for option pricing', *SIAM J. Math. Finance* **2**, 287–316.

J. Cox (1975), Notes on option pricing I: Constant elasticity of variance diffusions. Working Paper, Stanford University.

J. Cox, J. Ingersoll and S. Ross (1985), 'A theory of the term structure of interest rates', *Econometrica* **53**, 389–408.

S. Crépey (2003), 'Calibration of the local volatility in a generalized Black–Scholes model using Tikhonov regularization', *SIAM J. Math. Anal.* **34**, 1183–1206.

A. Dervieux (1996), About the basic methods. In *Handbook of Computational Fluid Dynamics* (R. Peyret, ed.), Academic Press, pp. 1–23.

B. Dupire (1997), Pricing and hedging with smiles. In *Mathematics of Derivative Securities* (M. Dempster and S. Pliska, eds), Cambridge University Press, pp. 103–111.

N. El Karoui and E. Gobet (2011), *Les Outils Stochastiques des Marchés Financiers: Une Visite Guidée de Einstein à Black–Scholes*, Ecole Polytechnique.

W. Farkas, N. Reich and C. Schwab (2006), Anisotropic stable Lévy copula processes? Analytical and numerical aspects. ETH Research Report No. 2006-08, Zurich.

P.-L. George and H. Borouchaki (1998), *Delaunay Triangulation and Meshing: Application to Finite Elements* (trans. P. J. Frey and S. A. Canann), Editions Hermès.

T. Gerstner and M. Griebel (1998), 'Numerical integration using sparse grids', *Numer. Algorithms* **18**, 209–232.

M. Giles (2008), 'Multilevel Monte Carlo path simulation', *Operations Research* **56**, 981–986.

P. Glasserman (2003), *Monte Carlo Methods in Financial Engineering*, Vol. 53 of *Stochastic Modelling and Applied Probability*, Springer.

F. Hecht, O. Pironneau, A. Le Hyaric and K. Otsuka (2008), The FreeFem++ documentation. UPMC Publications. See also www.freefem.org.

S. L. Heston (1993), 'A closed-form solution for options with stochastic volatility with application to bond and currency options', *Review of Financial Studies* **6**, 327–343.

M. Hintermüller, V. A. Kovtunenko and K. Kunisch (2004), 'Semismooth Newton methods for a class of unilaterally constrained variational problems', *Adv. Math. Sci. Appl.* **14**, 513–535.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

