Recursive Marginal Quantization: Extensions and Applications in Finance

Ralph Rudd

February 2018

Thesis Presented for the Degree of
Doctor of Philosophy
in the Department of
The African Institute of Financial Markets and Risk Management

Abstract

Quantization techniques have been used in many challenging finance applications, including pricing claims with path dependence and early exercise features, stochastic optimal control, filtering problems and the efficient calibration of large derivative books. Recursive marginal quantization of an Euler scheme has recently been proposed as an efficient numerical method for evaluating functionals of solutions of stochastic differential equations.

This algorithm is generalized and it is shown that it is possible to perform recursive marginal quantization for two higher-order schemes: the Milstein scheme and a simplified weak-order 2.0 scheme. Furthermore, the recursive marginal quantization algorithm is extended by showing how absorption and reflection at the zero boundary may be incorporated.

Numerical evidence is provided of the improved weak-order convergence and computational efficiency for the geometric Brownian motion and constant elasticity of variance models by pricing European, Bermudan and barrier options. The current theoretical error bound is extended to apply to the proposed higher-order methods.

When applied to two-factor models, recursive marginal quantization becomes computationally inefficient as the optimization problem usually requires stochastic methods, for example, the randomized Lloyd's algorithm or Competitive Learning Vector Quantization. To address this, a new algorithm is proposed that allows recursive marginal quantization to be applied to two-factor stochastic volatility models while retaining the efficiency of the original Newton-Raphson gradient-descent technique. The proposed method is illustrated for European options on the Heston and Stein-Stein models and for various exotic options on the popular SABR model.

Finally, the recursive marginal quantization algorithm, and improvements, are applied outside the traditional risk-neutral pricing framework by pricing long-dated contracts using the benchmark approach. The growth-optimal portfolio, the central object of the benchmark approach, is modelled using the time-dependent constant elasticity of variance model. Analytic European option prices are derived that generalize the current formulae in the literature. The time-dependent constant elasticity of variance model is then combined with a 3/2 stochastic short rate model to price zero-coupon bonds and zero-coupon bond options, thereby showing the departure from risk-neutral pricing.

Dedication

This thesis is dedicated to my parents, Ralph H. and Hanlie Rudd.

Acknowledgements

I would like to thank my co-supervisors, Professors Jörg Kienitz and Eckhard Platen, for their guidance, advice and patience. I am also grateful to Professors Kienitz and Platen for graciously accommodating me when I visited them during my doctoral studies.

I would like to thank Professor David Taylor for his resilience and fortitude while he was facilitating my degree.

I would like to thank Professors Martin Larsson, Erik Schlögl and Rodrigo Targino for their supervision during the past three years of the annual Financial Mathematics Team Challenge. Each year made me a better researcher.

Parts of this work have been presented at conferences all over the world and I would like to thank all the participants who asked questions, pointed out errors and contributed ideas.

I would like to thank Rand Merchant Bank, BankSETA and the African Institute of Financial Markets and Risk Management for their financial support during my doctoral studies. Without it, this work would not have been completed.

Finally, I would like to thank my primary supervisor, Professor Thomas McWalter:

Thanks, Tom. For everything.

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview

Quantization is a lossy compression technique that produces a discrete representation of a signal using less information than the original. The technique originated in the field of signal compression, but has found application in fields as far-reaching as image processing [Heckbert, 1982], clustering methods in data mining [Jain and Dubes, 1988], integration theory [Pagès, 1998], and numerical probability [Graf and Luschgy, 2000]. Recently, Pagès and Sagna [2015] proposed the recursive marginal quantization (RMQ) of an Euler scheme as an efficient numerical method for approximating functionals of solutions to stochastic differential equations (SDEs).

The aim of this thesis is three-fold: Firstly, to effectively extend the RMQ algorithm to both higher dimensions and higher-order discretization schemes. Secondly, to apply the extended RMQ algorithm to several of the popular and challenging models in mathematical finance, e.g., the Heston [1993] and SABR [Hagan et al., 2002] models, with specific emphasis on option pricing and calibration. And finally, to increase the adoption of quantization techniques in finance by providing a concise, clear and accessible guide to both the essential mathematics and the commmon numerical techniques.

To emphasize the importance of numerical methods for solving SDEs in finance, traditional arbitrage pricing theory is briefly reviewed. Following Glasserman [2003, Sec. 1.2], three sequential concepts can be highlighted:

1. If the payoff of a derivative security can be replicated, i.e., perfectly reproduced, by a self-financing portfolio strategy that trades in other assets, then the arbitrage-free cost of the derivative must be the cost of the trading strategy.

2. If there exists a probability measure, associated with a discount factor or numeraire, such that all discounted tradeable assets are martingales under this measure, then the prices of those derivative contracts that can be replicated are the expectations of their numeraire-denominated payoffs under this martingale measure.

3. If, for a given choice of discount factor, the martingale measure is unique, then every sufficiently regular payoff can be replicated. Such a market is complete. In an incomplete market, derivative claims exist that cannot be perfectly replicated, i.e., their prices are not completely determined by the prices of other tradeable assets.

The mathematical foundations of derivative pricing are both practical and theoretically well-developed. Although these three concepts are simple, to make a sufficiently rigorous presentation

of the accompanying Fundamental Theorems of Asset Pricing requires, at least, a book-length treatment on stochastic process and measure theory, see Delbaen and Schachermayer [2006].

Now consider an asset, $X = \{X_t,\ t \geq 0\}$, modelled by the Itô process

$$X_t = X_0 + \int_0^t a(X_s)\, ds + \int_0^t b(X_s)\, dW_s,$$

with the associated differential form

$$dX_t = a(X_t)\, dt + b(X_t)\, dW_t, \tag{1.1}$$

and the coefficients a and b sufficiently bounded to ensure the existence of a unique strong solution.

From the second basic idea highlighted above, computing the price of a derivative contract written on this underlying asset is equivalent to computing the discounted expectation of a functional of the solution to the SDE, under the correct measure. However, SDEs like (1.1) rarely have explicit solutions[1]. Instead, numerical methods must be relied upon.

Monte Carlo methods are a popular and widely applicable class of techniques that simulate sample paths of time-discrete approximations to the SDE. Using a large-scale simulation of different paths, various statistical properties of the solution can be estimated, such as the desired expectation. The advantage of these methods is that their computational costs, in terms of both time and memory, increase only polynomially with the dimension of the process [Glasserman, 2003]. However, not all heuristic time-discrete approximations of an SDE will converge in a useful sense as the maximum time-step size goes to zero [Kloeden and Platen, 1999].

Consider the time-discretization grid $t_0 = 0 < t_1 < \cdots < t_n = T$, where n is the number of discrete steps taken to reach the finite time horizon, $T > 0$. The size of each step is $\Delta t = \frac{T}{n}$. Now a step-by-step approximation to the path of (1.1) can be created with values at each of the discrete time points, t_k, for $k = 0, \ldots, n$.

The simplest scheme that exhibits useful convergence properties is the Euler-Maruyama [Maruyama, 1955] scheme, given by

$$\overline{X}_{k+1} = \overline{X}_k + a(\overline{X}_k)\Delta t + b(\overline{X}_k)\sqrt{\Delta t}\Delta W_k, \tag{1.2}$$

where $\overline{X}$ approximates X on the discrete time grid, and the increment of Brownian motion, ΔW_k, is simulated by a Gaussian-distributed pseudo-random number with mean zero and variance Δt. For more details on random number generation, see Niederreiter [1992]. It should be immediately clear that the approximation above is formed by using left-endpoint approximations to the integrands of (1.1).

An alternative to Monte Carlo methods is instead to discretize both time and space, such that the approximating processes are finite-state Markov chains, characterized by their transition probability matrices, see, e.g., Kushner and Dupuis [2001]. Quantization methods fall into this latter category.

The vector, or finite-dimensional, quantization of probability distributions was formalized in the seminal work of Graf and Luschgy [2000] and has been applied to the field of mathematical finance since its inception. It is a technique for optimally representing a continuous distribution by a discrete distribution, where a measure of the 'distance' between the two, called the distortion, is minimized. The distortion is most commonly specified using the squared Euclidean error.

The application of vector quantization to the solution of finance-related problems generally proceeds by discretising time and then quantizing the corresponding marginal distributions of the sys-

[1] Notable SDEs with explicit solutions are detailed in Kloeden and Platen [1999, Chap. 4]

tem of SDEs specific to the problem. The quantized grids and their associated probability weights are then used to compute the expectations required in pricing contingent claims. Quantization-based algorithms have been proposed for contingent claims with both path-dependency and early exercise features, e.g., multi-dimensional American options [Bally et al., 2005; Pagès and Wilbertz, 2009], swing options [Bardou et al., 2009], barrier options [Sagna, 2011], and Asian options [Bormetti et al., 2017].

The application of functional, or infinite-dimensional, quantization to financial problems is usually based on the Karhunen-Loéve expansion of Gaussian processes and is explored in Pagès and Printems [2005], Pagès [2008b] and, more recently, Corlay and Pagès [2015].

For a thorough historical overview of the development of quantization techniques across disciplines see Gray and Neuhoff [1998]. Review papers that specifically concern the application of quantization to finance include Pagès et al. [2004], Pagès and Printems [2009] and Pagès [2014].

The classical applications of quantization techniques generally require solving intensive minimization problems, which incur a heavy computational burden. There has been a recent resurgence of interest in quantization due to the development of RMQ by Pagès and Sagna [2015], an efficient approach for the single-factor case.

RMQ makes use of a Newton-Raphson iteration to quantize the Euler-Maruyama updates (1.2) of the underlying SDE (1.1). This technique has been shown to be effective for the fast calibration of large derivative books [Callegaro et al., 2015, 2016], another challenging application in finance, and extended for use with two-factor SDEs and applied to stochastic volatility models [Callegaro et al., 2016; Fiorin et al., 2017].

This thesis extends the RMQ algorithm in several important ways. In Chapter 3, it is shown how the RMQ algorithm may be modified to incorporate reflecting or absorbing behaviour at the zero boundary. This allows the RMQ algorithm to correctly model the boundary behaviour of the underlying SDE and it increases the range of parameter sets the algorithm can handle, which is important for calibration.

In Chapter 4, the RMQ algorithm is generalized such that it can be applied to two higher-order discretization schemes. Numerical evidence of the improved weak-order convergence and computational efficiency of the two higher-order schemes is provided. Furthermore, the current theoretical error bound is extended to apply to the proposed higher-order methods.

In Chapter 5, a modification of the RMQ algorithm, christened the joint recursive marginal quantization (JRMQ) algorithm, is derived and applied to two-factor stochastic volatility models, providing a significant computational advantage over the algorithm presented by Callegaro et al. [2016].

Finally, in Chapter 6, RMQ and the innovations are applied for the first time outside the traditional risk-neutral pricing framework by pricing long-dated contracts under the real-world measure, using the benchmark approach of Platen and Heath [2006].

A chapter-by-chapter summary of the thesis is now provided.

1.2 Reading Guide

The thesis proceeds as follows.

In Chapter 2, the mathematics of the vector quantization of probability distributions is reviewed. The essential theorems are highlighted and the most common numerical techniques for obtaining an optimal quantizer are introduced. Extra attention is paid to the Newton-Raphson technique, as well as the quantization of the Gaussian distribution and the noncentral chi-squared

distribution with one degree of freedom, as these are central to the later development of the RMQ algorithm.

A simple matrix formulation of the Newton-Raphson algorithm for the one-dimensional vector quantization of probability distributions is provided, such that it can be immediately implemented. The theme of providing matrix-based expressions to simplify the implementation of the required numerical methods is kept throughout.

In Chapter 3, the RMQ algorithm is presented in more generality than the original formulation of Pagès and Sagna [2015]. It is also shown how to modify the RMQ algorithm to allow for an absorbing or reflecting boundary at zero. These modifications allow the RMQ algorithm to be applied for parameter sets that would otherwise be problematic under the original formulation. The quantization grids generated by the algorithm and its modifications are presented for geometric Brownian motion (GBM) and the constant elasticity of variance (CEV) process.

Using the more general formulation, Chapter 4 extends the RMQ algorithm to discretization schemes with higher orders of strong and weak convergence than the Euler-Maruyama scheme, specifically the Milstein [1975] scheme and a simplified weak-order 2.0 scheme of Kloeden and Platen [1999]. The GBM and the CEV models serve as examples to illustrate the improved weak-order convergence and computational efficiency. European, barrier and Bermudan options are priced using RMQ under these models for all three schemes. Where possible, the results are compared to available closed-form solutions, otherwise they are compared to high-resolution finite difference or Monte Carlo implementations. As an example of the flexibility of the RMQ algorithm and its extensions, a single-day calibration is performed for the CEV model to American option market data. Chapter 4 also extends the theoretical error bound provided for the standard RMQ algorithm by Pagès and Sagna [2015] to the Milstein scheme.

In Chapter 5, the JRMQ algorithm for two-factor stochastic volatility models is derived. It retains the underlying Newton-Raphson technique, but provides a large increase in efficiency over the algorithm presented by Callegaro et al. [2016]. It requires the computation of a joint probability matrix and an efficient approximation to this matrix is presented. The efficiency and accuracy of the algorithm is illustrated by pricing options on the Stein-Stein [Stein and Stein, 1991], Heston and SABR stochastic volatility models. A proof-of-concept calibration example is provided by calibrating the Heston model to American option market data.

Chapter 6 applies the higher-order RMQ and the JRMQ algorithms outside of the traditional risk-neutral pricing framework by pricing long-dated options under the real-world probability measure, using the benchmark approach from Platen and Heath [2006]. The mathematics of the benchmark approach is introduced using a two-asset continuous market, and the form of the growth-optimal portfolio (GOP), the central object of real-world pricing, is derived. Analytical pricing formulae are derived for the time-dependent constant elasticity of variance model (TCEV) for the GOP, generalizing the pricing formulae from Miller and Platen [2008, 2010]. Long-dated options are priced using quantization for both the TCEV model and the hybrid model of Baldeaux et al. [2015], which allows for a stochastic short rate. The effect of correlation between the stochastic short rate and the GOP on zero-coupon bond prices is investigated using the JRMQ algorithm.

Chapter 7 concludes by reviewing the contributions of the preceding chapters and advising on further research.

Notes on Publications

An early version of the work contained in Chapters 3 and 4 is available online in paper form as McWalter et al. [2017] and has now been published as McWalter et al. [2018].

The work contained in Chapter 5 is available online as Rudd et al. [2017]. It is important to highlight here its relation to the work by Callegaro et al. [2016] and Fiorin et al. [2017]. It improves the algorithm presented by Callegaro et al. [2016] significantly, as the results in the chapter indicate. The earlier versions of Fiorin et al. [2017] extended the paper by Callegaro et al. [2016] to higher dimensions than two, but retained an inefficient conditioning argument in the algorithm. The currently available version of Fiorin et al. [2017], however, seems to have independently developed the improvement presented here, although a proof is omitted.

Chapter 6 is available online in paper form as Rudd et al. [2018].

Chapter 2

Vector Quantization

2.1 Overview

Let X be a continuous random vector taking values in $\mathbb{R}^d$, with distribution F_X, and defined on a probability space $(\Omega, \mathscr{F}, \mathbb{P})$. The aim of this section is to answer the following question:

How does one optimally approximate X, by a discrete random vector, $\widehat{X} : \Omega \to \Gamma$, where Γ is a finite set of elements in $\mathbb{R}^d$?

Vector quantization is a lossy compression technique that provides a way to encode a vector space using a discrete subspace. The one-dimensional case is depicted in Figure 2.1, which shows a density function of a continuous random variable and its corresponding encoded or *quantized* version.

One reason that vector quantization is useful is that it allows efficient approximations of expectations of functionals of X, that is,

$$\mathbb{E}[H(X)] = \int_{\mathbb{R}^d} H(x)\, dF_X(x) \approx \sum_{\gamma \in \Gamma} H(\gamma)\mathbb{P}\big(\widehat{X} = \gamma\big). \tag{2.1}$$

Here, for example, H may be the discounted payoff of a financial claim and $\mathbb{P}$ the risk-neutral probability measure.

A vector quantizer can be expressed as

$$\pi : \mathbb{R}^d \to \Gamma,$$

where π is known as the *quantization function* or *quantizer*, and the set

$$\Gamma = \{\gamma^1, \ldots, \gamma^N\}$$

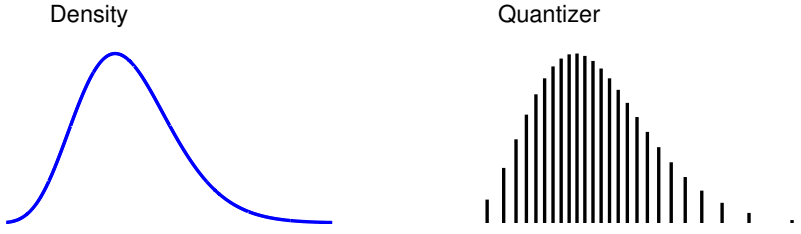

Figure 2.1: The continuous probability density function on the left is quantized on the right, with probabilities represented on the vertical axis.

is a subset of $\mathbb{R}^d$, with (at most) $N \geq 1$ elements, known as the *quantization grid*. Each element, γ^i, is known as an *elementary quantizer* or *codeword*. Owing to the wide application of vector quantization, terminology abounds: the set Γ is alternatively known as a *quantizer*, *codebook* or *code*.

Associated with the quantizer, the *regions* $R^i(\Gamma) \subset \mathbb{R}^d$ are the subsets of the values of X that are mapped to each codeword γ^i:

$$R^i(\Gamma) := \left\{x \in \mathbb{R}^d : \ \pi(x) = \gamma^i\right\}.$$

Consider the pointwise error made when approximating an input vector X by $\pi(X)$, its projection onto Γ. It is clear that

$$|X - \pi(X)| \geq \inf_{\gamma \in \Gamma} |X - \gamma|.$$

Here, $|\cdot|$ is the Euclidean norm. Equality will only hold in the above expression when π is chosen to be the *nearest-neighbor projection operator*, $\pi_\Gamma : \mathbb{R}^d \to \Gamma$, defined by

$$\pi_\Gamma(X) := \left\{\gamma^i \in \Gamma : \ |X - \gamma^i| \leq |X - \gamma^j| \text{ for } j = 1, \ldots, N, \ j \neq i\right\}.$$

As a simple tie-breaking rule, when an input vector is an equal distance from two or more elementary quantizers, the elementary quantizer with the lowest index is selected. With the quantization function chosen in this way, the set of regions $\{R^i(\Gamma)\}_{i=1}^N$, forms a special Borel partition satisfying

$$\begin{aligned} R^i(\Gamma) &= \{x \in \mathbb{R}^d : \ \pi_\Gamma(x) = \gamma^i\}, \\ &= \{x \in \mathbb{R}^d : \ |x - \gamma^i| \leq |x - \gamma^j| \text{ for } j = 1, \ldots, N, \ j \neq i\}, \end{aligned}$$

known as a *Voronoi* partition.

2.1.1 Optimal Quantization Grids

Consider the approximation of X by $\widehat{X}$, a discrete random vector defined as the nearest-neighbour projection of X onto the quantization grid Γ,

$$\widehat{X} := \pi_\Gamma(X).$$

This is known as the *quantized* version of X. The aim of this section is to find the quantization grid Γ, such that $\widehat{X}$ "best" approximates X. The optimality of a quantization grid is measured using the average error made when reproducing the input vectors, known as the *distortion*. Although the theory of quantization can be established for a variety of norms, see Graf and Luschgy [2000], in this work only the Euclidean norm is considered.

Definition 2.1.1 (Distortion)**.** *Let $X \in \mathcal{L}^2$, i.e., $\int_{\mathbb{R}^d} |x|^2 \, dF_X(x) < +\infty$. The $\mathbb{R}^+$-valued function D, defined on $(\mathbb{R}^d)^N$ by*

$$D : (\gamma^1, \ldots, \gamma^N) \mapsto \|X - \pi_\Gamma(X)\|_2^2 = \mathbb{E}\left[|X - \widehat{X}|^2\right] = \int_{\mathbb{R}^d} |x - \pi_\Gamma(x)|^2 \, dF_X(x) \tag{2.2}$$

is known as the distortion *function.*

Here, $\|\cdot\|_p$ indicates the $\mathcal{L}^p$-norm. An optimal quantization grid will be one that minimizes this distortion function. The following proposition justifies this pursuit.

Proposition 2.1.1 (Existence of Optimal Quantizers). *Assume $X \in \mathcal{L}^2$, so that the distortion function D is finite everywhere on $(\mathbb{R}^d)^N$. Then:*

1. *The distortion function D attains a minimum at an N-tuple, $\gamma^{(N)} = (\gamma^1, \ldots, \gamma^N)$.*
2. *If $\operatorname{card}(\operatorname{supp}(F_X)) \geq N$, then the quantization grid corresponding to the N-tuple at which the distortion function attains a minimum, $\Gamma = \{\gamma^1, \ldots, \gamma^N\}$, has full size N, i.e., pairwise distinct components. Furthermore, for every Voronoi partition $\{R^i(\Gamma)\}_{i=1}^N$ induced by Γ, $\mathbb{P}(X \in R^i(\Gamma)) > 0$.*
3. *The sequence*

$$N \mapsto \min_{\gamma^{(N)} \in (\mathbb{R}^d)^N} \|X - \widehat{X}\|_2$$

strictly decreases as long as $N \leq \operatorname{card}(\operatorname{supp}(F_X))$ and

$$\lim_{N \to \infty} \min_{\gamma^{(N)} \in (\mathbb{R}^d)^N} \|X - \widehat{X}\|_2 = 0.$$

Proof. See Pagès [2014, Sec. 2.1] □

Thus, when only considering the $\mathcal{L}^2$-norm and random vectors in $\mathbb{R}^d$, the distortion function will attain a minimum. Furthermore, the optimal quantization grid has full size and does not induce any Voronoi regions with zero measure.

To search for an optimal grid, the differentiability of the distortion function must be established.

Proposition 2.1.2 (Differentiability of the Distortion). *Let $X \in \mathcal{L}^2$. If $\gamma^{(N)} = (\gamma^1, \ldots, \gamma^N) \in (\mathbb{R}^d)^N$ has pairwise distinct components, the distortion function, D, is finite and differentiable at $\gamma^{(N)}$ and*

$$\begin{aligned} [\nabla D(\gamma^{(N)})]_i &= \frac{\partial D(\gamma^{(N)})}{\partial \gamma^i} \\ &= 2\mathbb{E}\big[(\gamma^i - X)\mathbb{I}_{\{X \in R^i(\Gamma)\}}\big] \\ &= 2\int_{R^i(\Gamma)} (\gamma^i - x)\, dF_X(x) \end{aligned} \tag{2.3}$$

for $i = 1, \ldots, N$.

Proof. See Pagès [2014, Sec. 3.1] □

This result leads directly to the following corollary, which is central to all numerical applications of vector quantization.

Corollary 2.1.3 (Self-consistent Quantizers). *Let $X \in \mathcal{L}^2$. Any grid Γ attached to an N-tuple $\gamma^{(N)}$ which minimizes the distortion function D is a* self-consistent *quantizer, i.e.,*

$$\widehat{X} = \mathbb{E}\big[X \big| \widehat{X}\big],$$

or equivalently

$$\begin{aligned} \gamma^i &= \frac{\mathbb{E}[X\mathbb{I}_{\{X \in R^i(\Gamma)\}}]}{\mathbb{P}(X \in R^i(\Gamma))} \\ &= \frac{\int_{R^i(\Gamma)} x\, dF_X(x)}{\int_{R^i(\Gamma)} dF_X(x)} \end{aligned} \tag{2.4}$$

for $i = 1, \ldots, N$.

Intuitively, knowledge of $\widehat{X}$ is knowledge of the *region* in which X belongs. The self-consistency condition states that, for an optimal quantization grid, averaging X over any region yields the codeword that generates that region.

Numerical methods for obtaining optimal quantization grids now proceed either by minimizing the distortion function directly, or by searcing for self-consistent[1] grids.

Searching for a self-consistent quantization grid is equivalent to constructing a centroidal Voronoi tesselation under the distribution of X, where the points that generate each of the Voronoi regions are simultaneously the probability mass centroids of their regions. This can be seen directly from (2.4). This geometric view of the optimal quantization problem is thoroughly explored in Du et al. [1999].

2.2 Error Analysis

This section establishes the required error bounds for vector quantization. Of central theoretical importance are the two theorems presented in Section 2.2.3. Note that the notation $D(\Gamma)$ is used to indicate the distortion of the N-tuple, $\gamma^{(N)}$, associated with the quantization grid, Γ, in the obvious way.

2.2.1 Numerical Integration

This section summarizes the error bounds for the application of quantization to numerical integration and follows Pagès [2008a, Sec. 5.2]. The error bounds can be classified by the smoothness of the function being integrated, f.

Specifically, consider $f : \mathbb{R}^d \to \mathbb{R}$, with f integrable and $X \in \mathcal{L}^2$, and the approximation

$$\begin{aligned}\mathbb{E}[f(X)] &\approx \mathbb{E}[f(\widehat{X})] \\ &= \int_{\mathbb{R}^d} f(x)\, dF_{\widehat{X}}(x) \\ &= \sum_{i=1}^{N} f(\gamma^i)\mathbb{P}(X \in R^i(\Gamma)).\end{aligned}$$

Proposition 2.2.1 (Lipschitz Continuous Function)**.** *If f is* Lipschitz continuous*, such that $[f]_{\text{Lip}} = \sup_{x\neq y} \frac{|f(x)-f(y)|}{|x-y|} < +\infty$, then*

$$\left|\mathbb{E}[f(X)] - \mathbb{E}[f(\widehat{X})]\right| \leq [f]_{\text{Lip}}\sqrt{D(\Gamma)}.$$

Proof. See Appendix A. □

Because the Lipschitz continuous functionals form a characterizing family for the weak convergence of probability measures in $\mathbb{R}^d$, Proposition 2.2.1 shows that for a sequence of optimal quantization grids, the probability measure induced by the quantization, $F_{\widehat{X}}$, weakly converges to F_X.

Proposition 2.2.2 (Lipschitz Continuous Derivative)**.** *If f is* differentiable with Lipschitz continuous derivative *f' and the grid Γ is an optimal, and thus self-consistent, quantization grid, then*

$$\left|\mathbb{E}[f(X)] - \mathbb{E}[f(\widehat{X})]\right| \leq [f']_{\text{Lip}} D(\Gamma).$$

[1] The self-consistency property is often known as *stationarity* in the literature. This term is avoided here owing to the potential confusion with stationary stochastic processes.

Proof. See Appendix A. □

Note that $D(\Gamma)$ is much smaller than $\sqrt{D(\Gamma)}$ when Γ is an optimal quantization grid and N is large enough. Finally, when f is *convex* and Γ is an optimal quantization grid, Jensen's inequality immediately yields

$$\mathbb{E}[f(\widehat{X})] \leq \mathbb{E}[f(X)], \tag{2.5}$$

implying that numerical integration using quantization approaches the true value of $\mathbb{E}[f(X)]$ from below.

2.2.2 Conditional Expectation

The numerical computation of conditional expectations is of central importance in mathematical finance, owing to the presence of early exercise in the valuation of American-style (or more generally, *callable*) options.

It is desired to form approximations of the kind

$$\mathbb{E}[f(X)|Y] \approx \mathbb{E}\big[f(\widehat{X})|\widehat{Y}\big],$$

where $\widehat{X}$ and $\widehat{Y}$ are the quantized versions of the random vectors X and Y, with their own quantization grids, Γ^x and Γ^y, possibly of different cardinality. The proposition below summarizes the main result.

Proposition 2.2.3. *Let X, Y be $\mathbb{R}^d$-valued random vectors on $(\Omega, \mathscr{F}, \mathbb{P})$, and $f : \mathbb{R}^d \to \mathbb{R}$ a Borel function. Let $\varphi_f : \mathbb{R}^d \to \mathbb{R}$ be a Borel version of the conditional expectation,*

$$\varphi_f(Y) = \mathbb{E}[f(X)|Y].$$

Lastly, assume f and φ_f are Lipschitz continuous with Lipschitz coefficients $[f]_{\text{Lip}}$ and $[\varphi_f]_{\text{Lip}}$ respectively. Then

$$\begin{aligned}\big\|\mathbb{E}[f(X)|Y] - \mathbb{E}\big[f(\widehat{X})|\widehat{Y}\big]\big\|_2^2 &\leq [f]_{\text{Lip}}^2 \big\|X - \widehat{X}\big\|_2^2 + [\varphi_f]_{\text{Lip}}^2 \big\|Y - \widehat{Y}\big\|_2^2 \\ &= [f]_{\text{Lip}}^2 D(\Gamma^x) + [\varphi_f]_{\text{Lip}}^2 D(\Gamma^y).\end{aligned}$$

Proof. The proof for the quadratic case above appears in Pagès [2008a, Sec. 5.2.4] whereas the general $\mathcal{L}^p$-case is considered in Pagès [2014, Sec. 1.1.3]. □

2.2.3 Convergence

The asymptotic (or *sharp*) rate of convergence of vector quantizers is governed by Zador's theorem, reproduced below in a simplified form.

Theorem 2.2.4 (Zador's Theorem)**.** *Let $X \in \mathcal{L}^{2+\delta}(\Omega, \mathscr{F}, \mathbb{P})$ for some $\delta > 0$. Define $\varphi = \frac{dF_X}{d\lambda_d}$ as the Radon-Nikodym density of the absolutely continuous part of F_X with respect to the Lebesgue measure λ_d on $\mathbb{R}^d$. Then there exists a real constant $\widetilde{J}_{2,d} \in (0, \infty)$ such that*

$$\lim_{N\to\infty} N^{\frac{1}{d}} \min_{\gamma^{(N)} \in (\mathbb{R}^d)^N} \big\|X - \widehat{X}\big\|_2 = \widetilde{J}_{2,d} \|\varphi\|_{\mathcal{L}^{\frac{d}{d+2}}(\lambda_d)}$$

or, alternatively,

$$\lim_{N\to\infty} N^{\frac{2}{d}} \min_{\gamma^{(N)} \in (\mathbb{R}^d)^N} D(\gamma^{(N)}) = (\widetilde{J}_{2,d})^2 \|\varphi\|_{\mathcal{L}^{\frac{d}{d+2}}(\lambda_d)},$$

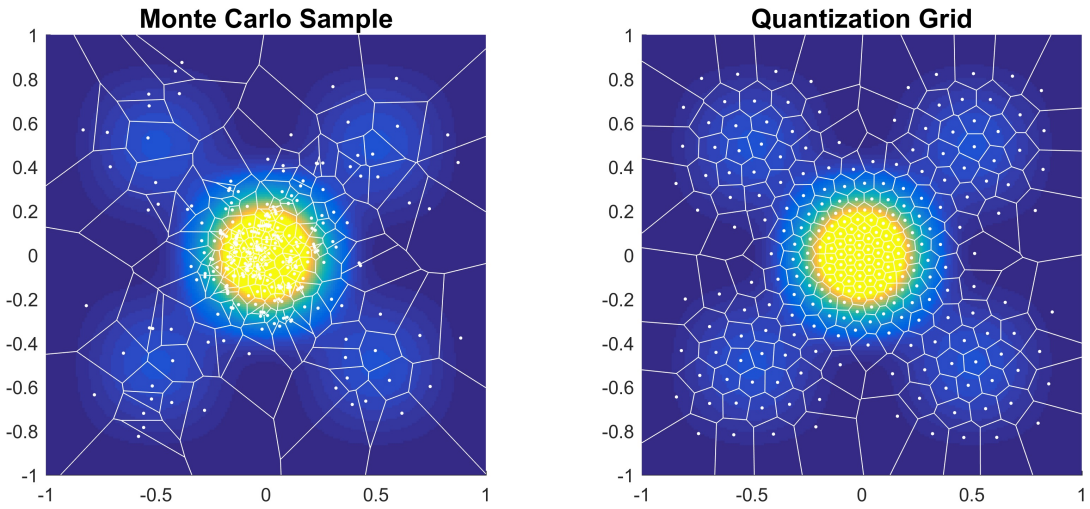

Figure 2.2: The two-dimensional vector quantization of the density given by (2.7), using the randomized Lloyd's algorithm and $N = 256$ codewords.

where $\|\varphi\|_{\mathcal{L}^r(\lambda_d)} = \left(\int_{\mathbb{R}^d} |\varphi|^r \, d\lambda_d\right)^{\frac{1}{r}}$ *for* $r > 0$.

Proof. Originally from Zador [1966], a proof appears in Graf and Luschgy [2000]. □

Thus if X has a continuous density, f_X, Zador's theorem states that the minimum value of the distortion obtained over all possible grids, scaled by $N^{\frac{2}{d}}$, converges to a constant multiple, $(\widetilde{J}_{2,d})^2$, of the $\mathcal{L}^{\frac{d}{d+2}}$-norm of f_X. Zador's theorem in fact holds for any norm on $\mathbb{R}^d$, and the value of $\widetilde{J}_{2,d}$ will depend on the norm selected, hence the double subscript.

Zador's theorem provides the vector quantization error bound in the limiting case; the minimum distortion that can be attained as the number of elementary quantizers in the quantization grid goes to infinity. The non-asymptotic upper bound for vector quantization is provided by Pierce's lemma below; it bounds the distortion for a fixed number of elementary quantizers.

Theorem 2.2.5 (Extended Pierce's Lemma)**.** *Let* $X \in \mathcal{L}^2(\Omega, \mathscr{F}, \mathbb{P})$ *and* $\delta > 0$. *Then there exists a real constant* $K_{d,\delta} \in (0, \infty)$ *such that*

$$\min_{\gamma^{(N)} \in (\mathbb{R}^d)^N} \left\|X - \widehat{X}\right\|_2 \leq K_{d,\delta} \sigma_{2+\delta}(X) N^{-\frac{1}{d}},$$

for all $N \geq 1$ *and where* $\sigma_r(X) = \min_{a \in \mathbb{R}^d} \|X - a\|_r$.

Proof. See Luschgy and Pagès [2008]. □

2.3 Numerical Methods

This section highlights the three most common numerical methods used to solve the vector quantization problem and obtain optimal quantization grids. For further detail, specifically on initialization methods, see Pagès et al. [2003].

2.3.1 Fixed-point Methods

A common algorithm to obtain an optimal quantization grid is *Lloyd's algorithm*, originally due to Lloyd [1982]. It is based on recursively enforcing the necessary self-consistency condition of the quantization grid. It proceeds as follows:

1. Select a set of N pairwise distinct points to form the initial quantization grid, Γ.
2. Construct the Voronoi partition, $\{R^i(\Gamma)\}_{i=1}^N$, associated with the current grid.
3. Compute the probability mass centroids of the Voronoi regions found in Step 2. Assign these centroids as the codewords for the new quantization grid.
4. If this new quantization grid meets a pre-specified convergence criterion, terminate. Otherwise, return to Step 2.

In this way, the necessary self-consistency condition for optimal quantization grids,

$$\gamma^i = \frac{\mathbb{E}[X\mathbb{I}_{\{X\in R^i(\Gamma)\}}]}{\mathbb{P}(X \in R^i(\Gamma))}$$

gives rise to an iterative fixed-point procedure

$$^{(l+1)}\gamma^i = \frac{\int_{R^i(^{(l)}\Gamma)} x\, dF_X(x)}{\int_{R^i(^{(l)}\Gamma)} dF_X(x)}, \tag{2.6}$$

for $1 \le i \le N$ with $0 \le l < l_{\max}$ indicating the iteration index. It should be clear that computing the mass centroids in (2.6) can require high-dimensional integrals over Voronoi cells. In a probabilistic setting, Monte Carlo (or Quasi-Monte Carlo) methods are used to compute these integrals. This variation is known as the *randomized* Lloyd's algorithm,

$$^{(l+1)}\gamma^i = \frac{\sum_{m=1}^M X_m \mathbb{I}_{\{X_m\in R^i(^{(l)}\Gamma)\}}}{\sum_{m=1}^M \mathbb{I}_{\{X_m\in R^i(^{(l)}\Gamma)\}}},$$

where X_m is a random or quasi-random sample generated from the distribution of X and M is the number of samples generated per iteration of the algorithm. Convergence is established for Lloyd's algorithm in Emelianenko et al. [2008].

In Du et al. [1999], the bivariate density function

$$f(x,y) = e^{-20x^2-20y^2} + 0.05\sin^2(\pi x)\sin^2(\pi y) \tag{2.7}$$

is selected as an example of a density that has a large central peak but varies, with a small amplitude, away from the peak. Figure 2.2 illustrates the vector quantization of this density using the randomized Lloyd's algorithm. The left panel of Figure 2.2 shows 256 Monte Carlo samples, generated using the rejection method, along with the accompanying Voronoi tesselation. These were used as the initial quantization grid for the randomized Lloyd's algorithm. After 120 iterations the quantization grid shown in the right panel of the figure is obtained.

2.3.2 Gradient Descent Methods

A gradient descent procedure can be implemented to determine the critical points of the distortion function,

$$^{(l+1)}\mathbf{\Gamma} = {}^{(l)}\mathbf{\Gamma} - \alpha_{l+1}\nabla D(^{(l)}\mathbf{\Gamma}), \qquad {}^{(0)}\mathbf{\Gamma} \in (\text{hull}(\text{supp}(F_X)))^N,$$

where $l \ge 0$ is the iteration index, and $(\alpha_l)_{l\ge 1} \in (0,1)$ is a sequence of *step* or *gain* parameters satisfying the usual conditions,

$$\sum_{l\ge 1}\alpha_l = +\infty \qquad \text{and} \qquad \sum_{l\ge 1}\alpha_l^2 < +\infty.$$

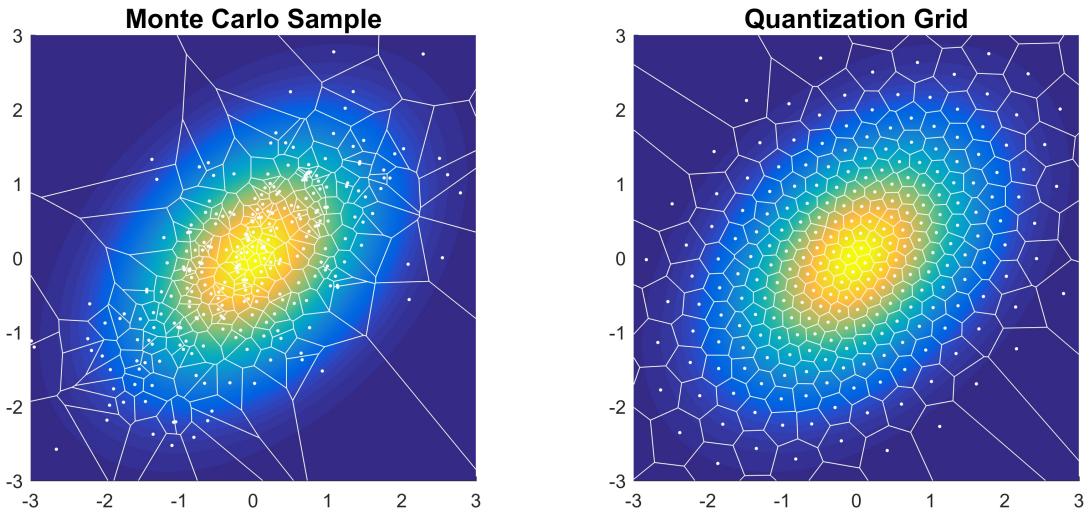

Figure 2.3: Vector quantization of a bivariate Gaussian distribution with correlation $\rho = 0.4$ and $N = 256$ codewords.

Note that here

$$\left[{}^{(l)}\mathbf{\Gamma}\right]_i := {}^{(l)}\gamma^i,$$

and thus $\mathbf{\Gamma}$ indicates a column vector of length N containing the codewords associated with the quantization grid, ${}^{(l)}\Gamma$, and the distortion function acts upon this vector in the obvious way.

When the dimensions are low, various deterministic methods can be used (see the next section). However, as the number of dimensions increases, these methods quickly become untenable and must be replaced with stochastic gradient descent, such as the *Competitive Learning Vector Quantization* (CLVQ) algorithm.

The CLVQ algorithm can be specified as

$${}^{(l+1)}\mathbf{\Gamma} = {}^{(l)}\mathbf{\Gamma} - \alpha_{l+1}\left(\mathbb{I}_{\left\{X_{l+1}\in R^i({}^{(l)}\mathbf{\Gamma})\right\}}\left({}^{(l)}\gamma^i - X_{l+1}\right)\right)_{1\le i\le N}, \tag{2.8}$$

where X_l is a random sample from the distribution of X. The algorithm proceeds in two phases:

- The Competitive Phase:
  Also known as the winner-selection phase, here the "winning" index is selected using a nearest-neighbour search,

  $$i_{\text{win}} = \text{argmin}_{i\in\{1,\dots,N\}}\left|{}^{(l)}\gamma^i - X_{l+1}\right|.$$

- The Learning Phase:
  The winning index is updated by moving it toward the random sample,

  $${}^{(l+1)}\gamma^{i_{\text{win}}} = {}^{(l)}\gamma^{i_{\text{win}}} - \alpha_{l+1}\left({}^{(l)}\gamma^{i_{\text{win}}} - X_{l+1}\right),$$

  and

  $${}^{(l+1)}\gamma^{i} = {}^{(l)}\gamma^{i}$$

  for every $i \neq i_{\text{win}}$.

In (2.8) the Competitive Phase is combined with the Learning Phase by using the indicator function

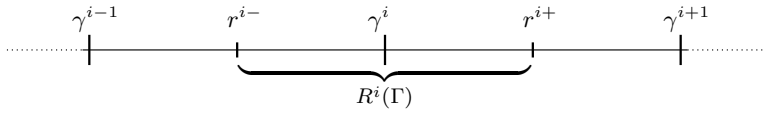

Figure 2.4: Representation of the region $R^i(\Gamma)$ associated with codeword γ^i in the one-dimensional case.

to represent the nearest-neighbour search. The numerical aspects of the CLVQ algorithm are investigated in Pagès et al. [2003] for the d-dimensional Gaussian case.

In the left panel of Figure 2.3, a Voronoi diagram is shown for a Monte Carlo sample of the bivariate Gaussian distribution with correlation $\rho = 0.4$ and $N = 256$ points. The Voronoi diagram of the optimal quantizer is displayed on the right. The optimal quantizer was obtained using successive passes of the CLVQ algorithm and the *splitting* initialization technique described in Pagès et al. [2003]. The result was then refined using 120 iterations of the randomized Lloyd's algorithm.

The Newton-Raphson Method

The special case when X is a one-dimensional random vector with a well-defined density function is relevant for many applications, including the recursive marginal quantization and joint recursive marginal quantization algorithms presented later.

In this case, given the gradient and Hessian of the distortion, the quantizer can be computed using the Newton-Raphson method,

$$ {}^{(l+1)}\boldsymbol{\Gamma} = {}^{(l)}\boldsymbol{\Gamma} - \left[\nabla^2 D\left({}^{(l)}\boldsymbol{\Gamma}\right)\right]^{-1} \nabla D\left({}^{(l)}\boldsymbol{\Gamma}\right), \tag{2.9} $$

for $0 \leq l < l_{\max}$. What remains is to compute the gradient and the Hessian of the distortion function explicitly for an arbitrary random variable.

In one dimension, the regions associated with a quantization grid may be defined directly as $R^i = \{x \in \mathbb{R} : r^{i-} < x \leq r^{i+}\}$ with

$$ r^{i-} = \frac{\gamma^{i-1} + \gamma^i}{2} \qquad \text{and} \qquad r^{i+} = \frac{\gamma^i + \gamma^{i+1}}{2}, $$

for $1 \leq i \leq N$, where, by definition, $r^{1-} = -\infty$ and $r^{N+} = \infty$. If the distribution under consideration is not defined over the whole real line, then r^{1-} and r^{N+} are adjusted to reflect the interval of support. Figure 2.4 shows a simple graphical representation of these regions.

Suppose f_X and F_X are the density and distribution functions of X, respectively. Define the p-th lower partial expectation as

$$ M_X^p(x) := \mathbb{E}[X^p \mathbb{I}_{\{X<x\}}], $$

where $M_X^0(X) = F_X(x)$ represents the distribution function of X. Then, direct integration of the

distortion function (2.2) gives

$$\begin{aligned} D(\Gamma) &= \sum_{i=1}^{N} \int_{R^i(\Gamma)} |x-\gamma^i|^2 \, dF_X(x) \\ &= \sum_{i=1}^{N} \int_{r^{i-}}^{r^{i+}} |x-\gamma^i|^2 f_X(x) \, dx \\ &= \sum_{i=1}^{N} \Big[M_X^2(r^{i+}) - M_X^2(r^{i-}) - 2\gamma^i \left(M_X^1(r^{i+}) - M_X^1(r^{i-}) \right) + (\gamma^i)^2 \left(F_X(r^{i+}) - F_X(r^{i-}) \right) \Big]. \end{aligned}$$

Consequently, the elements of the vector $\nabla D(\mathbf{\Gamma})$ are given by

$$\frac{\partial D(\Gamma)}{\partial \gamma^i} = 2\gamma^i \left(F_X(r^{i+}) - F_X(r^{i-}) \right) - 2 \left(M_X^1(r^{i+}) - M_X^1(r^{i-}) \right),$$

for $1 \leq i \leq N$.

Similarly, the tridiagonal Hessian matrix, $\nabla^2 D(\mathbf{\Gamma})$, may be computed. It has diagonal elements given by

$$\frac{\partial^2 D(\Gamma)}{\partial (\gamma^i)^2} = 2 \left(F_X(r^{i+}) - F_X(r^{i-}) \right) + \tfrac{1}{2} \left(f_X(r^{i+})(\gamma^i - \gamma^{i+1}) + f_X(r^{i-})(\gamma^{i-1} - \gamma^i) \right),$$

and super- and sub-diagonal elements given by

$$\frac{\partial^2 D(\Gamma)}{\partial \gamma^i \partial \gamma^{i+1}} = \tfrac{1}{2} f_X(r^{i+})(\gamma^i - \gamma^{i+1}) \qquad \text{and} \qquad \frac{\partial^2 D(\Gamma)}{\partial \gamma^i \partial \gamma^{i-1}} = \tfrac{1}{2} f_X(r^{i-})(\gamma^{i-1} - \gamma^i),$$

respectively. Note that the quantities required to compute a Newton-Raphson iteration (i.e., the gradient and Hessian) only require the density function, distribution function and first lower partial expectation to be known. The second lower partial expectation is required only if one wishes to compute the final distortion.

Having computed the quantizer $\mathbf{\Gamma}$, the associated row vector of probabilities, $\mathbf{p}$, is computed as

$$[\mathbf{p}]_i = \mathbb{P}(\widehat{X} = \gamma^i) = \mathbb{P}(X \in R^i(\Gamma)) = F_X(r^{i+}) - F_X(r^{i-}),$$

for $1 \leq i \leq N$. Defining $\mathbf{p}$ as a row vector is convenient since the expectation of a functional H applied to the quantizer is

$$\mathbb{E}[H(X)] = \sum_{i=1}^{N} H(\gamma^i) \mathbb{P}\big(\widehat{X} = \gamma^i\big) = \mathbf{p} H(\mathbf{\Gamma}),$$

where H is applied element-wise to $\mathbf{\Gamma}$. Moreover, this will be compatible with a Markov chain formulation of the recursive marginal quantization technique presented later.

2.3.3 Matrix Formulation

Owing to the central importance of the Newton-Raphson algorithm for vector quantization in this work, an efficient matrix formulation of these equations is now provided in order to facilitate easy implementation.

As stated previously, the quantizer is represented by a column vector $\mathbf{\Gamma}$. This vector and three

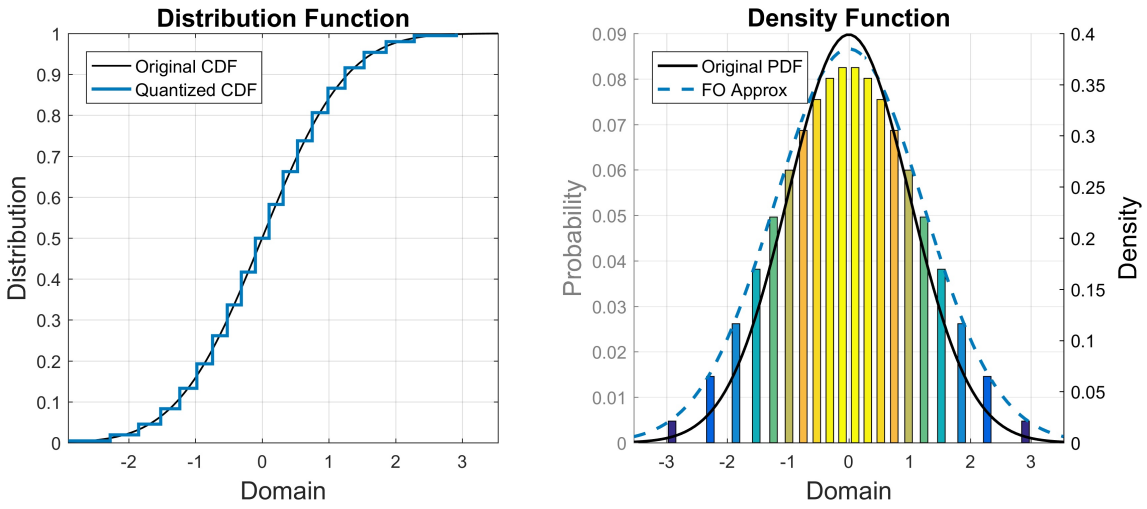

Figure 2.5: Vector quantization of the standard Gaussian distribution using the Newton-Raphson method with $N = 20$ codewords.

other column vectors required are defined by

$$[\boldsymbol{\Gamma}]_i = \gamma^i, \qquad [\mathbf{M}]_i = M_X^1(r^{i+}) - M_X^1(r^{i-}), \qquad 1 \le i \le N,$$
$$[\mathbf{f}]_i = f_X(r^{i+}), \qquad [\Delta\boldsymbol{\Gamma}]_i = \gamma^{i+1} - \gamma^i, \qquad 1 \le i \le N-1.$$

Note that the last two vectors are one element shorter than the first two.

Using these vectors the gradient of the distortion function is then

$$\nabla D(\boldsymbol{\Gamma}) = 2\boldsymbol{\Gamma} \circ \mathbf{p}^\top - 2\mathbf{M},$$

where $\circ$ indicates the element-wise Hadamard product.

The super- and sub-diagonal (or off-diagonal) entries of the Hessian matrix $\nabla^2 D(\boldsymbol{\Gamma})$ are given by the length-$(N-1)$ row vector

$$\mathbf{h}_{\mathsf{off}} = -\frac{1}{2}[\mathbf{f} \circ \Delta\boldsymbol{\Gamma}]^\top,$$

with the main diagonal given by

$$\mathbf{h}_{\mathsf{main}} = 2\mathbf{p} + [\mathbf{h}_{\mathsf{off}}|0] + [0|\mathbf{h}_{\mathsf{off}}],$$

where the copies of the $\mathbf{h}_{\mathsf{off}}$ vector are appended and prepended with a zero. It is now straightforward to set up the Newton-Raphson iteration in terms of the quantities.

2.4 Examples

In this section, the expressions required to implement (2.9) are provided for the standard normal distribution and the noncentral chi-squared distribution with one degree of freedom. These two distributions play a central role in the recursive marginal quantization algorithm presented in Chapter 3.

2.4.1 The Gaussian Distribution

When X is a standard normal random variable the Newton-Raphson procedure for the optimal vector quantization can be used, with

$$\begin{aligned} f_X(x) &= \phi(x), \\ F_X(x) &= \Phi(x), \\ M_X^1(x) &= -\frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}} = -\phi(x), \end{aligned}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the standard normal density and distribution functions, respectively. Here, a good guess for the initial quantizer $\mathbf{\Gamma}^{(0)}$ is

$$\gamma^i = \frac{5.5i}{N+1} - 2.75,$$

for $1 \leq i \leq N$. This formula linearly spaces the initial points about the mean out to 2.75 standard deviations.