J. C. Hull (2003), *Options, Futures and Other Derivatives*, fifth edition, Prentice-Hall.
J. C. Hull and A. White (1987), 'The pricing of options on assets with stochastic volatilities', *J. Finance* **42**, 281–300.
S. Ikonen and J. Toivanen (2008), 'Efficient numerical methods for pricing American options under stochastic volatility', *Numer. Methods Partial Diff. Equations* **24**, 104–126.
S. Ikonen and J. Toivanen (2009). 'Operator splitting methods for pricing American options under stochastic volatility', *Numer. Math.* **113**, 299–324.
K. Ito and K. Kunisch (2006), 'Parabolic variational inequalities: The Lagrange multiplier approach', *J. Math. Pures Appl.* **85**, 415–449.
K. R. Jackson, S. Jaimungal, and V. Surkov (2008), 'Fourier space time-stepping for option pricing with Lévy models', *J. Comput. Finance* **12**, 1–29.
L. Jiang and M. Dai (1999), Convergence of binomial tree method for American options. In *Partial Differential Equations and their Applications* (C. Hua and L. Rodino, eds), World Scientific, pp. 106–108.
M. S. Joshi (2011), *More Mathematical Finance*, Pilot Whale Press, Melbourne.
A. Kolmogoroff (1931), 'Über die analytischen Methoden in der Wahrscheinlichkeitsrechnung', *Math. Annalen* **104**, 415–458.
D. Lamberton and B. Lapeyre (1997), *Introduction au Calcul Stochastique Appliqué à la Finance*, Ellipses.
B. Lapeyre, C. Martini and A. Sulem (2010), The Premia consortium and software. www.premia.fr.
A. Lewis (2001), A simple option formula for general jump-diffusion and other exponential Lévy processes. www.optioncity.net.
P.-L. Lions and J.-M. Lasry (2007), 'Mean field games', *Japan. J. Math.* **2**, 229–260.
F. A. Longstaff and E. S. Schwartz (1992), 'Interest rate volatility and the term structure: A two-factor general equilibrium model', *J. Finance* **47**, 1259–1282.
R. Lord, F. Fang, F. Bervoets and C. W. Oosterlee (2007), A fast and accurate FFT-based method for pricing early-exercise options under Lévy processes. Available at SSRN: ssrn.com/abstract=966046
A.-M. Matache, T. von Petersdoff and C. Schwab (2004), 'Fast deterministic pricing of Lévy driven assets', *J. Math. Model. Numer. Anal.* **38**, 37–72.
R. C. Merton (1976), 'Option pricing when underlying stock returns are discontinuous', *J. Financial Econ.* **3**, 125–144.
S. Mikhailov and U. Nögel (2003), Heston's stochastic volatility model implementation, calibration and some extensions. *Wilmott Magazine*, July 2003.
B. Øksendal (1998), *Stochastic Differential Equations*, Springer.
G. Pagès (2008), Méthodes de Monte-Carlo. Notes de cours Master 2, Université Pierre et Marie Curie. www.proba.jussieu.fr/pageperso/pages/
O. Pironneau (2009), 'Calibration of options on a reduced basis', *J. Comput. Appl. Math.* **232**, 139–147.
W. H. Press, S. A. Teukolsky, W. T. Vettering and B. P. Flannery (1992), *Numerical Recipes in C*, second edition, Cambridge University Press.
C. Prud'homme, D. Rovas, K. Veroy, Y. Maday, A. T. Patera and G. Turinici (2002), 'Reliable real-time solution of parametrized partial differential equations: Reduced-basis output bound methods', *J. Fluid Engng* **124**, 70–80.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press

R. Rannacher (1984), 'Finite element solution of diffusion problems with irregular data', *Numer. Math.* **43**, 309–327.

R. D. Richtmyer and Morton (1967), *Difference Methods for Initial-Value Problems*, second edition, Wiley–Interscience.

E. W. Sachs and M. Schu (2008), Reduced order models (POD) for calibration problems in finance. In *Numerical Mathematics and Advanced Applications: Proc. ENUMATH 2007* (K. Kunisch, G. Of, and O. Steinbach, eds), Springer, pp. 735–742.

M. Schroder (1989), 'Computing the Constant Elasticity of Variance option pricing formula', *J. Finance* **44**, 211–219.

S. A. Smolyak (1963), 'Quadrature and interpolation formulas for tensor products of certain classes of functions', *Dokl. Akad. Nauk. USSR* **4**, 240–243.

E. Stein and J. Stein (1991), 'Stock price distributions with stochastic volatility: An analytic approach', *Rev. Financ. Stud.* **4**, 727–752.

D. Talay and L. Tubaro (1996), *Probabilistic Models for Nonlinear Partial Differential Equations*, Vol. 1627 of *Lecture Notes in Mathematics*, Springer.

P. Tankov and E. Voltchkova (2009), Jump-diffusion models: A practitioner's guide. *Banque et Marchés*, no. 99.

P. Wilmott, S. Howison, and J. Dewynne (1995), *The Mathematics of Financial Derivatives: A Student Introduction*, Cambridge University Press.

Xiaoye S. Li, J. W. Demmel and J. R. Gilbert (1999), The SuperLU library. crd.lbl.gov/∼xiaoye/SuperLU.

Xuan Luo (2007), Bermudan swaption hedging in Black–Karasinski model. PhD thesis, New York University.

https://doi.org/10.1017/S0962492912000074 Published online by Cambridge University Press