It was proven in Delattre et al. [2004] that if Γ is an optimal quantizer for the random variable X of cardinality N, the mass of each of the elementary codewords is given by

$$\mathbb{P}(X \in R^i(\Gamma)) = \frac{1}{N} f_X^{\frac{2}{3}}(\gamma^i) \left(\int_{\mathbb{R}} f_X^{\frac{2}{3}}(z)\, dz \right) + o\left(\frac{1}{N}\right). \tag{2.10}$$

The optimal quantizer of a standard Gaussian random variable using 20 codewords is shown in Figure 2.5. The left panel shows the quantized cumulative distribution function, whereas the right panel shows the quantization grid itself, along with its accompanying probabilities. The first term of the above formula, (2.10), is indicated by the dashed blue line in the right panel of Figure 2.5. For comparison, the standard Gaussian density is displayed with its own scaled axis on the right.

2.4.2 The Noncentral Chi-squared Distribution

While, in general, the noncentral chi-squared distribution must be specified using Bessel functions, this is not the case when the degree of freedom equals one. In particular, consider the random variable $X = (z + \mu)^2$, where $z \sim \mathcal{N}(0, 1)$. Then $X \sim \chi'^2(1, \lambda)$, is a noncentral chi-squared distributed random variable with one degree of freedom and noncentrality parameter $\lambda = \mu^2$. Moreover, on $x \in \mathbb{R}^+$,

$$\begin{aligned} f_X(x) &= \frac{1}{2\sqrt{x}} \left(\phi(x^+) + \phi(x^-) \right) \\ F_X(x) &= \Phi(x^+) - \Phi(x^-) \\ M_X^1(x) &= (1+\lambda) \left(\Phi(x^+) - \Phi(x^-) \right) + \phi(x^+)x^- - \phi(x^-)x^+, \end{aligned}$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the standard normal density and distribution functions, respectively, and

$$x^{\pm} = \pm\sqrt{x} - \sqrt{\lambda}.$$

This means that the noncentral chi-squared distribution with one degree of freedom may be expressed using the standard normal density and distribution functions, thus allowing efficient computation of a quantization scheme. This will be important for computational efficiency when higher-order recursive marginal quantization schemes are presented in Chapter 4.

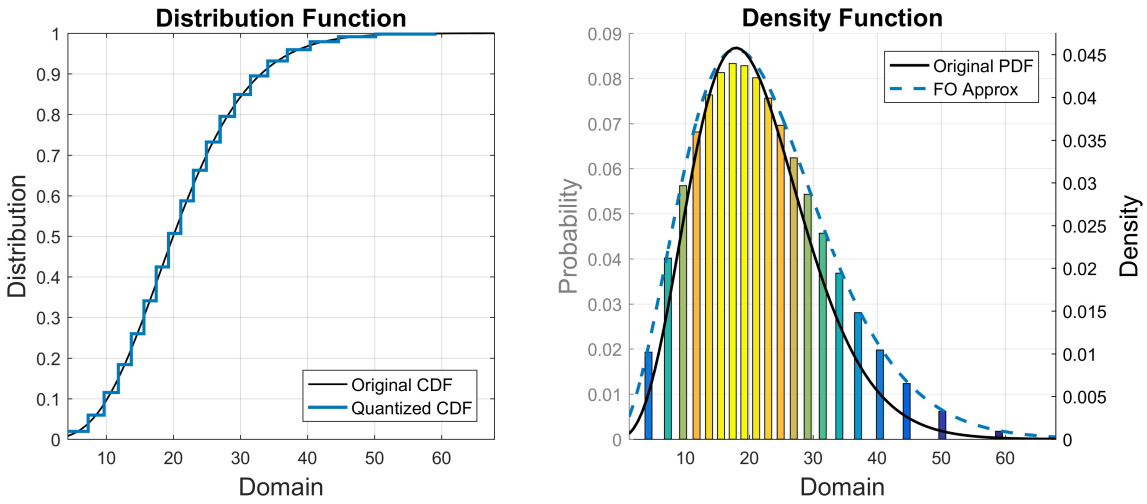

Figure 2.6: Vector quantization of a noncentral chi-squared distribution, using $N = 20$ codewords, with one degree of freedom and non-centrality $\lambda = 20$.

When implementing vector quantization, care must be taken when evaluating these functions at the left and right limits. To ensure convergence, set $f_X(0) = F_X(0) = M_X^1(0) = 0$, $f_X(\infty) = 0$, $F_X(\infty) = 1$ and $M_X^1(\infty) = 1 + \lambda$. Of course, all three functions are zero when x is negative. A good initial guess for $\mathbf{\Gamma}^{(0)}$ is given by

$$\gamma^n = \begin{cases} \left(\frac{(3+\sqrt{\lambda})n}{N}\right)^2 & \text{for } \sqrt{\lambda} < 2.5 \\ \left(\frac{5n}{N+1} - 2.5 + \sqrt{\lambda}\right)^2 & \text{for } \sqrt{\lambda} \geq 2.5, \end{cases}$$

for $1 \leq n \leq N$. The second part of the formula above, for when $\sqrt{\lambda} \geq 2.5$, follows directly from the initial guess for the standard Gaussian random variable with the 2.75 standard deviation bound decreased to 2.5. However, when $\sqrt{\lambda} < 2.5$, this causes points to be reflected about the origin and instead the first formula is used. Numerical experiments validate this choice of initial quantizer.

Similarly to Figure 2.5, the optimal quantizer for a noncentral chi-squared random variable with one degree of freedom and a noncentrality of $\lambda = 20$ is displayed in Figure 2.6.

Chapter 3

Recursive Marginal Quantization

3.1 Overview

Consider the continuous-time vector-valued diffusion specified by the stochastic differential equation (SDE)

$$dX_t = a(X_t)\,dt + b(X_t)\,dW_t, \qquad X_0 = x_0 \in \mathbb{R}^d, \tag{3.1}$$

defined on the filtered probability space $(\Omega, \mathscr{F}, (\mathscr{F}_t)_{t\in[0,T]}, \mathbb{P})$, where W is a standard q-dimensional Brownian motion. Here $a : \mathbb{R}^d \to \mathbb{R}^d$ and $b : \mathbb{R}^d \to \mathbb{R}^{d\times q}$ are assumed to be sufficiently smooth and bounded to ensure the existence of a strong solution. The question of interest is:

> How does one optimally approximate $X_{t_k} : \Omega \to \mathbb{R}^d$, for some time discretisation point $t_k \in [0, T]$, when the distribution of X_{t_k} is unknown?

Usually this is achieved by performing a Monte Carlo experiment using a discrete-time approximation scheme for the SDE, the simplest scheme being the Euler-Maruyama [Maruyama, 1955] update

$$\begin{aligned}\overline{X}_{k+1} &= \overline{X}_k + a(\overline{X}_k)\Delta t + b(\overline{X}_k)\sqrt{\Delta t} z_{k+1} \\ &=: \mathcal{U}(\overline{X}_k, z_{k+1}),\end{aligned}$$

for $0 \le k < n$, where $\Delta t = T/n$ and independent $z_{k+1} \sim \mathcal{N}(0, \mathbf{I}_q)$, with initial value $\overline{X}_0 = x_0$. The innovation of Pagès and Sagna [2015] was to show that a recursive procedure based on quantizing these updates is possible.

Since $\overline{X}_1$ has a multivariate Gaussian distribution, it is possible to use vector quantization to obtain $\widehat{X}_1$, an optimal quantizer for the first step of the above scheme. This yields $\Gamma_1 = \{\gamma_1^1, \ldots, \gamma_1^{N_1}\}$ and its associated probabilities. One must, however, find a way to quantize the successive (marginal) distributions of $\overline{X}_{k+1}$. Given knowledge of the distribution of $\overline{X}_k$, the distortion of the quantizer Γ_{k+1} may be written as

$$\begin{aligned}\overline{D}(\Gamma_{k+1}) &= \mathbb{E}\Big[\big|\overline{X}_{k+1} - \pi_{\Gamma_{k+1}}(\overline{X}_{k+1})\big|^2\Big] \\ &= \mathbb{E}\Big[\mathbb{E}\Big[\big|\overline{X}_{k+1} - \pi_{\Gamma_{k+1}}(\overline{X}_{k+1})\big|^2 \,|\, \overline{X}_k\Big]\Big] \\ &= \mathbb{E}\Big[\mathbb{E}\Big[\big|\mathcal{U}(\overline{X}_k, z_{k+1}) - \pi_{\Gamma_{k+1}}(\mathcal{U}(\overline{X}_k, z_{k+1}))\big|^2 \,|\, \overline{X}_k\Big]\Big] \\ &= \int_{\mathbb{R}^d} \mathbb{E}\Big[\big|\mathcal{U}(x, z_{k+1}) - \pi_{\Gamma_{k+1}}(\mathcal{U}(x, z_{k+1}))\big|^2\Big]\, d\mathbb{P}(\overline{X}_k \le x).\end{aligned} \tag{3.2}$$

Unfortunately, the exact distribution of $\overline{X}_k$ is unknown for $k > 1$. The main result of Pagès and Sagna [2015] shows that if one uses the previously quantized distribution of $\widehat{X}_k$, instead of the continuous distribution of $\overline{X}_k$, the resultant procedure converges. Furthermore, the error associated with this procedure is bounded by a constant, which is dependent on the parameters used; see Section 3.2.
This gives rise to the following recursive procedure:

$$\begin{aligned} &\widetilde{X}_0 := \overline{X}_0, \\ &\widehat{X}_k := \pi_{\Gamma_k}(\widetilde{X}_k) \qquad \text{and} \qquad \widetilde{X}_{k+1} = \mathcal{U}(\widehat{X}_k, z_{k+1}) \end{aligned}$$

for $k = 0, \dots, n-1$.

The effect of this algorithm is to approximate the integral in (3.2) as a sum over the codewords in the quantizer Γ_k and their associated probabilities,

$$\begin{aligned} \overline{D}(\Gamma_{k+1}) &= \mathbb{E}\Big[\big|\overline{X}_{k+1} - \pi_{\Gamma_{k+1}}(\overline{X}_{k+1})\big|^2\Big] \\ &\approx \mathbb{E}\Big[\big|\widetilde{X}_{k+1} - \widehat{X}_{k+1}\big|^2\Big] \\ &= \sum_{i=1}^{N_k} \mathbb{E}\Big[\big|\mathcal{U}(\gamma_k^i, z_{k+1}) - \widehat{X}_{k+1}\big|^2\Big]\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big) \\ &=: D(\Gamma_{k+1}). \end{aligned} \tag{3.3}$$

Here, N_k is the cardinality of the quantizer $\Gamma_k = \{\gamma_k^1, \dots, \gamma_k^{N_k}\}$ at time step k, which is allowed to vary. With this definition of $D\big(\Gamma_{k+1}\big)$, the numerical methods outlined in Section 2.3 may now be used to compute the quantizer at time-step $k+1$, which minimizes this distortion.

Note that the distribution of the random variable defined in the recursive algorithm, $\widetilde{X}_{k+1}$, can be found explicitly, and approximates the distribution of $\overline{X}_{k+1}$ as intended,

$$\begin{aligned} F_{\widetilde{X}_{k+1}}(x) &= \sum_{i=1}^{N_k} \mathbb{P}(\mathcal{U}(\gamma_k^i, z_{k+1}) \le x)\mathbb{P}(\widehat{X}_k = \gamma_k^i) \\ &\approx \int_{\mathbb{R}^d} \mathbb{P}(\mathcal{U}(s, z_{k+1}) \le x)\, d\mathbb{P}(\overline{X}_k \le s) \\ &= F_{\overline{X}_{k+1}}(x). \end{aligned} \tag{3.4}$$

3.2 Error Analysis

The error bound for the recursive marginal quantization procedure is governed by Theorem 3.2.1 below, which forms the primary mathematical result of Pagès and Sagna [2015].

Theorem 3.2.1 (Error Bound for Recursive Marginal Quantization)**.** *Let a and b from (3.1) be measurable and satisfy the uniform global Lipschitz continuity assumption, i.e., for every $x,\ y \in \mathbb{R}^d$*

$$|a(x) - a(y)| \le [a]_{\text{Lip}}|x - y| \qquad \text{and} \qquad |b(x) - b(y)| \le [b]_{\text{Lip}}|x - y|,$$

where the matrix norm is defined by $|M| := \sqrt{\text{Tr}(MM^\top)}$ for $M \in \mathbb{R}^{d\times q}$. For every $k = 0, \dots, n$ let Γ_k be a quadratic optimal quantizer for $\widetilde{X}_k$ with cardinality N_k. Then, for every $k = 0, \dots, n$

and for every $\eta \in (0,1]$,

$$\|\overline{X}_k - \widehat{X}_k\|_2 \leq K_{d,\eta} \sum_{l=1}^{k} a_l(a,b,t_k,\Delta t, x_0, L, 2+\eta) N_l^{-\frac{1}{d}},$$

where $K_{d,\eta}$ *is the universal constant defined in Theorem 2.2.5, and, for every* $p \in (2,3]$,

$$a_l(a,b,t_k,\Delta t, x_0, L, p) := e^{C_{a,b}\frac{(t_k - t_l)}{(p)}} \left[e^{(\kappa_p + K_p)t_l} |x_0|^p + \frac{e^{\kappa_p \Delta t} L + K_p}{\kappa_p + K_p} (e^{(\kappa_p + K_p)t_l} - 1) \right]^{\frac{1}{p}},$$

with

$$C_{a,b} := [a]_{\text{Lip}} + \frac{1}{2}[b]^2_{\text{Lip}}, \qquad \kappa_p := \frac{(p-1)(p-2)}{2} + 2pL,$$
$$K_p := 2^{p-1} L^p (1 + p + \Delta t^{\frac{p}{2}-1}) \mathbb{E}[|z|^p], \qquad L := \max([a]_{\text{Lip}}, [b]_{\text{Lip}})$$

and $z \sim \mathcal{N}(0, \mathbf{I}_q)$.

Proof. See Pagès and Sagna [2015, Sec. 3.2]. □

A thorough investigation of this error is left for Section 4.2, where a more general error bound is established for the one-dimensional case. It is, however, important to note that the real coefficients, $a_l(\cdot)$, do not explode when n goes to infinity.

Furthermore, Theorem 3.2.1 establishes an error bound for the quantization of the Euler process; it does not consider the true process at all. Thus, RMQ relies on the convergence properties of the Euler process to ensure that the resultant quantization grid approximates the solution of the original SDE. This is exploited in Chapter 4 when recursive marginal quantization is applied to higher-order schemes.

3.3 Numerical Methods

When X_t is a random vector in $\mathbb{R}^d$, it is possible to use the randomized Lloyd's algorithm, or CLVQ, to minimize the distortion given by (3.3). However, both methods require sampling from the distribution of $\widetilde{X}_k$, see (3.4). This can be done using rejection-type methods or Markov chain Monte Carlo methods, but these quickly reduce the efficiency of the RMQ algorithm to the point where it is not favourably comparable with traditional Monte Carlo schemes. Thus, it is only possible to talk of *fast* recursive marginal quantization in one dimension, where the Newton-Raphson algorithm in $\mathbb{R}^{N_k}$ can be used at each time step. An alternative fast algorithm for the case when $X_t \in \mathbb{R}^2$, specifically the case when X describes a stochastic volatility process, is presented in Chapter 5.

Before deriving the necessary expressions for the Newton-Raphson algorithm, an intuitive explanation of the RMQ algorithm can be provided. Figure 3.1 is a depiction of the process that occurs. The top panel shows the quantizer at time step k. Conditional on each codeword, a Gaussian Euler update is propagated (second panel). In panel three, these updates are weighted by the probability of the associated originating codeword and summed to produce the implied marginal density at time step $k+1$, as shown in the final panel. The distribution associated with this marginal density is $F_{\widetilde{X}_{k+1}}$, which is quantized to produce the quantizer at time step $k+1$. This

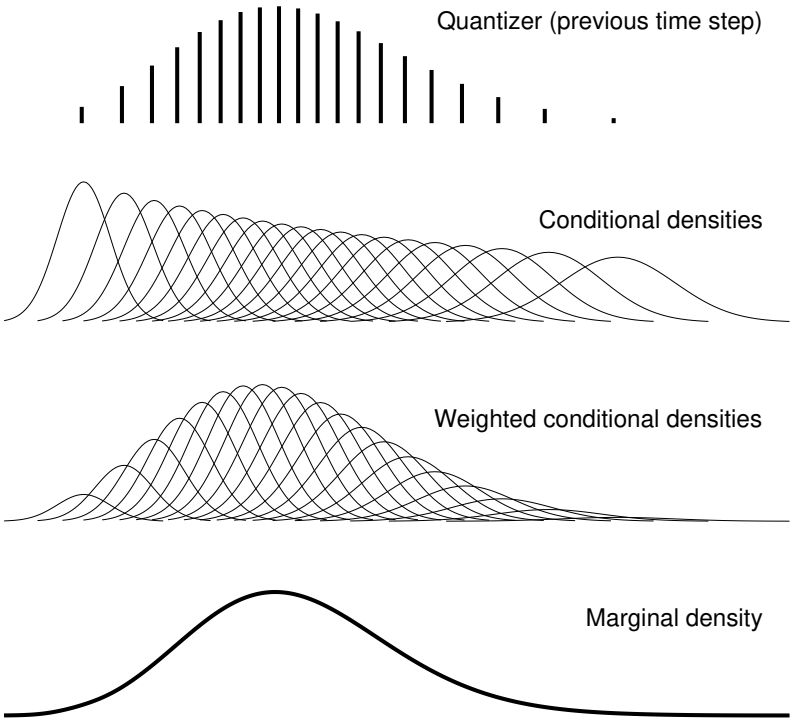

Figure 3.1: Illustration of the RMQ algorithm.

process is repeated until the quantizer at the final time is produced. The output of the algorithm is an inhomogenous Markov chain, depicted as a multinomial tree in Figure 3.2.

Given the quantizer at time t_k, represented as a column vector $\mathbf{\Gamma}_k$, and the associated probabilities, $\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big)$ for $1 \le i \le N_k$, the Newton-Raphson iteration to obtain the quantizer $\mathbf{\Gamma}_{k+1}$, at time t_{k+1}, is given by

$$ {}^{(l+1)}\mathbf{\Gamma}_{k+1} = {}^{(l)}\mathbf{\Gamma}_{k+1} - \left[\nabla^2 D\left({}^{(l)}\mathbf{\Gamma}_{k+1}\right)\right]^{-1} \nabla D\left({}^{(l)}\mathbf{\Gamma}_{k+1}\right), \tag{3.5} $$

where $0 \le l < l_{\max}$ is the iteration index.

The quantities required for the Newton-Raphson iterations are now derived. To summarize notation, the update is written in affine form as

$$ \mathcal{U}(\gamma_k^i, Z_{k+1}^i) =: U_{k+1}^i = m_k^i Z_{k+1}^i + c_k^i, \tag{3.6} $$

where

$$ m_k^i := b(\gamma_k^i)\sqrt{\Delta t} \qquad \text{and} \qquad c_k^i := \gamma_k^i + a(\gamma_k^i)\Delta t, $$

with $Z_{k+1}^i \sim \mathcal{N}(0,1)$ identically distributed to z_{k+1}. Here, a new index i is introduced for the random variable Z_{k+1}^i anticipating that it may depend on γ_k^i. This is redundant in the case of the Euler update because Z_{k+1}^i is a standard Gaussian random variate irrespective of starting point. This more general notation will become necessary in Chapter 4. The corresponding density and distribution functions are denoted by $f_{Z_{k+1}^i}$ and $F_{Z_{k+1}^i}$, respectively.

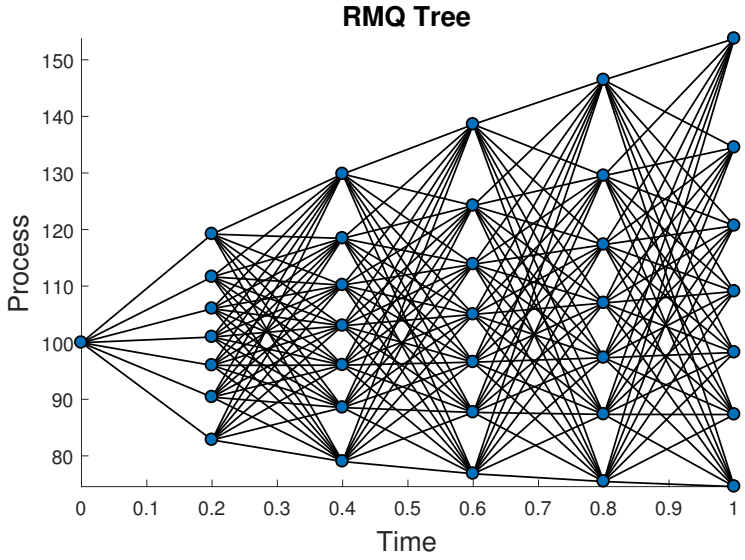

Figure 3.2: RMQ produces an inhomogeneous Markov chain. The grid represented here has 5 time steps and 7 codewords per time step.

With this notation in place, the distribution of $\widetilde{X}_{k+1}$ from (3.4) becomes

$$\begin{aligned} F_{\widetilde{X}_{k+1}}(x) &= \sum_{i=1}^{N_k} \mathbb{P}(U_{k+1}^i \leq x)\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big) \\ &= \sum_{i=1}^{N_k} \left[\mathbb{I}_{\{m_k^i<0\}} + \operatorname{sgn}(m_k^i) F_{Z_{k+1}^i}\left(\frac{x-c_k^i}{m_k^i}\right)\right] \mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big), \end{aligned} \tag{3.7}$$

where $\operatorname{sgn}(\cdot)$ is the signum function. This follows due to the fact that the left-hand probability on the first line may be written as

$$\mathbb{P}(U_{k+1}^i \leq x) = \begin{cases} \mathbb{P}\left(Z_{k+1}^i \leq \frac{x-c_k^i}{m_k^i}\right) & \text{for } m_k^i \geq 0 \\ 1 - \mathbb{P}\left(Z_{k+1}^i \leq \frac{x-c_k^i}{m_k^i}\right) & \text{for } m_k^i < 0. \end{cases}$$

It should be noted that, for the Euler update, m_k^i is a proxy for the diffusion part of the SDE and its positivity is usually guaranteed, in which case (3.7) may be simplified. However, this formulation is necessary because in the general case m_k^i may not be guaranteed to be positive.

The elements of the gradient of the distortion $\nabla D(\Gamma_{k+1})$ may then be written as

$$\begin{aligned} \frac{\partial D\left(\Gamma_{k+1}\right)}{\partial \gamma_{k+1}^j} &= 2\sum_{i=1}^{N_k} \mathbb{E}\Big[\mathbb{I}_{\{U_{k+1}^i \in R^j(\Gamma_{k+1})\}}\left(\gamma_{k+1}^j - U_{k+1}^i\right)\Big]\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big) \\ &= 2\sum_{i=1}^{N_k} \int_{U_{k+1}^i \in R^j(\Gamma_{k+1})} \left(\gamma_{k+1}^j - U_{k+1}^i\right) d\mathbb{P}(Z_{k+1}^i \leq x)\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big), \end{aligned} \tag{3.8}$$

where $1 \leq j \leq N_{k+1}$ is the index tracking the elements of the t_{k+1} quantizer, and the index associated with the t_k quantizer is i. The integration bounds in (3.8) must now be expressed in terms of the variable of integration.

Here, as in Section 2.3.2, $U_{k+1}^i \in R^j(\Gamma_{k+1})$ is equivalent to the inequality

$$r_{k+1}^{j-} < U_{k+1}^i \leq r_{k+1}^{j+} \qquad \text{with} \qquad r_{k+1}^{j\pm} = \tfrac{1}{2}(\gamma_{k+1}^{j\pm 1} + \gamma_{k+1}^j), \tag{3.9}$$

and $r_{k+1}^{1-} = -\infty$ and $r_{k+1}^{N_{k+1}+} = \infty$ by definition. Defining the standardized region boundaries,

$$r_{k+1}^{i,j\pm} = \frac{r_{k+1}^{j\pm} - c_k^i}{m_k^i}, \tag{3.10}$$

allows the inequality to be written in terms of the Gaussian random variable Z_{k+1}^i as

$$U_{k+1}^i \in R^j(\Gamma_{k+1}) = \begin{cases} r_{k+1}^{i,j-} < Z_{k+1}^i \leq r_{k+1}^{i,j+} & \text{for } m_k^i \geq 0 \\ r_{k+1}^{i,j-} > Z_{k+1}^i \geq r_{k+1}^{i,j+} & \text{for } m_k^i < 0, \end{cases}$$

which can now be used as the range over which the integration is taken.

Directly evaluating (3.8), each element of the gradient of the distortion at time t_{k+1} is given by

$$\begin{aligned} \frac{\partial D(\Gamma_{k+1})}{\partial \gamma_{k+1}^j} = 2\sum_{i=1}^{N_k} \Big[& (\gamma_{k+1}^j - c_k^i)\operatorname{sgn}(m_k^i)\left(F_{Z_{k+1}^i}(r_{k+1}^{i,j+}) - F_{Z_{k+1}^i}(r_{k+1}^{i,j-})\right) \\ & -|m_k^i|\left(M^1_{Z_{k+1}^i}(r_{k+1}^{i,j+}) - M^1_{Z_{k+1}^i}(r_{k+1}^{i,j-})\right)\Big]\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big). \end{aligned}$$

Furthermore, the diagonal of the tridiagonal Hessian, $\nabla^2 D(\Gamma_{k+1})$, is given by

$$\begin{aligned} \frac{\partial^2 D(\Gamma_{k+1})}{\partial \big(\gamma_{k+1}^j\big)^2} = \sum_{i=1}^{N_k} \Big[& 2\operatorname{sgn}(m_k^i)\left(F_{Z_{k+1}^i}(r_{k+1}^{i,j+}) - F_{Z_{k+1}^i}(r_{k+1}^{i,j-})\right) \\ & + \frac{1}{2|m_k^i|} f_{Z_{k+1}^i}(r_{k+1}^{i,j+})(\gamma_{k+1}^j - \gamma_{k+1}^{j+1}) \\ & + \frac{1}{2|m_k^i|} f_{Z_{k+1}^i}(r_{k+1}^{i,j-})(\gamma_{k+1}^{j-1} - \gamma_{k+1}^j)\Big]\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big), \end{aligned}$$

with the super-diagonal and sub-diagonal elements given by

$$\frac{\partial^2 D(\Gamma_{k+1})}{\partial \gamma_{k+1}^j \partial \gamma_{k+1}^{j+1}} = \sum_{i=1}^{N_k} \frac{1}{2|m_k^i|} f_{Z_{k+1}^i}(r_{k+1}^{i,j+})(\gamma_{k+1}^j - \gamma_{k+1}^{j+1})\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big)$$

and

$$\frac{\partial^2 D(\Gamma_{k+1})}{\partial \gamma_{k+1}^j \partial \gamma_{k+1}^{j-1}} = \sum_{i=1}^{N_k} \frac{1}{2|m_k^i|} f_{Z_{k+1}^i}(r_{k+1}^{i,j-})(\gamma_{k+1}^{j-1} - \gamma_{k+1}^j)\mathbb{P}\big(\widehat{X}_k = \gamma_k^i\big),$$

respectively.

Although these expressions may appear complicated, they are simply summations over the density function, cumulative distribution function and first lower partial expectation of the random variable, Z_{k+1}^i, and are thus easy to compute when these functions are known.

All the detail required to implement the Newton iteration (3.5) has now been provided with the exception of the initial guess. In all applications considered, $N_k = N$ for $1 \leq k \leq n$, and the quantizer from the previous time step was used as the initial guess, i.e., ${}^{(0)}\boldsymbol{\Gamma}_{k+1} = \boldsymbol{\Gamma}_k$. An efficient matrix formulation of the above equations is now provided in order to facilitate the implementation.

3.3.1 Matrix Formulation

As in Section 2.3.3, where an efficient matrix formulation for the Newton-Raphson iteration required for VQ was provided, RMQ is also amenable to a matrix specification. This aids simple and computationally efficient implementation.

Aside from an initial guess for $\mathbf{\Gamma}_{k+1}$, three time-indexed column vectors are required. The vectors

$$[\mathbf{m}_k]_i = m_k^i \quad \text{and} \quad [\mathbf{c}_k]_i = c_k^i,$$

defined for $1 \leq i \leq N_k$, and the vector

$$[\Delta\mathbf{\Gamma}_{k+1}]_i = \gamma_{k+1}^{i+1} - \gamma_{k+1}^i,$$

defined for $1 \leq i \leq (N_{k+1} - 1)$. The row vector of probabilities

$$\mathbf{p}_k = [\mathbb{P}(\widehat{X}_k = \gamma_k^1), \ldots, \mathbb{P}(\widehat{X}_k = \gamma_k^{N_k})], \tag{3.11}$$

is retained and a row-vector of ones of length d is denoted by $\mathbf{j}_d$. With the exception of $\Delta\mathbf{\Gamma}_{k+1}$, which must be recomputed before each Newton-Raphson iteration, the other vectors are computed once per time step.

Before each Newton-Raphson iteration, three matrices must be computed in terms of the new estimate of $\mathbf{\Gamma}_{k+1}$: an $N_k \times N_{k+1}$ matrix of transition probabilities

$$\begin{aligned}[\mathbf{P}_{k+1}]_{i,\,j} &= \mathbb{P}(\widehat{X}_{k+1} = \gamma_{k+1}^j | \widehat{X}_k = \gamma_k^i) \\ &= \operatorname{sgn}(m_k^i) \left[F_{Z_{k+1}^i}(r_{k+1}^{i,j+}) - F_{Z_{k+1}^i}(r_{k+1}^{i,j-}) \right], \end{aligned} \tag{3.12}$$

another matrix, of the same size, of the lower partial moment values

$$[\mathbf{M}_{k+1}]_{i,j} = M^1_{Z_{k+1}^i}(r_{k+1}^{i,j+}) - M^1_{Z_{k+1}^i}(r_{k+1}^{i,j-}) \tag{3.13}$$

and an $N_k \times (N_{k+1} - 1)$ matrix of density values at the positive region boundaries

$$[\mathbf{f}_{k+1}]_{i,j} = f_{Z_{k+1}^i}(r_{k+1}^{i,j+}).$$

The gradient of the distortion function at time step $k+1$ may then be written in terms of these vectors and matrices as

$$\nabla D\left(\mathbf{\Gamma}_{k+1}\right)^\top = 2\mathbf{p}_k \left(\left((\mathbf{\Gamma}_{k+1}\mathbf{j}_{N_k})^\top - \mathbf{c}_k \mathbf{j}_{N_{k+1}} \right) \circ \mathbf{P}_{k+1} - (|\mathbf{m}_k| \mathbf{j}_{N_{k+1}}) \circ \mathbf{M}_{k+1} \right), \tag{3.14}$$

where $\circ$ is the Hadamard (or element-wise) product.

The super and sub-diagonal elements of the (tridiagonal) Hessian matrix, $\nabla^2 D\left(\mathbf{\Gamma}_{k+1}\right)$, are given by the vector

$$\mathbf{h}_{\mathsf{off}} = -\tfrac{1}{2}\mathbf{p}_k \left(\left(|\mathbf{m}_k|^{\circ -1} \mathbf{j}_{(N_{k+1}-1)} \right) \circ \mathbf{f}_{k+1} \circ (\Delta\mathbf{\Gamma}_{k+1}\mathbf{j}_{N_k})^\top \right), \tag{3.15}$$

while the main diagonal is given by

$$\mathbf{h}_{\mathsf{main}} = 2\mathbf{p}_k\mathbf{P}_{k+1} + \left[\mathbf{h}_{\mathsf{off}} | 0\right] + \left[0 | \mathbf{h}_{\mathsf{off}}\right], \tag{3.16}$$

where $\circ -1$ in the exponent refers to the element-wise inverse.

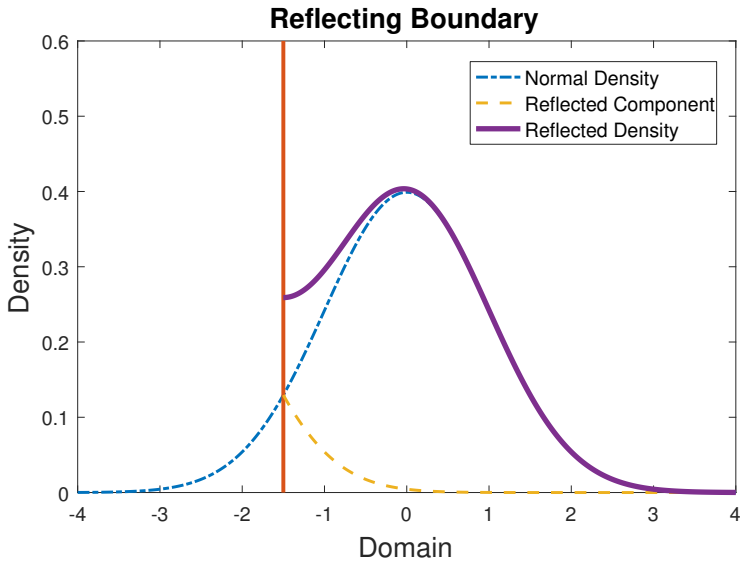

Figure 3.3: Illustration of the standard Gaussian density reflected around -1.5.

Equations (3.14), (3.15) and (3.16) provide the necessary components required for implementation of the Newton-Raphson iteration derived in the previous section. After the requisite number of iterations, the probabilities associated with the final quantizer $\mathbf{\Gamma}_{k+1}$ are computed using

$$\mathbf{p}_{k+1} = \mathbf{p}_k \mathbf{P}_{k+1},$$

where $\mathbf{P}_{k+1}$ must be recomputed in terms of the final $\mathbf{\Gamma}_{k+1}$. Thus, the matrix formulation presented here allows RMQ to be interpreted as the propagation of an inhomogeneous, discrete time Markov chain, where $\mathbf{\Gamma}_k$ represents the Markov states at time-step k, the probability of being in those states is $\mathbf{p}_k$, and the associated transition probability matrix is $\mathbf{P}_{k+1}$. Sometimes in the literature the transition probability matrix between time-step k and time-step $k+1$ is represented as $\mathbf{P}_{k,k+1}$; in this work the first index is omitted.

3.4 The Zero Boundary

Sometimes common discrete-time approximations of an SDE may exhibit behaviour that is inconsistent with the true solution. For example, an Euler-Maruyama approximation of geometric Brownian motion or the CEV process can, under certain circumstances, generate negative values, even though the SDE specification guarantees non-negativity in each case. As a result, discrete-time Monte Carlo simulations are often modified to generate reflecting or absorbing behaviour at zero; see for example Lord et al. [2010]. This section describes how the RMQ algorithm may be modified in a similar manner.

3.4.1 Absorbing Boundary

To model an absorbing boundary, the domain of the approximate marginal distribution of $\overline{X}$, see (3.7), must be left-truncated at zero. Implementing the RMQ algorithm with a left limit of zero results in a quantizer that has probabilities at each time step that do not sum to unity. The probability that is not accounted for as a result of the domain truncation is the mass accumulated at the absorbing zero boundary. To compensate for this, the quantizer at each time step is augmented

with an extra codeword, which has a value of zero and a probability equal to one minus the sum of the probabilities associated with all the other codewords at that time step. The transition probability matrix is augmented in a consistent manner by realising that once the process attains the zero state it must remain in that state indefinitely, that is, the conditional probability of moving from the absorbing state to any other state is zero, and, correspondingly, the conditional probability of remaining in the absorbing state is one.

Modifying the algorithm is straightforward and incurs no additional computational burden. Given that the elements of the previous quantizer $\mathbf{\Gamma}_k$ are all positive, the affine form of the update (3.6), will be negative when

$$Z_{k+1}^i < -\frac{c_k^i}{m_k^i}.$$

This implies that the domain of each Z_{k+1}^i must be left-truncated at $-\frac{c_k^i}{m_k^i}$ to ensure only positive codewords at time-step $k+1$. This is achieved by setting

$$r_{k+1}^{i,1-} = -\frac{c_k^i}{m_k^i}, \tag{3.17}$$

for $1 \leq i \leq N_k$, in (3.10). This is equivalent to assuming $r_{k+1}^{1-} = 0$ in (3.9). The rest of the algorithm proceeds without modification.

Of course, this all depends on the fact that the quantizer at the first time step, $\mathbf{\Gamma}_1$, also has only positive elements. This is achieved by using an analogous truncation in the vector quantization algorithm. The initial guess for the Newton iteration must also ensure positivity.

3.4.2 Reflecting Boundary

Figure 3.3 shows a density function, $f(x)$, — in this case a standard Gaussian density. The red line represents a reflecting boundary at $\bar{x} = -1.5$. The values f takes to the left of the boundary are reflected and depicted by the dashed yellow line in the figure. These reflected values are given by $f(2\bar{x} - x)$ for $x > \bar{x}$.

Thus, restricting the domain to $[\bar{x}, \infty)$ the reflected density, denoted $f^{\mathrm{R}}(x)$, is given as the sum of the blue line, the density itself, and the dashed yellow line, the reflected component,

$$f^{\mathrm{R}}(x) = f(x) + f(2\bar{x} - x).$$

Direct integration of this expression over the integration limits from $\bar{x}$ to $x \in [\bar{x}, \infty)$ gives the reflected distribution function

$$F^{\mathrm{R}}(x) = F(x) - F(2\bar{x} - x)$$

and the reflected first lower partial expectation function

$$M^{\mathrm{R}}(x) = M^1(x) + M^1(2\bar{x} - x) - 2\bar{x}F(2\bar{x} - x) - 2M^1(\bar{x}) + 2\bar{x}F(\bar{x}), \tag{3.18}$$

where $F(x)$ and $M^1(x)$ are the un-reflected distribution and first lower expectation functions associated with f.

Modifying the RMQ algorithm to allow for a reflecting boundary at zero requires two changes to the implementation. Firstly, the lower bound for the integration, that is, the domain of the Z_{k+1}^i random variable in each affine update, must be left-truncated by replacing the furthest left region boundary as in (3.17) above. Secondly, the density, distribution and first lower partial expectation

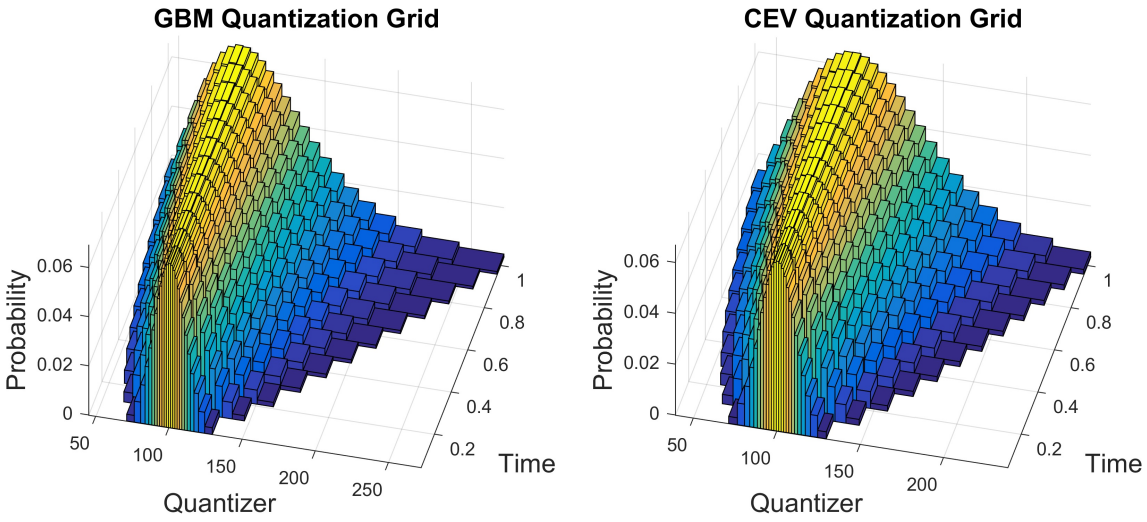

Figure 3.4: Recursive marginal quantization of the GBM and CEV models.

functions associated with each random variable, must be replaced by their reflected counterparts

$$f^{\mathrm{R}}_{Z^i_{k+1}}(x) = f_{Z^i_{k+1}}(x) + f_{Z^i_{k+1}}(2\bar{x}^i_k - x),$$
$$F^{\mathrm{R}}_{Z^i_{k+1}}(x) = F_{Z^i_{k+1}}(x) - F_{Z^i_{k+1}}(2\bar{x}^i_k - x),$$

and

$$M^{\mathrm{R}}_{Z^i_{k+1}}(x) = M^1_{Z^i_{k+1}}(x) + M^1_{Z^i_{k+1}}(2\bar{x}^i_k - x) - 2\bar{x}^i_k F_{Z^i_{k+1}}(2\bar{x}^i_k - x), \tag{3.19}$$

for $x \in [\bar{x}^i_k, \infty)$, where $\bar{x}^i_k = -\frac{c^i_k}{m^i_k}$. The remainder of the algorithm proceeds as normal. An attentive reader will have noticed that there are two terms missing in (3.19) when compared with (3.18). The reason for this omission is that these terms are constants for each i and that the RMQ algorithm always only requires differences of partial moment terms. There is, therefore, a cancelation of the constant terms when this difference is taken and thus they are excluded.

As in the case of the absorbing boundary, the analogous reflection must be applied in the vector quantization algorithm to ensure that $\mathbf{\Gamma}_1$ is consistent.

3.5 Examples

In this section, the RMQ algorithm is illustrated for geometric Brownian motion (GBM) and its generalization, the constant elasticity of variance model (CEV). The SDE for GBM may be specified in the notation of (3.1) as

$$a(X_t) = rX_t, \qquad b(X_t) = \sigma X_t, \tag{3.20}$$

and, in the considered example, the model-specific parameters chosen are $X_0 = 100$, $r = 5\%$ and $\sigma = 30\%$. The SDE for the CEV model may be specified in the notation of (3.1) as

$$a(X_t) = rX_t, \qquad b(X_t) = \sigma_{\mathrm{CEV}} X_t^{\alpha}, \tag{3.21}$$

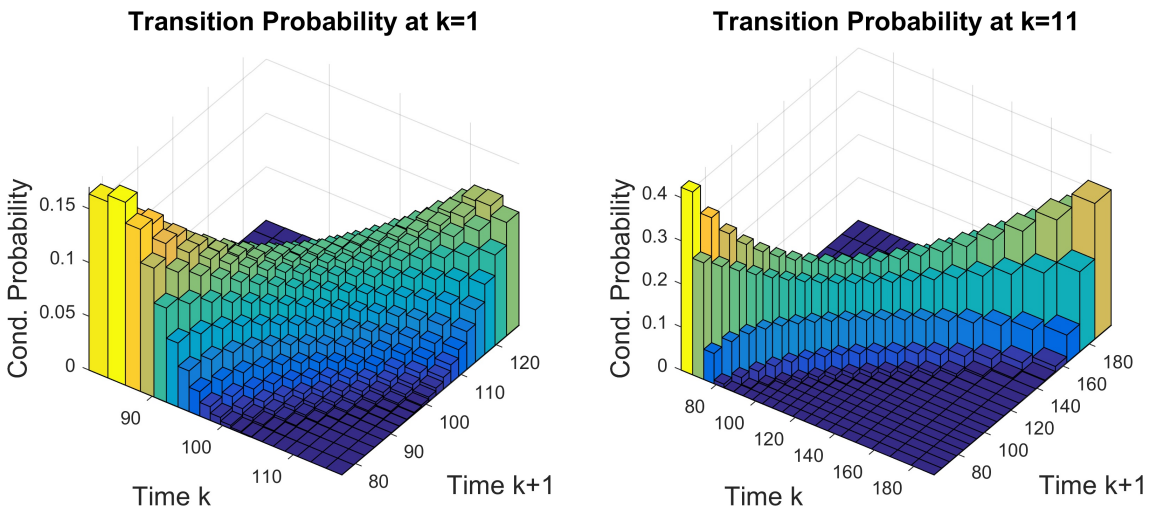

Figure 3.5: RMQ transition probabilities for the GBM model.

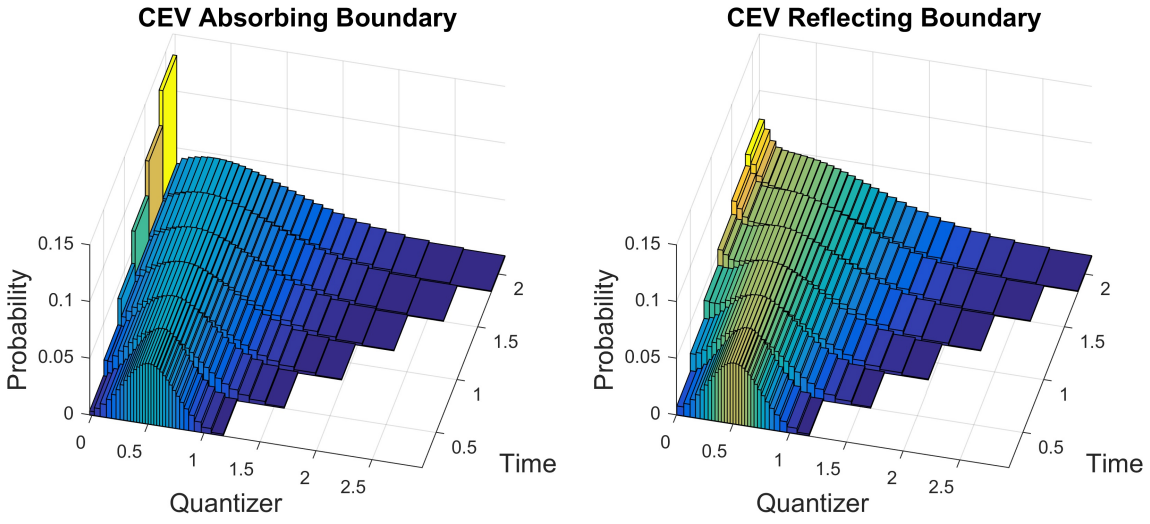

Figure 3.6: Recursive marginal quantization of the CEV model with an absorbing (left panel) and reflecting (right panel) boundary.

with the additional parameters chosen as $\alpha = 0.7$ and $\sigma_{\text{CEV}} = \sigma X_0^{1-\alpha}$, with σ_{CEV} given in terms of the instantaneous log-normal volatility $\sigma = 30\%$.

For a time-horizon of $T = 1$ year, with monthly time-steps, i.e., $n = 12$, and a constant cardinality of $N_k = 25$ for all k, the series of optimal quantizers generated by the RMQ algorithm is depicted in Figure 3.4 for the above models. Please note that although 3D bars are used to indicate the size of the regions associated with each of the codewords, the codewords themselves are, of course, point masses. The left panel of Figure 3.4 clearly depicts the heavy tail expected of the true log-normal distribution whereas the right panel shows that the marginal distributions of the CEV model have lower variance than those of the GBM model for these parameters, at least out to one year.

As part of the RMQ algorithm, the matrix of transition probabilities is computed at each time-step, see (3.12). The left panel of Figure 3.5 shows the transition probabilities of the GBM model when moving from the first month to the second month of the year, whereas the right panel shows the transition probabilities from the second-last month to the last month. Here it can be seen that, as the codewords spread out through time, the transition probabilities remain concentrated around the mean of the distribution.

It is well known that when $0 < \alpha < 0.5$, the CEV process may attain zero and that to account for a unique solution either an absorbing or reflecting boundary condition must be assigned, see Lindsay and Brecher [2012]. For the example above, the CEV process with $\alpha > 0.5$ was considered, which only allows absorption at zero. Now, consider the case where $X_0 = 0.5$, $\alpha = 0.35$ and $\sigma_{\text{LN}} = 65\%$, with the rest of the parameters as before.

Figure 3.6 illustrates the effect of the boundary condition for this CEV process. Here $N_k = 30$ codewords were selected and $n = 6$ time steps were used out to 2 years. The coarse time discretization was selected to make the impact of the boundary condition clearly visible. In the left panel of Figure 3.6 the probability of the process being absorbed at zero is being accumulated over time at the pseudo-codeword at zero, which augments the original quantization grid. In the right panel of Figure 3.6, the quantization grid is not augmented. Instead, the probability that would accumulate at zero for an absorbing boundary condition is reflected at the zero boundary, raising the probabilities of the codewords close to zero.

The effectiveness of the RMQ algorithm with regards to option pricing and approximating the true underlying marginal distributions of the GBM and CEV processes is investigated in the next chapter.

Chapter 4

Recursive Marginal Quantization of Higher-order Schemes

4.1 Overview

The RMQ algorithm in Chapter 3 is presented in more generality than in Pagès and Sagna [2015]. The expressions for the Newton-Raphson algorithm are derived in terms of the distribution function (CDF), density function (PDF) and first lower partial moment of the random variable that appears in (3.6), the affine form of the discrete-time update. The possibility of a sign-reversal in the affine coefficient is also accounted for.

Given this general formulation, two higher-order extensions are now explored: the Milstein scheme and a simplified weak-order 2.0 scheme. Any numerical scheme for an SDE that can be written in the affine form (3.6), may be used with the RMQ algorithm as long as the CDF, PDF and lower partial expectation of the random variable Z_{k+1}^i can be computed for all i and k.

4.1.1 The Milstein Scheme

The Milstein [1975] scheme for the one-dimensional version of the SDE in (3.1) is given by

$$\overline{X}_{k+1} = \overline{X}_k + a(\overline{X}_k)\Delta t + b(\overline{X}_k)\sqrt{\Delta t} z_{k+1} + \tfrac{1}{2} b(\overline{X}_k) b'(\overline{X}_k) \Delta t \left(z_{k+1}^2 - 1\right),$$

for $0 \leq k < n$, where $\Delta t = T/n$ and $z_{k+1} \sim \mathcal{N}(0,1)$ with initial value $\overline{X}_0 = x_0$.

By completion of the square, this may be written as

$$\begin{aligned}\overline{X}_{k+1} = \overline{X}_k + \left(a(\overline{X}_k) - \tfrac{1}{2} b(\overline{X}_k) b'(\overline{X}_k)\right) \Delta t - \tfrac{1}{2} b(\overline{X}_k) b'(\overline{X}_k)^{-1} \\ + \tfrac{1}{2} b(\overline{X}_k) b'(\overline{X}_k) \Delta t \left(z_{k+1} + \left(\sqrt{\Delta t} b'(\overline{X}_k)\right)^{-1} \right)^2.\end{aligned}$$

Thus, the Milstein update may be written in the affine form of (3.6) as

$$U_{k+1}^i = m_k^i Z_{k+1}^i + c_k^i,$$

where

$$m_k^i = \tfrac{1}{2} b(\gamma_k^i) b'(\gamma_k^i) \Delta t$$

and

$$c_k^i = \gamma_k^i + \left(a(\gamma_k^i) - \tfrac{1}{2}b(\gamma_k^i)b'(\gamma_k^i)\right)\Delta t - \tfrac{1}{2}b(\gamma_k^i)b'(\gamma_k^i)^{-1}.$$

The random variable Z_{k+1}^i is now noncentral chi-squared distributed with one degree of freedom and noncentrality parameter

$$\lambda_{k+1}^i = \left(\sqrt{\Delta t}b'(\gamma_k^i)\right)^{-2}.$$

It is important to note that, unlike in the Euler-Maruyama case, the distribution of the random variable $Z_{k+1}^i \sim \chi'^2(1, \lambda_{k+1}^i)$ now depends on the codeword γ_k^i. Note that the PDF, CDF and lower partial expectation of this random variable can be expressed in terms of the PDF and CDF of the standard Gaussian, as shown in Section 2.4.2, which enables efficient implementation.

Although the Milstein scheme possesses a strong order of convergence of 1, compared to the Euler scheme, which only has strong order of convergence of $\frac{1}{2}$, both schemes have a weak-order of convergence of 1. Thus, while the Milstein scheme is more accurate in a strong sense than the Euler scheme, a different update is required to achieve second weak-order convergence in a Monte Carlo simulation. In general, such a higher weak-order scheme improves the approximation of the expectation of financial payoffs.

4.1.2 A Weak Order 2.0 Taylor Scheme

While it is not possible to write a weak-order 2.0 Taylor scheme in the affine form required, the simplified weak-order 2.0 scheme of Kloeden and Platen [1999] is amenable. This scheme is given by

$$\begin{aligned}\overline{X}_{k+1} = \overline{X}_k &+ a(\overline{X}_k)\Delta t + b(\overline{X}_k)\sqrt{\Delta t}z_{k+1} + \tfrac{1}{2}b(\overline{X}_k)b'(\overline{X}_k)\Delta t(z_{k+1}^2 - 1)\\ &+ \tfrac{1}{2}\left(a'(\overline{X}_k)b(\overline{X}_k) + a(\overline{X}_k)b'(\overline{X}_k) + \tfrac{1}{2}b''(\overline{X}_k)b^2(\overline{X}_k)\right)(\Delta t)^{\frac{3}{2}}z_{k+1}\\ &+ \tfrac{1}{2}\left(a(\overline{X}_k)a'(\overline{X}_k) + \tfrac{1}{2}a''(\overline{X}_k)b^2(\overline{X}_k)\right)(\Delta t)^2,\end{aligned}$$

for $0 \leq k < n$, where $\Delta t = T/n$ and $z_{k+1} \sim \mathcal{N}(0,1)$ with initial value $\overline{X}_0 = x_0$. Again, completion of the square is used to write this update in the required affine form,

$$U_{k+1}^i = m_k^i Z_{k+1}^i + c_k^i,$$

where

$$m_k^i = \tfrac{1}{2}b(\gamma_k^i)b'(\gamma_k^i)\Delta t$$

and

$$\begin{aligned}c_k^i = \gamma_k^i &+ \left(a(\gamma_k^i) - \tfrac{1}{2}b(\gamma_k^i)b'(\gamma_k^i)\right)\Delta t + \tfrac{1}{2}\left(a(\gamma_k^i)a'(\gamma_k^i) + \tfrac{1}{2}a''(\gamma_k^i)b^2(\gamma_k^i)\right)(\Delta t)^2\\ &- \frac{\left(b(\gamma_k^i) + \tfrac{1}{2}\left(a'(\gamma_k^i)b(\gamma_k^i) + a(\gamma_k^i)b'(\gamma_k^i) + \tfrac{1}{2}b''(\gamma_k^i)b^2(\gamma_k^i)\right)\Delta t\right)^2}{2b(\gamma_k^i)b'(\gamma_k^i)}.\end{aligned}$$

Here, Z_{k+1}^i is again noncentral chi-squared distributed with one degree of freedom, with noncentrality parameter given by

$$\lambda_{k+1}^i = \left(\frac{b(\gamma_k^i) + \tfrac{1}{2}\left(a'(\gamma_k^i)b(\gamma_k^i) + a(\gamma_k^i)b'(\gamma_k^i) + \tfrac{1}{2}b''(\gamma_k^i)b^2(\gamma_k^i)\right)\Delta t}{b(\gamma_k^i)b'(\gamma_k^i)\sqrt{(\Delta t)}}\right)^2,$$

or, more succinctly, $Z_{k+1}^i \sim \chi'^2(1, \lambda_{k+1}^i)$.

4.2 Error Analysis

In this section, Theorem 3.2.1 is adapted to apply to the higher-order schemes. This provides analytical error bounds for the proposed method. Furthermore, numerical evidence is provided for the improved accuracy and convergence of the higher-order schemes.

4.2.1 Theoretical Error Bounds

Here it is shown that the error bound provided by Theorem 3.2.1 can be adapted to the Milstein higher-order update in the one-dimensional case. The case for the simplified weak-order 2.0 scheme is similar, but requires that the Lipschitz continuity and linear growth assumptions be applied to the second derivatives of the a and b coefficients as well.

The work-horse of the proof is Lemma 4.2.1, adapted from Pagès and Sagna [2015, Lemma 3.1].

Lemma 4.2.1 (Moment Bound for Milstein Update)**.** *In the one-dimensional Milstein-update case, let a and b from (3.1), as well as the derivative b', be measurable and satisfy the uniform global Lipschitz continuity assumption, i.e., for every x, $y \in \mathbb{R}$, there exist positive constants, $[a]_{\text{Lip}}$, $[b]_{\text{Lip}}$ and $[b']_{\text{Lip}}$, such that*

$$|a(x) - a(y)| \leq [a]_{\text{Lip}}|x - y|, \qquad |b(x) - b(y)| \leq [b]_{\text{Lip}}|x - y|,$$

and

$$|b'(x) - b'(y)| \leq [b']_{\text{Lip}}|x - y|.$$

Then, for every $p \in (2, 3]$ and every $k = 0, \ldots, n$,

$$\mathbb{E}\left[|\widetilde{X}_k|^p\right] \leq e^{(\kappa_p' + K_p')t_k}|x_0|^p + \frac{e^{\kappa_p'\Delta t}L + K_p'}{\kappa_p' + K_p'}\left(e^{(\kappa_p' + K_p')t_k} - 1\right), \tag{4.1}$$

with

$$\kappa_p' := \frac{(p-1)(p-2)}{2}\Delta t + 2pL, \qquad K_p' := 2^p L^p(\Delta t(p-1) + (\Delta t)^{p-1})\mathbb{E}[|Z|^p],$$
$$L := \max([a]_{\text{Lip}}, [b]_{\text{Lip}}, [b']_{\text{Lip}})$$

where Z is a noncentral chi-squared random variable with one degree of freedom and positive and finite non-centrality.

Proof. The proof closely follows the proof of Pagès and Sagna [2015, Lem. 3.1] and appears in Appendix A.2. □

Note that the coefficients defined above differ slightly from those used in Theorem 3.2.1, hence the prime superscripts.

Theorem 4.2.2 (Error Bound for Milstein Update)**.** *In the one-dimensional Milstein-update case, let a and b from (3.1), as well as the derivative b', be measurable and satisfy the uniform global Lipschitz continuity assumption as stated in Lemma 4.2.1. For every $k = 0, \ldots, n$ let Γ_k be a*

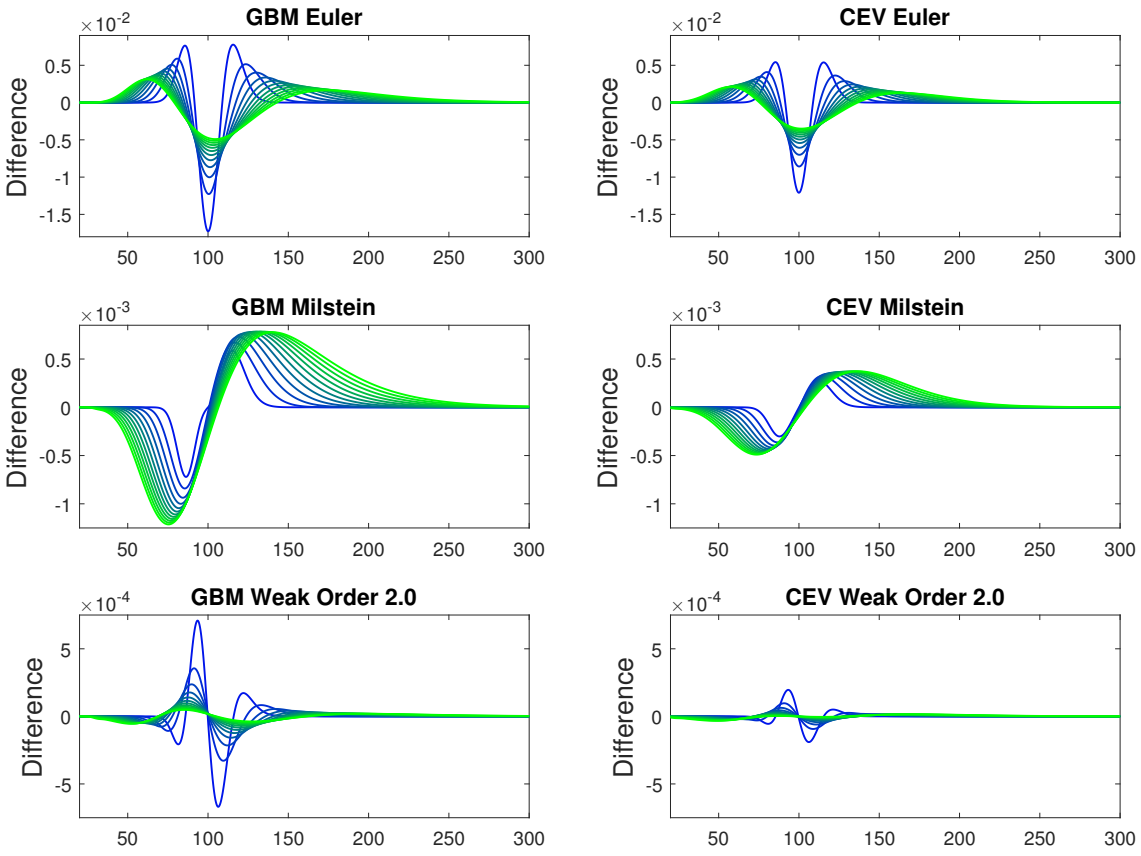

Figure 4.1: The error in the marginal distribution implied by quantization for GBM and CEV.

quadratic optimal quantizer for $\widetilde{X}_k$ *with cardinality* N*. Then, for every* $k = 1, \ldots, n$ *and* $\delta \in (0, 1]$*,*

$$\|\overline{X}_k - \widehat{X}_k\|_2 \leq K_{1,\delta} \sum_{i=1}^{k} a_i([a]_{\text{Lip}}, [b]_{\text{Lip}}, \Delta t, x_0, L, 2+\delta) N^{-1} \tag{4.2}$$

with $K_{1,\delta}$ *the universal constant from Theorem 2.2.5 and*

$$a_i([a]_{\text{Lip}}, [b]_{\text{Lip}}, \Delta t, x_0, L, p) :=$$
$$e^{\left([a]_{\text{Lip}} + \frac{1}{2}[b]^2_{\text{Lip}}\right)\frac{\Delta t(k-i)}{p}} \left[e^{(\kappa'_p + K'_p)k\Delta t} |x_0|^p + \frac{e^{\kappa'_p \Delta t} L + K'_p}{\kappa'_p + K'_p} \left(e^{(\kappa'_p + K'_p)k\Delta t} - 1 \right) \right]^{\frac{1}{p}}.$$

Proof. The proof closely follows the proof of Pagès and Sagna [2015, Thm. 3.1] and appears in Appendix A.2. □

4.2.2 Numerical Evidence

To illustrate the accuracy of the above schemes, the RMQ algorithm is again applied to geometric Brownian motion, as previously described by (3.20), and the constant elasticity of variance model, given by (3.21). The model-specific parameters from Section 3.5 are retained. The time-horizon is set at $T = 1$ year, and $n = 12$ time steps were used in the RMQ algorithm.

Since the true conditional distributions for GBM and CEV are known in closed form, the approximate marginal distributions at time step $k+1$, given in (3.4), as implied by the quantizer at time step k, can be compared with the exact distributions at time step $k+1$. In the case of the

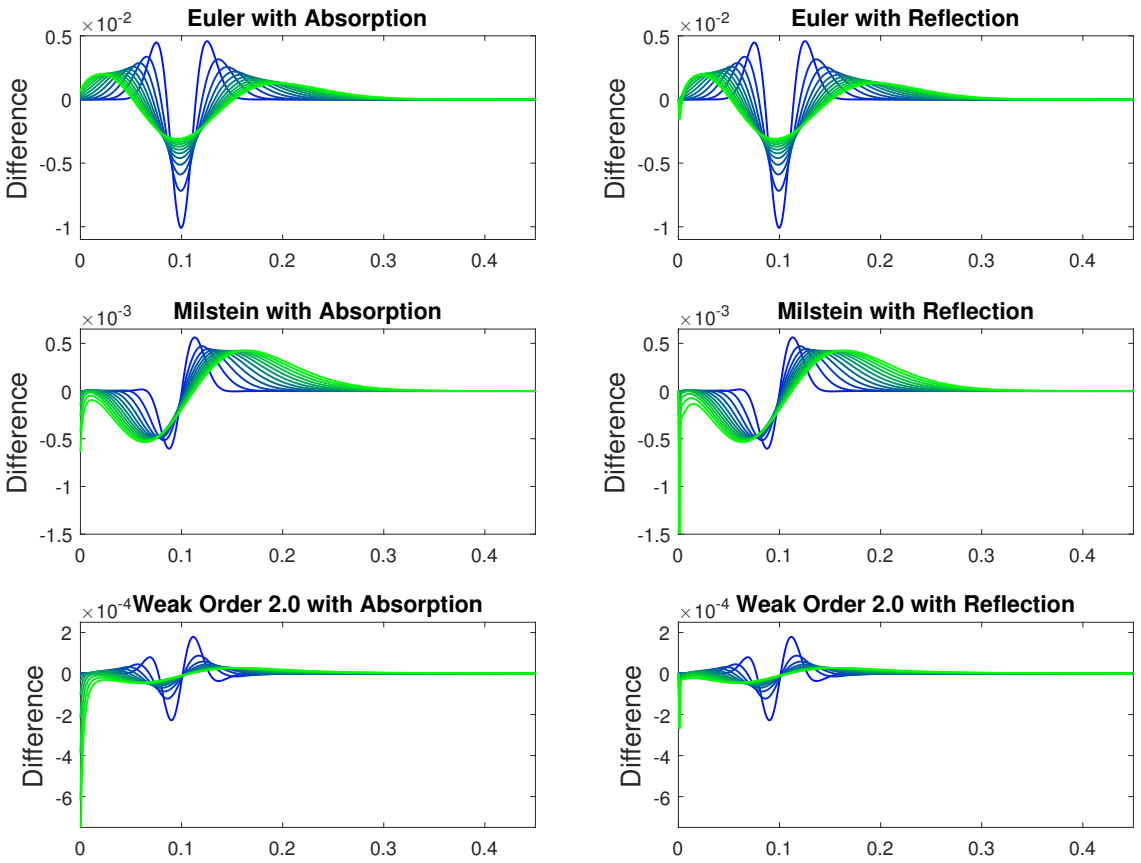

Figure 4.2: The difference between the true and approximate marginal distributions for the CEV process. The left panel shows the case of absorption while the right panel shows reflection.

CEV process, the analytical expression for the distribution given by Lindsay and Brecher [2012] was used, adjusted for the drift component in the SDE.

For each of the three schemes, under the GBM and CEV test cases, the difference between the exact marginal distribution and the implied marginal distribution is plotted in Figure 4.1. Note that the maximum possible error is 1. In these plots, the colour of the quantizer indicates the associated time step, with lines in blue closer to initial time and lines in green closer to final time. This convention is kept throughout. Here a cardinality of $N_k = 200$ was used for all $k = 0, \ldots, n$.

Note the scale of the y-axes of the graphs in the figure — from top to bottom, the magnitude of the error decreases by an order of magnitude in each successive row. This gives an indication of the improvement that can be expected when these higher order schemes are used to price contingent claims.

In a similar way, the effect of modelling the zero boundary, as outlined in Section 3.4, can be investigated for the CEV process. Consider again the case where $X_0 = 0.5$, $\alpha = 0.35$ and $\sigma_{\text{LN}} = 65\%$, with the rest of the parameters as before. Figure 4.2 shows the difference between the exact marginal distribution of the CEV process and the marginal distribution implied by the quantization, for the three update schemes, as modified to account for an absorbing boundary (left panel) and a reflecting boundary (right panel). Note that, although the RMQ algorithm appears to systematically underestimate the probabilities of the codewords close to zero, the scale of the error in the implied marginal distribution is in line with that of Figure 4.1, the case without a boundary condition.

In Figure 4.3 numerical evidence for weak-order convergence is provided. The absolute value of

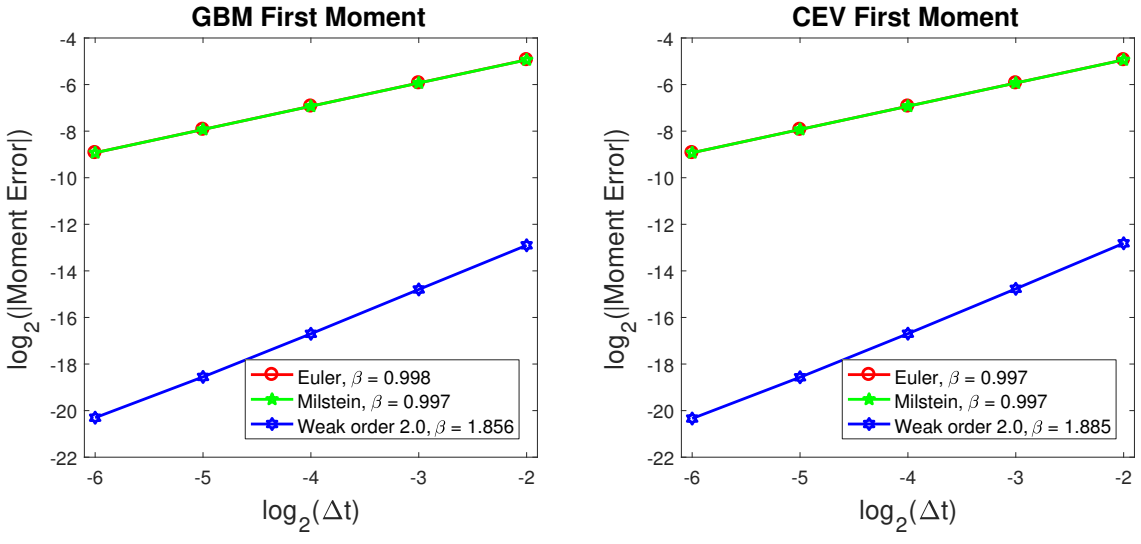

Figure 4.3: Convergence of the first moment for GBM and CEV.

the difference between the first moment of the resulting terminal quantizer and the true moment is plotted for a range of time step sizes. In each case, the horizontal axis provides the base-2 logarithm of the step size and the vertical axis the base-2 logarithm of the error in the first moment. Thus, the slope of each graph reflects the power of the step size and hence the order of weak convergence for the error. In the figure legends, the regressed gradients of the graphs, denoted by β, indicate the approximate weak-orders of convergence. As is theoretically expected, see Kloeden and Platen [1999], both the Euler and Milstein schemes have approximately weak-order one convergence, whereas the weak-order 2.0 Taylor scheme reaches a weak-order close to two. Therefore, the latter scheme is an order of magnitude more accurate and can be expected to produce results with substantially lower error than the Euler scheme when valuing contingent claims. For increased accuracy in these plots, the cardinality used was $N_k = 1\,000$.

4.3 Option Pricing

In this section, contingent claims are priced using the RMQ algorithm and the three update schemes are compared for accuracy and efficiency. The claims priced include European, Bermudan and discretely-monitored barrier options under the dynamics of both GBM and its generalization, the CEV model.

Both the GBM and CEV models, and the selected model-specific parameter sets, are specified in Section 3.5. As is implied by these specifications, the continuously compounded interest rate is assumed constant with a value $r = 5\%$. All option maturities are one year and the RMQ algorithm is executed using $n = 12$, i.e., using monthly steps, with constant cardinality of $N_k = 250$ for all k.

All simulations were executed using MATLAB 2014a on a computer with a 2.67 GHz Intel Core i7-620M processor and 8 GB of RAM. Although specific times are provided in each subsection, on this system all the RMQ pricing examples execute in less than a second.

4.3.1 European Option Pricing

Once a terminal quantizer has been obtained using the RMQ algorithm, a European option with payoff function $H(X, K)$ at maturity $T = t_n$, where X represents the asset process and K the

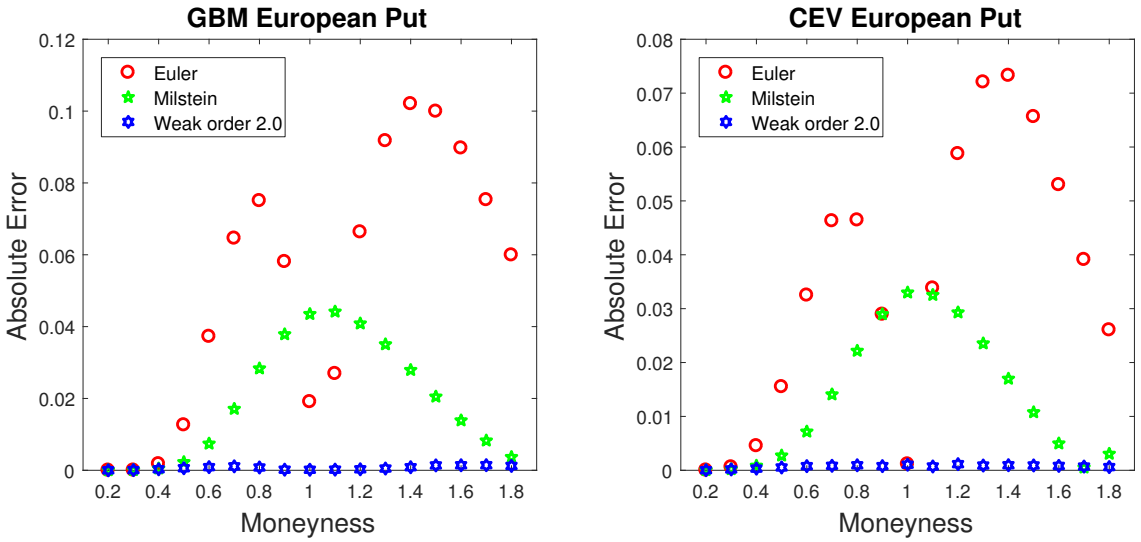

Figure 4.4: Accuracy of GBM and CEV European put prices computed using RMQ, as compared to analytical solutions.

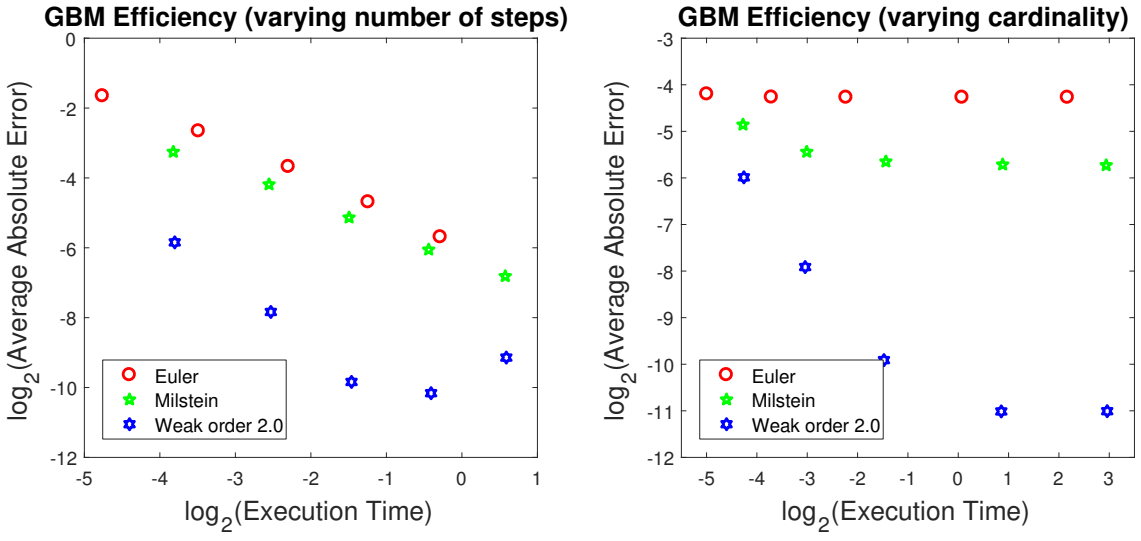

Figure 4.5: Numerical efficiency of the update schemes for European put options.

strike, may be priced directly by using the expectation defined in (2.1). The price is given by

$$H_0 = e^{-rT}\mathbb{E}[H(X_T, K)] \approx e^{-rT}\mathbf{p}_n H(\mathbf{\Gamma}_n, K), \tag{4.3}$$

where H_0 is the value of the claim at initial time $t_0 = 0$, $\mathbf{p}_n$ is the row vector of terminal probabilities defined in (3.11) and $H(\mathbf{\Gamma}_n, K)$ is the function H applied element-wise to $\mathbf{\Gamma}_n$, which, as specified previously, is a column vector of length N_n.

Figure 4.4 shows the accuracy of put option prices for the GBM and CEV models for a wide range of strikes. The GBM option prices are compared against the Black-Scholes option pricing formula, whereas the CEV prices are compared against the analytical solution originally due to Schroder [1989] and reformulated in terms of the noncentral chi-squared distribution by Hsu et al. [2008]. Both formulae are provided in Appendix D for completeness. As in Section 3.5, the x-axis in the figures represents fixed-spot inverse moneyness, which is determined as the variable strike value over the initial asset price, X_0. The oscillation of the error as a function of strike is due to the tree-nature of the resulting quantization grid, as previously depicted in Figure 3.2. This oscillatory behaviour has been investigated for binomial trees in Diener and Diener [2004].

Even though the Euler scheme is reasonably accurate to start with, the increased accuracy of the Milstein and the simplified weak-order 2.0 schemes is evident. For certain strikes the error is reduced by an order of magnitude.

Averaged over 20 runs, the Euler RMQ algorithm took approximately 0.32 seconds, the Milstein RMQ took approximately 0.56 seconds and the simplified weak-order 2.0 RMQ algorithm took approximately 0.58 seconds to price all strikes.

The question of overall efficiency is complex, but general guidelines can be provided. As can be seen in Figure 4.4, the Euler scheme approximates at-the-money options reasonably well, while performing poorly for out-the-money options. Thus, accuracy is dependent not only on the process but also the option and associated parameters (e.g. strikes, barriers, etc.) being considered. Therefore, efficiency should be considered as a trade-off between accuracy across all strikes and total execution time.

Figure 4.5 shows log-log plots of average absolute error (over all strikes) as a function of execution time for the GBM European put options with parameters and strikes as before. A similar graph can be generated for the case of the CEV process. In the left panel the number of time steps was varied ($n \in [2, 4, 8, 16, 32]$), while in the right panel cardinality was varied ($N_k \in [50, 100, 200, 400, 800]$ for all $0 < k \leq n$), with all other parameters remaining the same. The graphs clearly indicate that for any given execution time, the weak-order 2.0 Taylor scheme provides a lower aggregate error, followed by the Milstein scheme and finally the Euler scheme. It is also interesting to note that, without a corresponding increase in cardinality, there is a point where adding more time steps may lead to less accurate values, as evidenced in the weak-order 2.0 results in the left graph of Figure 4.5. If, for instance, the cardinality in these simulations were increased above 250, then the weak-order 2.0 results would continue to have a decreasing absolute error as a function of execution time.

To illustrate the necessity of accurately modelling the zero boundary, consider the GBM model with $X_0 = 0.5$ and $\sigma = 90\%$, and the CEV model with $X_0 = 0.5$, $\alpha = 0.35$ and $\sigma_{\text{LN}} = 50\%$, with the rest of the parameters as before. Figure 4.6 shows the error when pricing European put options under these extreme regimes. Here, to maintain consistency in the models, reflection is used for the GBM model, to prevent the value of the stock reaching zero, and absorption is used for the CEV model, to prevent arbitrage. Again, the weak-order 2.0 Taylor scheme performs the best on average across strikes.

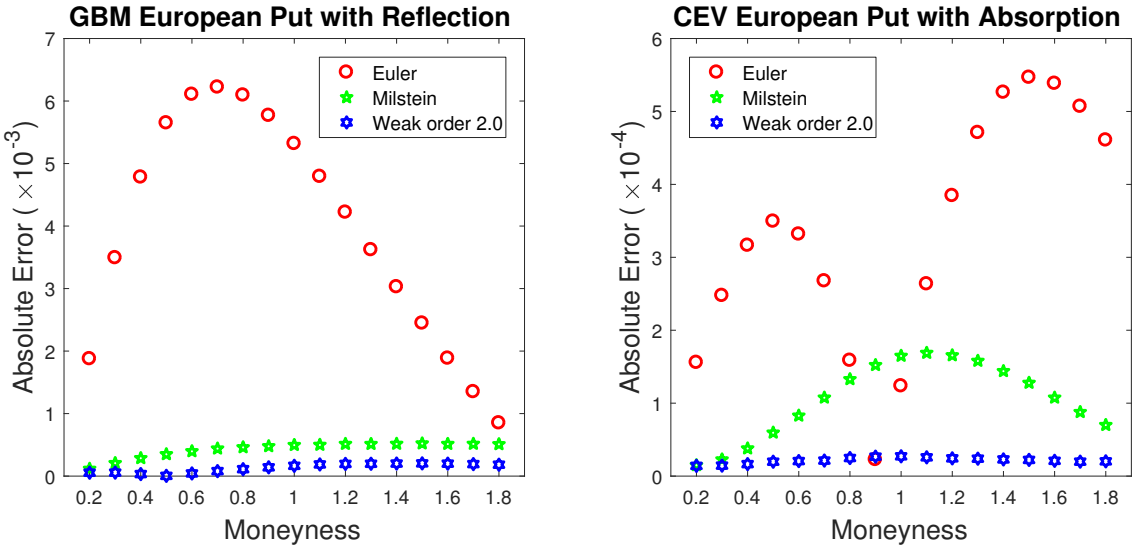

Figure 4.6: Accuracy of GBM and CEV European put prices with extreme parameters necessitating reflection and absorption, as compared to analytical solutions.

It is important to note that under this choice of parameters for the GBM and CEV models, the standard RMQ formulation of Pagès and Sagna [2015] fails. Without implementing the modifications for either absorption or reflection proposed in Section 3.4, some codewords become negative at a certain point in the execution of the RMQ algorithm, leading to discrete-time updates with imaginary values.

4.3.2 Bermudan Option Pricing

Bermudan option prices are computed using the standard Backward Dynamic Programming Principle (BDPP), an important result from discrete-time optimal stopping theory. Pagès [2014] reviews the use of the BDPP as applied to grids that result from quantization.

Once quantization grids and corresponding transition probability matrices have been computed using the RMQ algorithm, the high-level algorithm for Bermudan option pricing may be specified as follows:

1. Initialize $\mathbf{h}_n = H(\mathbf{\Gamma}_n, K)$
2. For $k = n-1, \ldots, 1$
   Set $\mathbf{h}_k = \max(H(\mathbf{\Gamma}_k, X), e^{-r\Delta t}\mathbf{P}_{k+1}\mathbf{h}_{k+1})$
3. Set $H_0 = e^{-r\Delta t}\mathbf{p}_1\mathbf{h}_1$

Here the max function is applied element-wise with its second argument being the continuation value, which is easily computed as a conditional expectation due to availability of the transition probability matrix at each time step. The initial value of the Bermudan claim is given by H_0.

In Figure 4.7 the accuracy of a Bermudan put option with monthly exercise opportunities is shown for the GBM and CEV models. The reference price is computed using a high resolution Crank-Nicholson finite difference scheme using 600 time steps and 800 stock increments, equally spaced between zero and $4 \times X_0$.

All three RMQ algorithms result in low absolute differences, with the weak-order 2.0 Taylor scheme again producing errors that are an order of magnitude smaller.

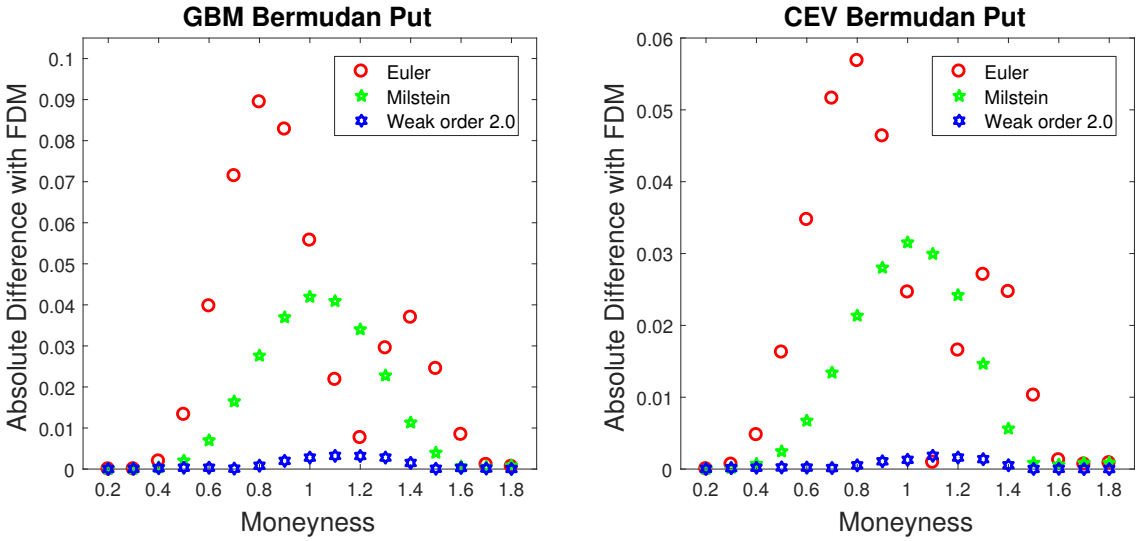

Figure 4.7: Accuracy of GBM and CEV Bermudan put option prices, as compared to a high resolution Crank-Nicholson finite difference scheme.

Again, averaged over 20 runs, the Euler RMQ algorithm took approximately 0.37 seconds, the Milstein RMQ took approximately 0.6 seconds and the weak-order 2.0 Taylor RMQ algorithm took approximately 0.64 seconds to price all strikes.

4.3.3 Barrier Option Pricing

The pricing of barrier options has previously been explored in the context of quantization by Sagna [2011]. This work showed that the barrier-crossing approach described in Glasserman [2003, Sec. 6.4] may be applied to marginal quantization using a so-called transition kernel formulation. This approach is now applied with the more accurate proposed schemes to price discretely monitored barrier options.

Consider expression (4.3) for pricing European options, which may be re-written as

$$H_0 \approx e^{-rT} \left(\mathbf{p}_1 \prod_{k=1}^{n-1} \mathbf{P}_{k+1} \right) H(\mathbf{\Gamma}_n, K).$$

To price a knock-out barrier option the transition probability matrix at each time step in this expression must be modified to take into account the possibility that the underlying process breaches the barrier. Thus, the transition probabilities are rescaled by multiplying them by the probability of not having crossed the barrier.

Let $g(x, y)$ be the probability of transitioning between states x and y without crossing the barrier. If an $N_k \times N_{k+1}$ matrix of values is formed by

$$[\mathbf{G}_{k+1}]_{i,j} = g(\gamma_k^i, \gamma_{k+1}^j),$$

then $\mathbf{P}_{k+1} \circ \mathbf{G}_{k+1}$ defines the transition kernel. Again, $\circ$ denotes the element-wise Hadamard product. The barrier option may then be priced using

$$H_0 \approx e^{-rT} \left((\mathbf{p}_1 \circ \mathbf{g}_1) \prod_{k=1}^{n-1} (\mathbf{P}_{k+1} \circ \mathbf{G}_{k+1}) \right) H(\mathbf{\Gamma}_n, K),$$

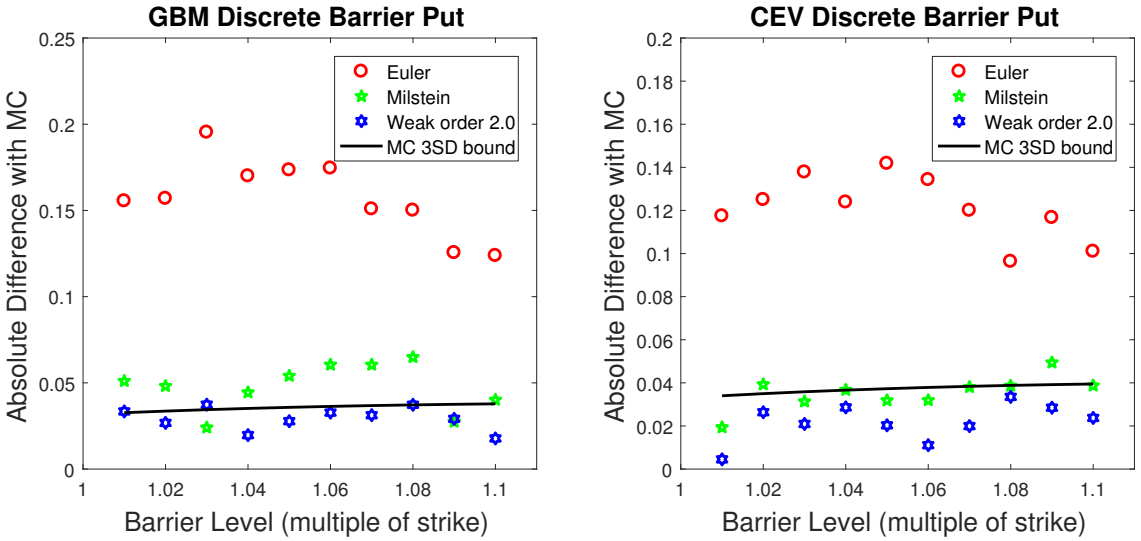

Figure 4.8: Accuracy of GBM and CEV discretely monitored up-and-out put option prices, as compared to a Monte Carlo simulation.

where $\mathbf{g}_1 = [g(X_0, \gamma_1^1), \ldots, g(X_0, \gamma_1^{N_1})]$ is a row vector.

In the case of discretely-monitored up-and-out barrier options with barrier level L, the function g is given simply as the indicator function

$$g(x, y) = \mathbb{I}_{\{\max(x,y)<L\}}.$$

Refer to Glasserman [2003] and Sagna [2011] for the continuous monitoring case using the Rayleigh distribution.

In Figure 4.8 the accuracy of discretely-monitored up-and-out put option prices generated using RMQ is compared to a Monte Carlo implementation under the GBM and CEV models. The barrier levels (x-axis) are expressed as multiples of the at-the-money strike. Since $n = 12$, the barrier is monitored monthly.

The reference prices are provided by a one million path Monte Carlo experiment. The Monte Carlo paths are generated using Euler-Maruyama updates with 1 200 time steps, while ensuring that the barriers are only monitored at monthly intervals. The exact transition density was used to generate Monte Carlo samples for GBM to confirm that results were consistent and generating the correct standard deviations.

The results show a similar pattern to those in the previous sections, with an important caveat: the weak-order 2.0 Taylor scheme produces prices that, for the majority of the barrier values considered, lie within the three standard deviation bound of the million-path Monte Carlo experiment. The other two RMQ schemes are producing results that are statistically significantly incorrect when compared to the Monte Carlo simulation; motivating the use of the weak-order 2.0 scheme for options with path-dependence.

Averaged over 20 runs, the Euler RMQ algorithm took approximately 0.34 seconds, the Milstein RMQ took approximately 0.57 seconds and the weak-order 2.0 Taylor RMQ algorithm took approximately 0.61 seconds to price all barrier options.

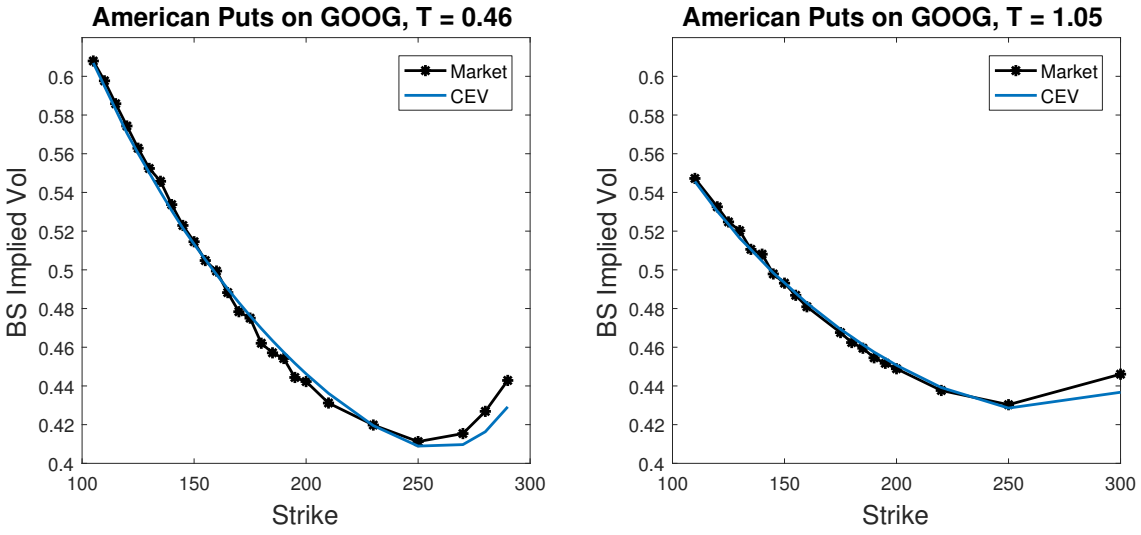

Figure 4.9: Calibration of the CEV model to American put options on GOOG on 03/01/2005.

4.4 Calibration

An advantage of the RMQ algorithm, like traditional tree methods, is the ability to price multiple options without needing to re-generate the underlying grid. Once the optimal quantization grid has been generated out to the furthest required option maturity, the computational cost of pricing options is negligible. An immediate application is the ability to calibrate single-factor models directly to non-vanilla products. RMQ has already been used to calibrate the quadratic normal volatility model to vanilla options on the DAX index by Callegaro et al. [2015].

For the CEV model, the absorbing boundary modification of the RMQ algorithm is necessary for effective calibration. Firstly, to allow α values of less than zero, for which an absorbing boundary is the only applicable boundary condition, and secondly, to stabilize the algorithm as the calibration procedure searches the parameter space.

The calibration problem can be formulated as

$$\min_{\Theta \in \mathbb{R}^2} F(\Theta),$$

where F is the objective or error function and $\Theta = \{\alpha,\ \sigma_{\text{CEV}}\}$ is the parameter set for the CEV model. In Gschnaidtner and Escobar [2015] the *relative squared volatility error* is recommended as the objective function for calibrating the Heston model, and it is adopted here. It is defined as

$$F(\Theta) = \sum_{l=1}^{L} \left(\frac{\sigma_l^{\text{Model}}(\Theta) - \sigma_l^{\text{Market}}}{\sigma_l^{\text{Market}}} \right)^2,$$

where L is the number of calibration instruments used, $\sigma_l^{\text{Model}}(\Theta)$ is the Black-Scholes implied volatility that corresponds to pricing calibration instrument l with the model parameters Θ and σ_l^{Market} is the implied volatility for that instrument in the market.

As a proof-of-concept example, the CEV model is calibrated to American put options of two different maturities on the GOOG stock on 03/01/2005. The results are displayed in Figure 4.9 with the data and calibrated parameters summarized in Table 4.1. Note that the table displays the calibrated log-normal volatility, given by $\sigma_{\text{LN}} = \sigma_{\text{CEV}} X_0^{\alpha-1}$. The stock price on this day was $X_0 = 202.71$. The U.S. Department of Treasury rate is used as a proxy for the risk-free rate and

	Maturity	T	Options	r	α	σ_{LN}	$F(\Theta)$
T_1	18/06/2005	0.458	26	0.0274	-0.035	0.439	0.0032
T_2	21/01/2006	1.05	19	0.0311	0.327	0.441	0.0008

Table 4.1: Summary of calibration data and results for the CEV model calibrated to American put options on GOOG on 03/01/2005.

has been linearly interpolated to the necessary maturities. Table 4.1 also indicates the number of calibration instruments for each maturity. Only options with strikes within 50% of at-the-money were considered.

The calibration results are encouraging; the log-normal volatility appears stable across the maturities. However, α is dramatically different, indicating that the CEV model would fail to calibrate effectively to the implied volatility surface on this day.

A problem that arises during calibration is the occasional inability to invert the Hessian of the distortion function, required to execute the Newton iteration (3.5). As the parameter space is searched, this matrix can become ill-conditioned. Further research into calibration using RMQ could consider using Lloyd's algorithm with the Anderson acceleration technique, as is done in Bormetti et al. [2017]. The results could potentially be compared to the de-Americanization technique investigated by Burkovska et al. [2016].

Chapter 5

Fast Quantization of Stochastic Volatility Models

5.1 Overview

Consider a special case of the multi-dimensional diffusion described by (3.1): a two-dimensional continuous-time diffusion process that specifies a *stochastic volatility* model,

$$dX_t = a^x(X_t)\,dt + b^x(X_t)\,dW_t^x, \qquad X_0 = x_0 \in \mathbb{R}, \tag{5.1}$$

$$dY_t = a^y(Y_t)\,dt + b^y(X_t, Y_t)\,dW_t^y, \qquad Y_0 = y_0 \in \mathbb{R}, \tag{5.2}$$

defined on the filtered probability space $(\Omega, \mathscr{F}, (\mathscr{F}_t)_{t\in[0,T]}, \mathbb{P})$, where $d\langle W^x, W^y\rangle_t = \rho\,dt$ and with sufficiently smooth and bounded drift and diffusion coefficient functions to ensure the existence of a strong solution. Here Y typically describes an asset price process or interest rate and the stochastic process X drives its volatility. The question of interest is:

> How does one optimally approximate $Y_{t_k} : \Omega \to \mathbb{R}$, for some time discretization point $t_k \in (0, T]$, by $\widehat{Y}_k : \Omega \to \Gamma_k^y$, where $\Gamma_k^y = \{y_k^1, \ldots, y_k^{N^y}\}$, when the distribution of Y_{t_k} is unknown?

Note that the quantization grid associated with the X-process at time-step t_k is denoted $\Gamma_k^x = \{x_k^1, \ldots, x_k^{N^x}\}$. The Euler scheme for the above system is given by

$$\begin{aligned} \overline{X}_{k+1} &= \overline{X}_k + a^x(\overline{X}_k)\Delta t + b^x(\overline{X}_k)\sqrt{\Delta t} z_{k+1}^x, && \overline{X}_0 = x_0, \qquad (5.3)\\ &=: \mathcal{U}^x(\overline{X}_k, z_{k+1}^x), \\ \overline{Y}_{k+1} &= \overline{Y}_k + a^y(\overline{Y}_k) + b^y(\overline{X}_k, \overline{Y}_k)\sqrt{\Delta t} z_{k+1}^y, && \overline{Y}_0 = y_0, \qquad (5.4)\\ &=: \mathcal{U}^y(\overline{X}_k, \overline{Y}_k, z_{k+1}^y), \end{aligned}$$

for $0 \le k < n$, $\Delta t = T/n$, and where $z_{k+1}^x,\ z_{k+1}^y \sim \mathcal{N}(0,1)$ are standard Gaussian random variables with correlation $\rho \in [-1, 1]$. Given $\overline{X}_k$ and $\overline{Y}_k$, $\overline{X}_{k+1}$ and $\overline{Y}_{k+1}$ jointly possess the bivariate Gaussian distribution. Thus, the RMQ algorithm from Chapter 3 can be applied, where the joint implied marginal distribution of $\overline{X}_{k+1}$ and $\overline{Y}_{k+1}$ becomes a weighted sum of bivariate

Gaussian distributions,

$$
\begin{aligned}
&F_{\overline{X}_{k+1},\overline{Y}_{k+1}}(x,y) \\
&\quad = \int_{\mathbb{R}^2} \mathbb{P}(\mathcal{U}^x(r, z^x_{k+1}) \le x, \mathcal{U}^y(r, s, z^y_{k+1}) \le y)\, d\mathbb{P}(\overline{X}_k \le r, \overline{Y}_k \le s) \\
&\quad \approx \sum_{i=1}^{N^x} \sum_{u=1}^{N^y} \mathbb{P}(\mathcal{U}^x(x^i_k, z^x_{k+1}) \le x, \mathcal{U}^y(x^i_k, y^u_k, z^y_{k+1}) \le y) \mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k). \qquad (5.5)
\end{aligned}
$$

The approximation in (5.5) is formed by replacing the continuous, and unknown, joint distribution of $\overline{X}_k$ and $\overline{Y}_k$ with the discrete, and known quantized distribution of $\widehat{X}_k$ and $\widehat{Y}_k$, as in the standard RMQ case. To apply the standard RMQ algorithm of Section 3.1 now requires the vector quantization of this two-dimensional distribution, which precludes the use of the efficient Newton-Raphson implementation and quickly becomes intractable.

Callegaro et al. [2016] propose a RMQ-based procedure for the case of stochastic volatility models that requires conditioning on the future quantization grid of the $\overline{X}$-process at each time step. In this work, an alternative is proposed: the Joint Recursive Marginal Quantization algorithm. The central idea is that instead of constructing a two-dimensional optimal quantizer for the $\overline{X}$- and $\overline{Y}$-processes as in the standard RMQ algorithm, two one-dimensional quantizers are constructed with the required two-dimensional grid generated from their Cartesian product. This is known as a *product quantizer*.

The main result of this section is to show that, when quantizing the $\overline{X}$- and $\overline{Y}$-updates recursively in this fashion, at each quantization step the correlation between the processes can be neglected. From the perspective of the distortion function, the processes can be viewed independently. After each quantization step, the correlation must still be accounted for when computing the joint probabilities and constructing the product grid.

Consider the Euler scheme in terms of the Cholesky decomposition,

$$
\begin{aligned}
\overline{X}_{k+1} &= \mathcal{U}^x(\overline{X}_k, z^x_{k+1}), && \overline{X}_0 = x_0, && (5.6)\\
\overline{Y}_{k+1} &= \overline{Y}_k + a^y(\overline{Y}_k) + b^y(\overline{X}_k, \overline{Y}_k)\sqrt{\Delta t}(\rho z^x_{k+1} + \sqrt{1-\rho^2} z^{\perp}_{k+1}), && \overline{Y}_0 = y_0, && (5.7)\\
&=: \mathcal{U}^y_{\mathrm{C}}(\overline{X}_k, \overline{Y}_k, z^x_{k+1}, z^{\perp}_{k+1}),
\end{aligned}
$$

for $0 \le k < n$, where $\mathcal{U}^y_{\mathrm{C}}$ is the new update function that acts on the independent standard Gaussian random variables, z^x_{k+1} and $z^{\perp}_{k+1}$.

Now $\overline{X}$, referred to as the *independent* process, drives the specification of the stochastic volatility factor in the *dependent* process $\overline{Y}$. It should be clear that the quantization of the Euler scheme for the independent process, $\overline{X}$, proceeds directly using the standard one-dimensional RMQ algorithm and can be performed for all time steps without reference to $\overline{Y}$. This results in a series of marginal optimal quantization grids for $\overline{X}_k$, denoted Γ^x_k, for $0 \le k \le n$, along with their associated probabilities and transition probability matrices.

Proposition 5.1.1 below shows that quantizing the Euler update $\overline{Y}_{k+1} = \mathcal{U}^y_{\mathrm{C}}(\overline{X}_k, \overline{Y}_k, z^x_{k+1}, z^{\perp}_{k+1})$ is equivalent to quantizing the original update, $\mathcal{U}^y(\overline{X}_k, \overline{Y}_k, z)$ where $z \sim \mathcal{N}(0,1)$ is a standard Gaussian random variable.

Proposition 5.1.1 (Margined Distortion)**.** *Given the Euler scheme defined by (5.6) and (5.7), and the quantizers Γ^x_k and Γ^y_k, $k \ge 0$, the distortion of the quantizer Γ^y_{k+1} may be expressed as*

$$
\widetilde{D}(\Gamma^y_{k+1}) = \mathbb{E}\big[\big|\mathcal{U}^y(\overline{X}_k, \overline{Y}_k, z) - \pi_{\Gamma^y_{k+1}}(\mathcal{U}^y(\overline{X}_k, \overline{Y}_k, z))\big|^2\big],
$$

where the update function is defined by

$$\mathcal{U}^y(\overline{X}_k, \overline{Y}_k, z) = \overline{Y}_k + a^y(\overline{Y}_k)\Delta t + b^y(\overline{X}_k, \overline{Y}_k)\sqrt{\Delta t}z,$$

with $z \sim \mathcal{N}(0,1)$.

Proof. The distortion of the quantizer Γ^y_{k+1} for $\overline{Y}_{k+1}$ is given in terms of the update as

$$\begin{aligned}
\widetilde{D}(\Gamma^y_{k+1}) :=& \mathbb{E}\Big[\big|\overline{Y}_{k+1} - \pi_{\Gamma^y_{k+1}}(\overline{Y}_{k+1})\big|^2\Big] \\
=& \mathbb{E}\Big[\mathbb{E}\Big[\big|\overline{Y}_{k+1} - \pi_{\Gamma^y_{k+1}}(\overline{Y}_{k+1})\big|^2 \,\Big|\, \overline{X}_k, \overline{Y}_k\Big]\Big] \\
=& \int_{\mathbb{R}^2} \mathbb{E}\Big[\big|\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1}) - \pi_{\Gamma^y_{k+1}}(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1}))\big|^2\Big]\, d\mathbb{P}(\overline{X}_k \le x, \overline{Y}_k \le y) \\
=& \int_{\mathbb{R}^2} \mathbb{E}[f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1})\big)]\, d\mathbb{P}(\overline{X}_k \le x, \overline{Y}_k \le y),
\end{aligned}$$

where $f(w) := \Big(w - \pi_{\Gamma^y_{k+1}}(w)\Big)^2$. The inner expectation may be written explicitly as

$$\begin{aligned}
&\mathbb{E}[f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1})\big)] \\
&\quad = \frac{1}{2\pi}\int_{\mathbb{R}^2} f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,u,v)\big) \exp\left(\frac{-u^2}{2}\right)\exp\left(\frac{-v^2}{2}\right) dv\, du.
\end{aligned}$$

Now, let

$$z = \rho u + \sqrt{1-\rho^2}v,$$

which means that

$$v = \frac{z - \rho u}{\sqrt{1-\rho^2}} \qquad \text{and} \qquad dv = \frac{1}{\sqrt{1-\rho^2}}\, dz.$$

Then,

$$\begin{aligned}
&\mathbb{E}[f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1})\big)] \\
&\quad = \frac{1}{2\pi\sqrt{1-\rho^2}}\int_{\mathbb{R}^2} f\big(\mathcal{U}^y(x,y,z)\big) \exp\left(\frac{-u^2}{2}\right)\exp\left(\frac{-(z-\rho u)^2}{2(1-\rho^2)}\right) dz\, du \\
&\quad = \frac{1}{2\pi\sqrt{1-\rho^2}}\int_{\mathbb{R}^2} f\big(\mathcal{U}^y(x,y,z)\big) \exp\left(\frac{-z^2}{2}\right)\exp\left(\frac{-(u-\rho z)^2}{2(1-\rho^2)}\right) dz\, du \\
&\quad = \frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}} f\big(\mathcal{U}^y(x,y,z)\big) \exp\left(\frac{-z^2}{2}\right) \underbrace{\frac{1}{\sqrt{2\pi(1-\rho^2)}}\int_{\mathbb{R}} \exp\left(\frac{-(u-\rho z)^2}{2(1-\rho^2)}\right) du}_{=1}\, dz \\
&\quad = \frac{1}{\sqrt{2\pi}}\int_{\mathbb{R}} f\big(\mathcal{U}^y(x,y,z)\big) \exp\left(\frac{-z^2}{2}\right) dz,
\end{aligned}$$

where Fubini's theorem has been used in the penultimate step. Thus,

$$\mathbb{E}[f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1}, z^\perp_{k+1})\big)] = \mathbb{E}[f\big(\mathcal{U}^y(x,y,z)\big)].$$

Combining these, yields

$$\begin{aligned}\widetilde{D}(\Gamma^y_{k+1}) &= \int_{\mathbb{R}^2} \mathbb{E}[f(\mathcal{U}^y(x,y,z))]\, d\mathbb{P}(\overline{X}_k \le x, \overline{Y}_k \le y) \\ &= \int_{\mathbb{R}^2} \mathbb{E}\Big[\big|\mathcal{U}^y(x,y,z) - \pi_{\Gamma^y_{k+1}}(\mathcal{U}^y(x,y,z))\big|^2\Big]\, d\mathbb{P}(\overline{X}_k \le x, \overline{Y}_k \le y) \\ &= \mathbb{E}\Big[\big|\mathcal{U}^y(\overline{X}_k,\overline{Y}_k,z) - \pi_{\Gamma^y_{k+1}}(\mathcal{U}^y(\overline{X}_k,\overline{Y}_k,z))\big|^2\Big],\end{aligned}$$

as required. □

Remark 5.1.2. *The above proposition shows that the quantization of $\overline{Y}_{k+1}$ depends only on its distribution, and, from the perspective of the distortion function, the correlation between $\overline{Y}_{k+1}$ and $\overline{X}_{k+1}$ is irrelevant. Another way of saying this is that*

$$f\big(\mathcal{U}^y_{\mathrm{C}}(x,y,z^x_{k+1},z^{\perp}_{k+1})\big) \overset{d}{=} f\big(\mathcal{U}^y(x,y,z)\big),$$

for any $z \sim \mathcal{N}(0,1)$, and, since only weighted sums of expectations of these values need to be considered when computing the distortion, the correlation between z^x_{k+1} and $z^{\perp}_{k+1}$ need not be considered. As seen later, it is still necessary to take correlation into account when computing the joint probabilities of $\widehat{Y}_{k+1}$ and $\widehat{X}_{k+1}$.

As in Chapter 3, this result gives rise to the following recursive procedure:

$$\begin{aligned}&\widetilde{Y}_0 := \overline{Y}_0, \\ &\widehat{Y}_k := \pi_{\Gamma^y_k}(\widetilde{Y}_k) \qquad \text{and} \qquad \widetilde{Y}_{k+1} = \mathcal{U}^y(\widehat{X}_k, \widehat{Y}_k, z_{k+1})\end{aligned}$$

for $z_{k+1} \sim \mathcal{N}(0,1)$ and $k = 0, \dots, n-1$.

What remains is to specify the distortion that must be minimized explicitly. Suppose, at time step k, the quantizer for the dependent process Γ^y_k has been computed along with the corresponding joint probabilities, $\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k)$, for $1 \le i \le N^x$ and $1 \le u \le N^y$, then the distortion for the quantizer of $\overline{Y}_{k+1}$ may be approximated by

$$\begin{aligned}\widetilde{D}(\Gamma^y_{k+1}) &= \int_{\mathbb{R}^2} \mathbb{E}\Big[\big|\mathcal{U}^y(x,y,z^y_{k+1}) - \pi_{\Gamma^y_{k+1}}(\mathcal{U}^y(x,y,z^y_{k+1})\big|^2\Big]\, d\mathbb{P}(\overline{X}_k \le x, \overline{Y}_k \le y) \\ &\approx \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \mathbb{E}\Big[\big|\mathcal{U}^y(x^i_k, y^u_k, z) - \widehat{Y}_{k+1}\big|^2\Big]\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k) \qquad (5.8) \\ &=: D(\Gamma^y_{k+1}).\end{aligned}$$

It is again assumed that the cardinality of Γ^y_k is fixed at N^y for all $0 < k \le n$ and that $\Gamma^y_0 = \{y_0\}$. As before, the quantizer Γ^y_1 may be computed using standard vector quantization of the normal distribution implied by the Euler update (5.4).

5.2 Numerical Methods

For the remainder of this section it is assumed that, conditional on knowing the quantizers Γ^x_k and Γ^y_k, their associated joint probabilities are known — in Section 5.2.1 two different approaches for computing these probabilities are provided. Under this assumption, and having rewritten the distortion (5.8) in terms of the margined update function, the minimization problem that generates

the quantizer at time-step $k+1$ may be specified using the Newton-Raphson iteration

$$ {}^{(l+1)}\mathbf{\Gamma}^y_{k+1} = {}^{(l)}\mathbf{\Gamma}^y_{k+1} - \left[\nabla^2 D\left({}^{(l)}\mathbf{\Gamma}^y_{k+1}\right)\right]^{-1} \nabla D\left({}^{(l)}\mathbf{\Gamma}^y_{k+1}\right), \tag{5.9} $$

where $\mathbf{\Gamma}^y_{k+1}$ is a column vector of the codewords in Γ^y_{k+1} and $0 \leq l < l_{\max}$ is the iteration index. Closely following Section 3.3, closed-form expressions for the gradient of the distortion, $\nabla D\left(\mathbf{\Gamma}^y_{k+1}\right)$, and the tridiagonal Hessian matrix, $\nabla^2 D\left(\mathbf{\Gamma}^y_{k+1}\right)$, may now be derived.

To summarise notation, the update of the dependent process is written as

$$ \mathcal{U}^y(x^i_k, y^u_k, z) =: U^{i,u}_{k+1} = \overline{m}^{i,u}_k Z_{k+1} + \overline{c}^u_k, $$

where

$$ \overline{m}^{i,u}_k := b^y(x^i_k, y^u_k)\sqrt{\Delta t} \qquad \text{and} \qquad \overline{c}^u_k := y^u_k + a^y(y^u_k)\Delta t $$

with $Z_{k+1} \sim \mathcal{N}(0,1)$. Note that the i and j indices, for $1 \leq i, j \leq N^x$, always refer to the codewords of the quantizers for the $\overline{X}$-process, whereas the u and v indices, for $1 \leq u, v \leq N^y$, always refer to the codewords of the quantizers for the $\overline{Y}$-process.

The gradient of the distortion is given by

$$ \begin{aligned} \frac{\partial D(\Gamma^y_{k+1})}{\partial y^v_{k+1}} &= 2\sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \mathbb{E}\Big[\mathbb{I}_{\{U^{i,u}_{k+1} \in R^v_{k+1}\}}(y^v_{k+1} - U^{i,u}_{k+1})\Big]\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k) \\ &= 2\sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \int_{U^{i,u}_{k+1} \in R^v_{k+1}} (y^v_{k+1} - U^{i,u}_{k+1})\, d\mathbb{P}(Z_{k+1} < z)\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k), \end{aligned} \tag{5.10} $$

where R^v_{k+1} is the region associated with codeword y^v_{k+1}. To rewrite the integration bounds in terms of the Gaussian random variable, consider that $U^{i,u}_{k+1} \in R^v_{k+1}$ implies that $U^{i,u}_{k+1}$ lies between the region boundaries of the codeword y^v_{k+1}. This means

$$ r^{v-}_{k+1} < U^{i,u}_{k+1} \leq r^{v+}_{k+1} \qquad \text{and} \qquad r^{v\pm}_{k+1} := \tfrac{1}{2}(y^{v\pm1}_{k+1} + y^v_{k+1}), $$

and $r^{1-}_{k+1} = -\infty$ and $r^{N^y+}_{k+1} = \infty$ by definition. Thus,

$$ U^{i,u}_{k+1} \in R^v_{k+1} = \begin{cases} r^{i,u,v-}_{k+1} < Z_{k+1} \leq r^{i,u,v+}_{k+1} & \text{for } \overline{m}^{i,u}_k \geq 0 \\ r^{i,u,v-}_{k+1} > Z_{k+1} \geq r^{i,u,v+}_{k+1} & \text{for } \overline{m}^{i,u}_k < 0, \end{cases} $$

where

$$ r^{i,u,v\pm}_{k+1} := \frac{r^{v\pm}_{k+1} - \overline{c}^u_k}{\overline{m}^{i,u}_k}, \tag{5.11} $$

is defined to be the standardized region boundary. Similar to the region boundaries of the independent process, see (3.10), it refers to the region boundaries of the codeword y^v_{k+1}, when viewed from the codewords x^i_k and y^u_k of the previous time step.

Let $f_{Z_{k+1}}$ and $F_{Z_{k+1}}$ denote the PDF and CDF of a standard normal random variable Z_{k+1}, respectively, and define $M_{Z_{k+1}}$ as the first lower partial expectation of Z_{k+1},

$$ M_{Z_{k+1}}(z) := \mathbb{E}[Z_{k+1}\mathbb{I}_{\{Z_{k+1}<z\}}]. $$

Then, by direct evaluation of the integral in (5.10), each element of the gradient of the distortion

at time-step $k+1$ is given by

$$\frac{\partial D(\Gamma_{k+1}^y)}{\partial y_{k+1}^v} = 2\sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\Big[(y_{k+1}^v - \overline{c}_k^u)\,\mathrm{sgn}(\overline{m}_k^{i,u})(F_{Z_{k+1}}(r_{k+1}^{i,u,v+}) - F_{Z_{k+1}}(r_{k+1}^{i,u,v-})) \\ -|\overline{m}_k^{i,u}|(M_{Z_{k+1}}(r_{k+1}^{i,u,v+}) - M_{Z_{k+1}}(r_{k+1}^{i,u,v-}))\Big]\,\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u). \quad (5.12)$$

The N^y-elements of the main diagonal of the tridiagonal Hessian matrix, $\nabla^2 D\left(\mathbf{\Gamma}_{k+1}^y\right)$, are given by

$$\frac{\partial^2 D(\Gamma_{k+1}^y)}{\partial (y_{k+1}^v)^2} = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\Bigg[2\,\mathrm{sgn}(\overline{m}_k^{i,u})(F_{Z_{k+1}}(r_{k+1}^{i,u,v+}) - F_{Z_{k+1}}(r_{k+1}^{i,u,v-})) \\ + \frac{1}{2|\overline{m}_k^{i,u}|} f_{Z_{k+1}}(r_{k+1}^{i,u,v+})(y_{k+1}^v - y_{k+1}^{v+1}) \quad (5.13) \\ + \frac{1}{2|\overline{m}_k^{i,u}|} f_{Z_{k+1}}(r_{k+1}^{i,u,v-})(y_{k+1}^{v-1} - y_{k+1}^{v})\Bigg]\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u),$$

with the $(N^y - 1)$-elements of the super-diagonal and sub-diagonal given by

$$\frac{\partial^2 D(\Gamma_{k+1}^y)}{\partial y_{k+1}^v \partial y_{k+1}^{v+1}} = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\frac{1}{2|\overline{m}_k^{i,u}|} f_{Z_{k+1}}(r_{k+1}^{i,u,v+})(y_{k+1}^v - y_{k+1}^{v+1})\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u) \quad (5.14)$$

and

$$\frac{\partial^2 D(\Gamma_{k+1}^y)}{\partial y_{k+1}^v \partial y_{k+1}^{v-1}} = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\frac{1}{2|\overline{m}_k^{i,u}|} f_{Z_{k+1}}(r_{k+1}^{i,u,v-})(y_{k+1}^{v-1} - y_{k+1}^{v})\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u), \quad (5.15)$$

respectively.

The formulae above are similar to those derived for the standard RMQ case, with an additional summation over the codewords of the independent process. This is again a one-dimensional vector quantization problem, but this time the marginal distribution to be quantized consists of a sum of Euler updates that are weighted using joint probabilities. For this reason, this variant of the RMQ algorithm as referred to as the *Joint* RMQ algorithm (JRMQ). This allows it to be distinguished in the text from the standard RMQ algorithm described in Chapter 3.

When the above formulation is compared with the approach proposed by Callegaro et al. [2016], see Appendix D of their paper, it is observed that these equations have one fewer summation, since it is not necessary to condition on the independent process at time step $k+1$. This means that the expressions for the gradient and Hessian presented here are an order of magnitude more efficient to implement.

5.2.1 Computing the Joint Probabilities

Up to this point, it has been assumed that the joint probabilities required in (5.12) to (5.15) are available. In this section, it is shown how to compute these probabilities exactly, using the bivariate Gaussian cumulative distribution function, and how to construct a computationally efficient approximation. To facilitate efficient implementation, a matrix formulation of the system is provided in the next section.

From (5.6) and (5.7) it is evident that, conditional on the realizations of $\overline{X}_k$ and $\overline{Y}_k$, the joint probability distribution of $\overline{X}_{k+1}$ and $\overline{Y}_{k+1}$ is bivariate Gaussian. Consider the approximate joint

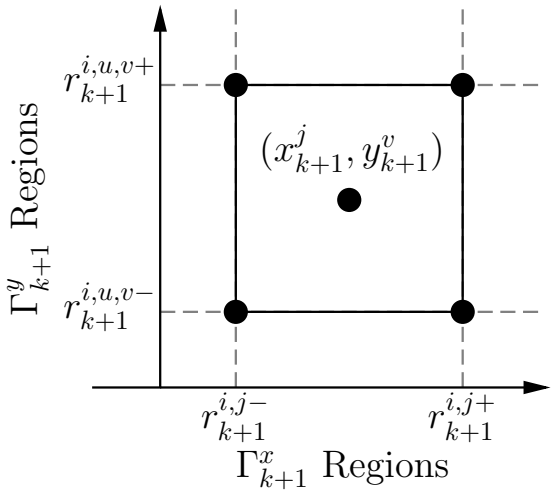

Figure 5.1: A standardized region for the bivariate Gaussian distribution with indices i and u fixed.

distribution of $\overline{X}_{k+1}$ and $\overline{Y}_{k+1}$ from (5.5)

$$\begin{aligned} F_{\overline{X}_{k+1},\overline{Y}_{k+1}}(x,y) &\approx \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \mathbb{P}(\mathcal{U}^x(x_k^i, z_{k+1}^x) \le x, \mathcal{U}^y(x_k^u, y_k^u, z_{k+1}^y) \le y)\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u), \\ &= \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \mathbb{P}\left(z_{k+1}^x \le \frac{x - c_k^i}{m_k^i}, z_{k+1}^y \le \frac{y - \overline{c}_k^u}{\overline{m}_k^{i,u}}\right) \mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u). \end{aligned}$$

The necessary joint probability is then given by

$$\begin{aligned} &\mathbb{P}(\widehat{X}_{k+1} = x_{k+1}^j, \widehat{Y}_{k+1} = y_{k+1}^v) = \\ &\qquad \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \left[\int_{r_k^{i,u,v-}}^{r_k^{i,u,v+}} \int_{r_k^{i,j-}}^{r_k^{i,j+}} \phi_2(x,y,\rho)\, dx\, dy\right] \mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u), \end{aligned} \tag{5.16}$$

where $\phi_2(x,y,\rho)$ is the bivariate Gaussian density function for two standard Gaussian random variables correlated by ρ. The double integral above refers to the probability mass of a rectangle delimited by the standardized regions of Γ_{k+1}^x and Γ_{k+1}^y, see Figure 5.1. Therefore, for each $1 \le j \le N^x$ and $1 \le v \le N^y$,

$$\begin{aligned} \mathbb{P}(\widehat{X}_{k+1} = x_{k+1}^j, \widehat{Y}_{k+1} = y_{k+1}^v) = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \Big[&\Phi_2\big(r_k^{i,j+}, r_k^{i,u,v+}, \rho\big) - \Phi_2\big(r_k^{i,j-}, r_k^{i,u,v+}, \rho\big) \\ &- \Phi_2\big(r_k^{i,j+}, r_k^{i,u,v-}, \rho\big) + \Phi_2\big(r_k^{i,j-}, r_k^{i,u,v-}, \rho\big)\Big] \\ &\times \mathbb{P}(\widehat{X}_k = x_k^i, \widehat{Y}_k = y_k^u), \end{aligned} \tag{5.17}$$

where $\Phi_2(x,y,\rho)$ is the standard bivariate Gaussian cumulative distribution function with correlation ρ evaluated at x and y.

Given the quantizers at time k, the joint probability in (5.17) is the exact probability of the two-dimensional elementary quantizer, (x_{k+1}^j, y_{k+1}^v), as implied by the Euler process, (5.3) and (5.4).

However, it requires the evaluation of the bivariate Gaussian distribution function. Although most programming languages have an efficient implementation of this function, it is significantly more expensive to compute than the univariate distribution. The joint probability can be approximated using only calls to the univariate Gaussian CDF by using quadrature to approximate the inner integral of (5.16).

While other approaches are possible, a simple quadrature rule is used by replacing $\overline{X}_{k+1}$ with its quantized version, $\widehat{X}_{k+1}$, which is constant over the interval. Then (5.16) becomes

$$\begin{aligned}
&\mathbb{P}(\overline{X}_{k+1} = x^j_{k+1}, \overline{Y}_{k+1} = y^v_{k+1}) \\
&\quad \approx \sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\left[\int_{r_k^{i,u,v-}}^{r_k^{i,u,v+}} \phi_2\big(y,\rho\big|\tfrac{x^j_{k+1}-c^i_k}{m^i_k}\big)dy\right]\mathbb{P}(\widehat{X}_{k+1}=x^j_{k+1})\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k) \\
&\quad = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y}\left[F_Z\left(\frac{r_k^{i,u,v+} - \rho\frac{x^j_{k+1}-c^i_k}{m^i_k}}{\sqrt{1-\rho^2}}\right) - F_Z\left(\frac{r_k^{i,u,v-} - \rho\frac{x^j_{k+1}-c^i_k}{m^i_k}}{\sqrt{1-\rho^2}}\right)\right] \\
&\qquad\qquad \times \mathbb{P}(\widehat{X}_{k+1}=x^j_{k+1})\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k),
\end{aligned} \tag{5.18}$$

where $\phi_2(y,\rho|x)$ is the conditional bivariate Gaussian density. It is worthwhile to note that this approximation to the joint probability, although derived differently, is identical to that of Callegaro et al. [2016]. The computational efficiency of this approximation is demonstrated in Sections 5.3.1 and 5.3.2.

5.2.2 Matrix Formulation

Throughout this section, the index $1 \le i \le N^x$ refers to time-step k and $1 \le j \le N^x$ refers to time-step $k+1$, and both are associated with the $\overline{X}$-process. For the $\overline{Y}$-process, the index $1 \le u \le N^y$ refers to time-step k and the index $1 \le v \le N^y$ refers to time-step $k+1$.

To initialize the JRMQ algorithm, the standard RMQ algorithm is applied to the $\overline{X}$-process and yields the quantizers $\mathbf{\Gamma}^x_k$ and associated probabilities $\mathbf{p}^x_k$ at each time-step $0 \le k \le n$. The following three variables are initialized

$$[\mathbf{\Gamma}^y_0]_1 = y_0, \qquad [\mathbf{p}^x_0]_1 = 1, \qquad [\mathbf{J}_0]_{1,1} = 1,$$

being the time-zero quantizer, associated probability and margined probability, respectively, of the $\overline{Y}$-process. The standard one-dimensional vector quantization algorithm (on the normal distribution) is used to produce $\mathbf{\Gamma}^y_1$ and $\mathbf{p}^y_1$, being the quantizer and associated probability vector of the $\overline{Y}$-process at the first time step. The corresponding joint probabilities at time-step one may then be computed using either (5.23) or (5.26), given at the end of this section, with $k = 0$ and $N^x = N^y = 1$.

Now consider the implementation of the recursive step from time-step k to $k+1$. Consider the time-step k quantizers

$$[\mathbf{\Gamma}^x_k]_i = x^i_k \qquad \text{and} \qquad [\mathbf{\Gamma}^y_k]_u = y^u_k,$$

of the independent and dependent processes, respectively, and the associated joint probability matrix $\mathbf{J}_k$, of size $N^x \times N^y$,

$$[\mathbf{J}_k]_{i,u} = \mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k),$$

all of which are assumed known (already computed). The rows of $\mathbf{J}_k$ are denoted by $\mathbf{J}^{(i)}_k$.

The time-step $k+1$ quantizer for the dependent process and associated probabilities are com-

puted as follows: Aside from an initial guess for $\mathbf{\Gamma}^y_{k+1}$, which is taken to be $\mathbf{\Gamma}^y_k$, initialize the N^y-element column vector

$$[\mathbf{c}_k]_u = \overline{c}^u_k$$

and a collection of N^y-element column vectors, indexed by i,

$$[\mathbf{m}_k]^{(i)}_u = \overline{m}^{i,u}_k,$$

in terms of the time-step k quantities listed above. For each iteration of the Newton-Raphson algorithm, three sets of matrices, indexed by i, are computed. The first two sets contain matrices of size $N^y \times N^y$, given by

$$\begin{aligned}[\mathbf{P}_{k+1}]^{(i)}_{u,v} &= \mathbb{P}(\widehat{Y}_{k+1} = y^v_{k+1} | \widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k), \\ &= F_Z(r^{i,u,v+}_{k+1}) - F_Z(r^{i,u,v-}_{k+1})\end{aligned}$$

and

$$[\mathbf{M}_{k+1}]^{(i)}_{u,v} = M_Z(r^{i,u,v+}_{k+1}) - M_Z(r^{i,u,v-}_{k+1}),$$

while the third contains matrices of size $N^y \times (N^y - 1)$, given by

$$[\mathbf{f}_{k+1}]^{(i)}_{u,v} = f_Z(r^{i,u,v+}_{k+1}).$$

The above matrices allow the gradient and the Hessian of the distortion for $\mathbf{\Gamma}^y_{k+1}$ to be written in simplified form. The N^y-element gradient vector is

$$\nabla D(\mathbf{\Gamma}^y_{k+1})^\top = \sum_{i=1}^{N^x} 2\mathbf{J}^{(i)}_k(((\mathbf{\Gamma}^y_{k+1}\mathbf{j}_{N^y})^\top - \mathbf{c}_k\mathbf{j}_{N^y}) \circ \mathbf{P}^{(i)}_{k+1} - (|\mathbf{m}^{(i)}_k|\mathbf{j}_{N^y}) \circ \mathbf{M}^{(i)}_{k+1}), \tag{5.19}$$

where $\circ$ is the Hadamard (or element-wise) product and $\mathbf{j}_d$ is defined to be a length-d row vector of ones. By specifying the column vector

$$[\Delta\mathbf{\Gamma}^y_{k+1}]_v = y^{v+1}_{k+1} - y^v_{k+1}, \tag{5.20}$$

with $1 \leq v \leq (N^y - 1)$, the $(N^y - 1)$-element off-diagonal of the tridiagonal Hessian matrix is given by

$$\mathbf{h}_{\mathsf{off}} = \sum_{i=1}^{N^x} -\frac{1}{2}\mathbf{J}^{(i)}_k((|\mathbf{m}^{(i)}_k|^{\circ-1}\mathbf{j}_{N^y-1}) \circ \mathbf{f}^{(i)}_{k+1} \circ (\Delta\mathbf{\Gamma}^y_{k+1}\mathbf{j}_{N^y})^\top) \tag{5.21}$$

and the N^y-element main diagonal by

$$\mathbf{h}_{\mathsf{main}} = \sum_{i=1}^{N^x} 2\mathbf{J}^{(i)}_k\mathbf{P}^{(i)}_{k+1} + [\mathbf{h}_{\mathsf{off}}|0] + [0|\mathbf{h}_{\mathsf{off}}]. \tag{5.22}$$

Here, $\circ - 1$ refers to the element-wise inverse.

Equations (5.19) to (5.22) provide a matrix representation of equations (5.12) to (5.15) and correspond to those in the matrix implementation of the single-dimensional RMQ case. This allows straightforward implementation of the Newton-Raphson algorithm described by (5.9), ultimately

yielding $\mathbf{\Gamma}^y_{k+1}$. It only remains to compute the necessary probabilities.

The elements of the joint probability matrix, $\mathbf{J}_{k+1}$, at time-step $k+1$, are computed using the bivariate Gaussian distribution as

$$\begin{aligned}
[\mathbf{J}_{k+1}]_{j,v} &= \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \mathbb{P}(\widehat{X}_{k+1} = x^j_{k+1}, \widehat{Y}_{k+1} = y^v_{k+1} | \widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k)\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k) \\
&= \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \Big(\Phi_2(r^{i,j+}_k, r^{i,u,v+}_k, \rho) - \Phi_2(r^{i,j-}_k, r^{i,u,v+}_k, \rho) \\
&\qquad -\Phi_2(r^{i,j+}_k, r^{i,u,v-}_k, \rho) + \Phi_2(r^{i,j-}_k, r^{i,u,v-}_k, \rho)\Big)\,[\mathbf{J}_k]_{i,u},
\end{aligned} \tag{5.23}$$

with the probabilities associated with the new quantizer given by

$$\mathbf{p}^y_{k+1} = \sum_{j=1}^{N^x} \mathbf{J}^{(j)}_{k+1}. \tag{5.24}$$

Finally, to compute the transition probability matrix for the time-step $k+1$, it is necessary to recompute the $\mathbf{P}_{k+1}$ matrix using the final regions associated with the new set of codewords at $k+1$. Then

$$\begin{aligned}
[\mathbf{P}^y_{k+1}]_{u,v} &= \frac{\mathbb{P}(\widehat{Y}_k = y^u_k, \widehat{Y}_{k+1} = y^v_{k+1})}{\mathbb{P}(\widehat{Y}_k = y^u_k)} \\
&= \frac{\sum_{i=1}^{N^x} \mathbb{P}(\widehat{Y}_{k+1} = y^v_{k+1} | \widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k)\mathbb{P}(\widehat{X}_k = x^i_k, \widehat{Y}_k = y^u_k)}{\mathbb{P}(\widehat{Y}_k = y^u_k)} \\
&= \frac{\sum_{i=1}^{N^x} [\mathbf{P}_{k+1}]^{(i)}_{u,v} [\mathbf{J}_k]_{i,u}}{[\mathbf{p}^y_k]_u}.
\end{aligned} \tag{5.25}$$

To compute the joint probabilities using the computationally efficient approximation, instead of the bivariate Gaussian distribution, (5.23) is replaced by

$$[\mathbf{J}_{k+1}]_{j,v} = \sum_{i=1}^{N^x}\sum_{u=1}^{N^y} \left[F_Z\left(\frac{r^{i,u,v+}_k - \rho\frac{x^j_{k+1} - c^i_k}{m^i_k}}{\sqrt{1-\rho^2}}\right) - F_Z\left(\frac{r^{i,u,v-}_k - \rho\frac{x^j_{k+1} - c^i_k}{m^i_k}}{\sqrt{1-\rho^2}}\right)\right] [\mathbf{p}^x_{k+1}]_j [\mathbf{J}_k]_{i,u}. \tag{5.26}$$

The time-step $k+1$ quantizer probabilities and transition probability matrix, (5.24) and (5.25), are now computed in terms of (5.26).

5.3 Option Pricing

5.3.1 European Option Pricing

In this section, the pricing of European options under the Stein and Stein [1991], Heston [1993] and SABR [Hagan et al., 2002] models are considered. The Stein-Stein and Heston models are both amenable to semi-analytical pricing using Fourier transform techniques, whereas an analytical approximation exists for both the Black and Bachelier implied volatility under the SABR model. The Fourier pricing technique implemented uses the little trap formulation of the characteristic function from Albrecher et al. [2006] for the Heston model, while the Schöbel and Zhu [1999] characteristic function formulation is used for the Stein-Stein model. The implied volatility approximation for

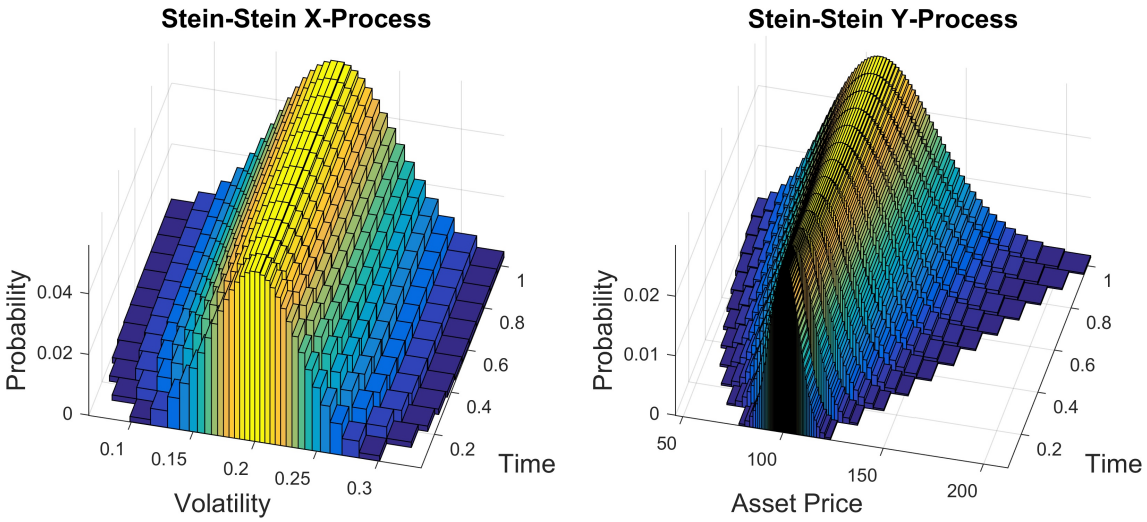

Figure 5.2: The quantizers through time for the Stein-Stein model generated by the JRMQ algorithm with $N^x = 30$ and $N^y = 60$.

the SABR model is the latest from Hagan et al. [2016].

The Stein-Stein example is used to illustrate the computational efficiency advantage of the new algorithm compared to the RMQ-based algorithm from Callegaro et al. [2016], whereas the Heston example serves to highlight the effectiveness of correctly modelling the zero-boundary behaviour of the independent process. For the SABR model, parameter sets were chosen that are difficult to handle with traditional methods, illustrating the flexibility of the JRMQ algorithm.

All simulations were executed using MATLAB 2016b on a computer with a 2.00 GHz Intel i-3 processor and 4 GB of RAM. All Monte Carlo simulations in this section used 500 000 paths with 120 time steps per path.

The Stein and Stein Model

The SDEs for the Stein-Stein model may be specified in the notation of (5.1) and (5.2) as

$$a^x(X_t) = \kappa(\theta - X_t), \qquad b^x(X_t) = \sigma,$$
$$a^y(Y_t) = rY_t, \qquad b^y(X_t, Y_t) = X_t Y_t,$$

and in the example considered the parameters chosen are $\kappa = 4$, $\theta = 0.2$, $\sigma = 0.1$, $r = 0.0953$, $\rho = -0.5$, $x_0 = 0.2$ and $y_0 = 100$, with the maturity of the option set at one year. These parameters are from Table 1 in Schöbel and Zhu [1999]. For the JRMQ algorithm, $n = 12$ time steps were used with $N^x = 30$ codewords at each step for the independent process and $N^y = 60$ codewords for the dependent process. The resultant quantizers for the Stein-Stein model are displayed in Figure 5.2.

Before pricing options under the Stein-Stein model, it is worth investigating the effect of approximating the joint probability. Two JRMQ algorithms for the Stein-Stein model are evolved until time-step k, the first using the joint probability approximation, (5.26), and the second using the bivariate Gaussian distribution to compute the joint probabilities, (5.23). Then a quantization grid is fixed for time-step $k+1$, and two sets of probabilities are computed across this grid as implied by the time-k quantizers of each of the aforementioned algorithms. In this way, the impact of the probability approximation can be directly measured at the time t_{k+1}. This is displayed for $k = 5$ in Figures 5.3 and 5.4, where $\rho = -0.5$ and $\rho = 0.1$, respectively.

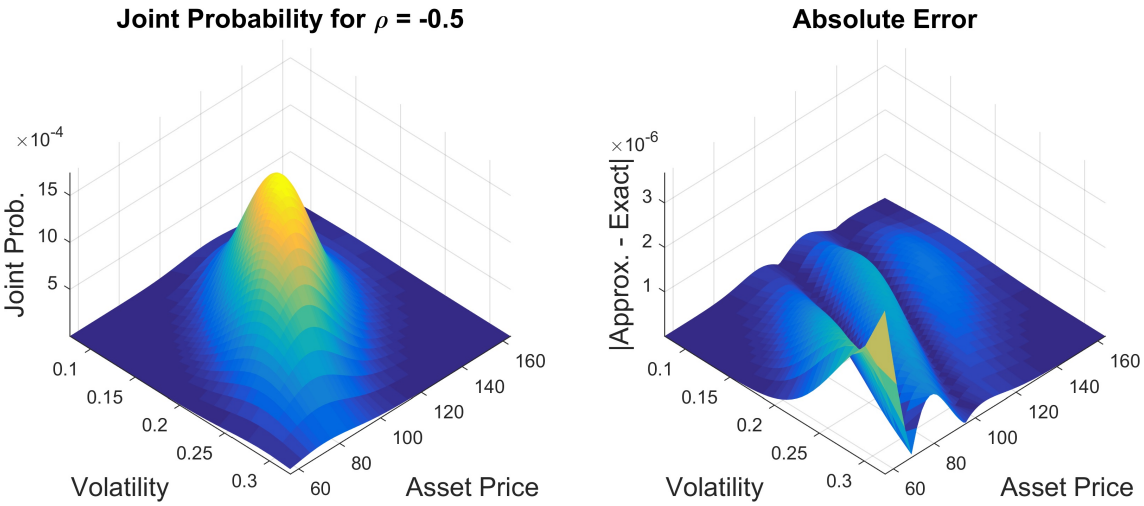

Figure 5.3: The error in the joint probability approximation at time-step $k = 6$, for the Stein-Stein model when $\rho = -0.5$.

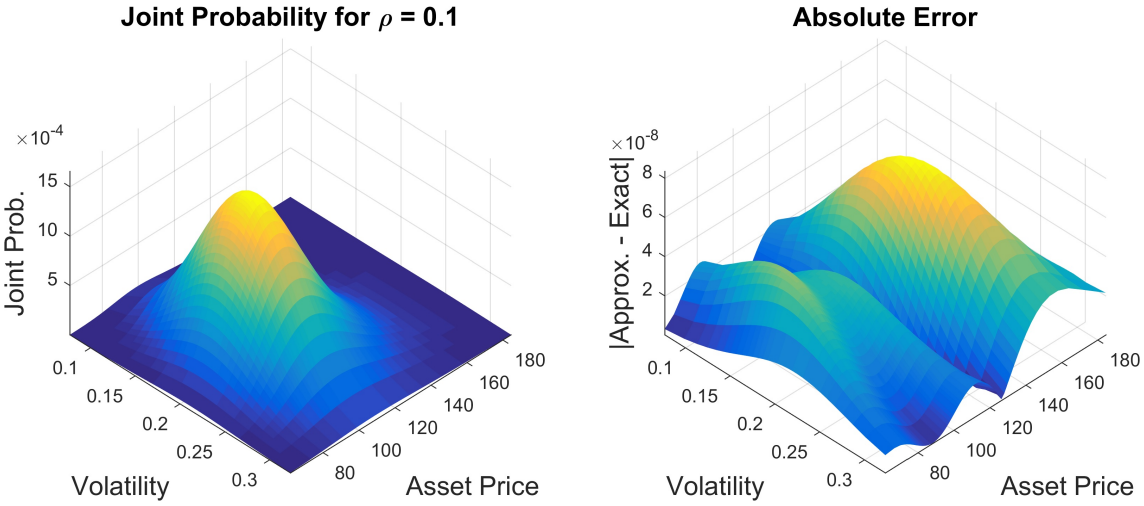

Figure 5.4: The error in the joint probability approximation at time-step $k = 6$, for the Stein-Stein model when $\rho = -0.1$.

In the first case, when $\rho = -0.5$, the error in the joint probability is two orders of magnitude smaller than the probability itself across the fixed grid. If the absolute correlation decreases, the impact of the approximation decreases as well, as the processes become less dependent. This can be seen in Figure 5.4 where the correlation is set to $\rho = 0.1$ and the error decreases dramatically. The left panels of Figures 5.3 and 5.4 also highlight how the correlation changes the shape of the resulting quantizer.

The left graph in Figure 5.5 displays the pricing error of four algorithms. The first is the JRMQ algorithm using the joint probability approximation from (5.18), the second is the JRMQ algorithm with the joint probabilities computed using the bivariate Gaussian distribution, the third is the RMQ-based algorithm from Callegaro et al. [2016], and the fourth is a two-dimensional standard Euler Monte Carlo simulation. Variable levels of moneyness are considered by changing the strike over the fixed initial asset price.

The JRMQ algorithm took 3.8 seconds to price all strikes when using the probability approximation and 77.2 seconds when using the bivariate Gaussian distribution. The algorithm from Callegaro et al. [2016] took 26.3 seconds to price all strikes, and the Monte Carlo simulation took

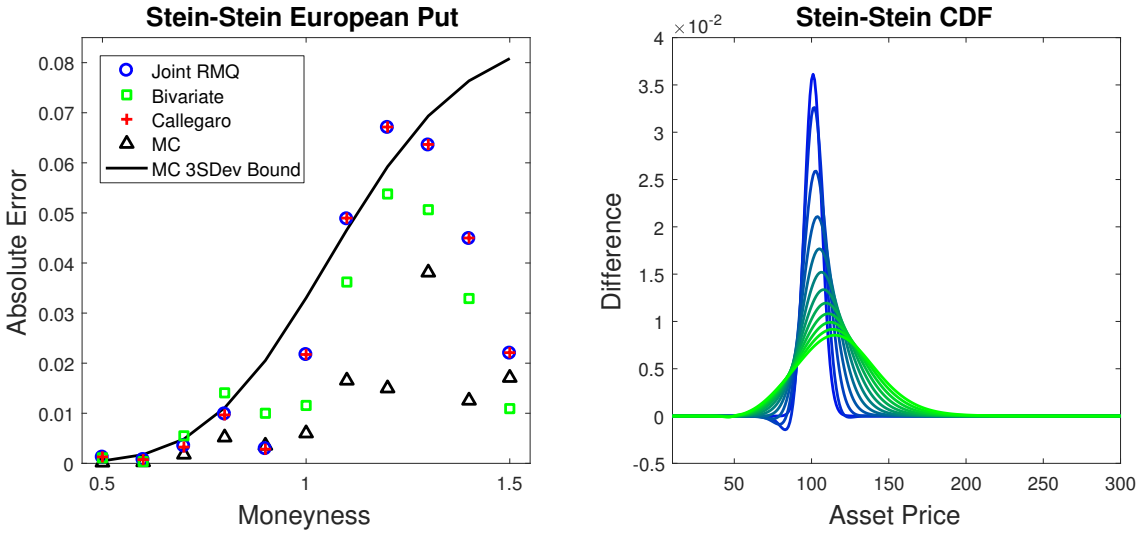

Figure 5.5: The European put pricing error under the Stein-Stein model as well as the difference between the quantization implied marginal distribution and the true marginal distribution at each time step.

6.6 seconds per strike.

The computation time of the JRMQ algorithm for this example was approximately 7 times faster than the algorithm of Callegaro et al. [2016], when using approximate joint probabilities. Despite this large decrease in computation time, the JRMQ algorithm prices with the same accuracy. Barring three points, both algorithms price to within the three standard deviation bound of the significantly higher resolution Monte Carlo simulation. Using the bivariate Gaussian distribution instead of the approximation significantly reduces the average error over the range of moneyness considered, but this is at the expense of a large increase in computation time. For this reason, the remaining applications use only the approximation.

Since the Stein-Stein model has a closed-form characteristic function, it is possible to compute the marginal distribution for the dependent process. The difference between this marginal distribution and the one computed using the JRMQ algorithm is presented in the right of Figure 5.5. The curve is blue at the initial time and changes color to green as it moves toward maturity. The maximum error is under 4% initially and decays to well under 1% as time advances. These errors are in line with those of the one-dimensional Euler RMQ case illustrated in Section 4.2.2.

The Heston Model

The SDEs for the Heston model may be specified in the notation of (5.1) and (5.2) as

$$a^x(X_t) = \kappa(\theta - X_t), \qquad b^x(X_t) = \sigma\sqrt{X_t},$$
$$a^y(Y_t) = rY_t, \qquad b^y(X_t, Y_t) = \sqrt{X_t}Y_t,$$

and in the example considered the parameters chosen are $\kappa = 2$, $\theta = 0.09$, $\sigma = 0.4$, $r = 0.05$, $\rho = -0.3$, $x_0 = 0.09$ and $y_0 = 100$, with the maturity of the option set at one year. These parameters are based on the SV-I parameter set from Table 3 of Lord et al. [2010], with σ adjusted from 1 to 0.4 to ensure that the Feller condition is satisfied for the square-root variance process.

The left panel of Figure 5.6 displays the pricing error for JRMQ compared with a two-dimensional fully truncated log-Euler scheme, suggested as the least-biased Monte Carlo scheme

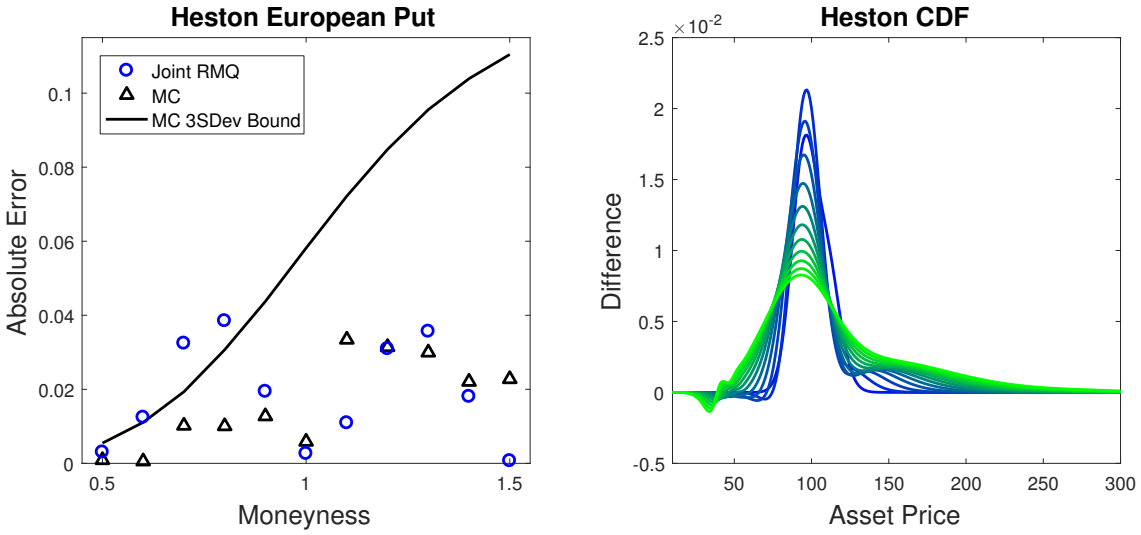

Figure 5.6: The European put pricing error under the Heston model as well as the difference between the quantization implied marginal distribution and the true marginal distribution at each time step.

for stochastic volatility models in Lord et al. [2010]. For the JRMQ algorithm, $n = 12$ time steps were used with $N^x = N^y = 30$ codewords at each step for both processes. The JRMQ algorithm took 1.4 seconds to price all strikes, whereas the Monte Carlo simulation took 7.8 seconds for a single strike.

Even though the Feller condition is satisfied, due to the discretization of time, there is a non-zero probability of the Euler approximation for the variance process becoming negative. This is handled in the RMQ algorithm by using a reflecting zero-boundary for the independent process, as detailed in Section 3.4. Modelling the boundary in this way leads to an increased accuracy in pricing, especially when compared to the Monte Carlo simulation.

The right of Figure 5.6 presents the error in the marginal distribution of the dependent process implied by the RMQ algorithm when compared to the distribution obtained from the characteristic function using the Fourier transform technique. The error is just over 2% initially and decreases to below 1% as time advances.

The SABR Model

The SDEs for the standard SABR model may be specified in the notation of (5.1) and (5.2) as

$$\begin{aligned} a^x(X_t) &= 0, & b^x(X_t) &= \nu X_t, \\ a^y(Y_t) &= 0, & b^y(X_t, Y_t) &= X_t Y_t^\beta, \end{aligned}$$

with $0 \leq \beta \leq 1$. A partial reason for the popularity of the SABR model is that the implied volatility may be computed using an analytical approximation [Hagan et al., 2002]. Further work has extended the original formula (see, for example, Oblój [2007] and Paulot [2015]), with the latest and most accurate approximation given in Hagan et al. [2016], which allows a more general specification of the volatility function.

In this section, European options are considered for two examples of extreme parameter sets that may arise in the context of interest rate modelling.

In Figure 5.7 the parameters chosen are $\beta = 0.7$, $\nu = 0.3$, $\rho = -0.3$, $x_0 = 20\%$ and $y_0 = 0.5\%$,

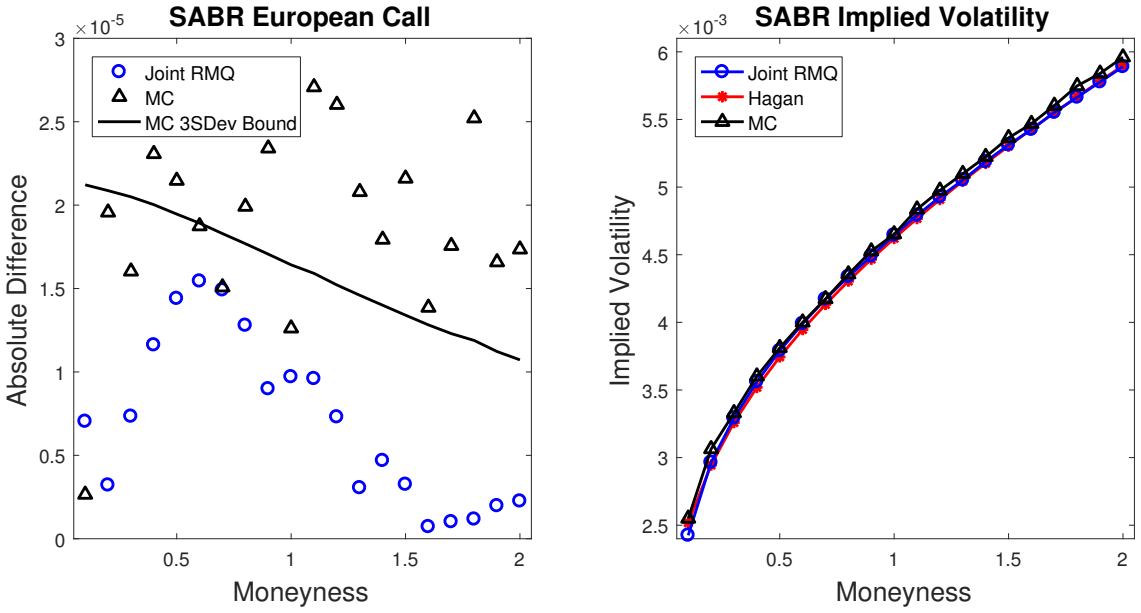

Figure 5.7: Pricing differences and implied Bachelier volatilities for the standard SABR model, using a parameter set applicable for interest rates.

with the maturity of the option set at one year. This parameter set is Test Case III from Chen et al. [2012], and was specifically chosen to be appropriate to the fixed income market and to illustrate the correct handling of zero-boundary behaviour. The reference price is the implied volatility formula with the boundary correction from Hagan et al. [2016].

For the JRMQ algorithm, $n = 24$ time steps were used with $N^x = N^y = 30$ codewords at each step for both processes. A reflecting zero-boundary was implemented for the dependent process. The Monte Carlo simulation utilized a fully-truncated Euler discretization scheme.

The three standard deviation bound in the left graph in Figure 5.7 indicates that the Monte Carlo simulation is not converging to the same result as the Hagan et al. [2016] implied volatility, used here as the reference price.

In their discussion, Chen et al. [2012] indicate that this is a challenging parameter set for traditional Monte Carlo simulations. Barring a single point, the JRMQ algorithm is more accurate than the Monte Carlo simulation across the range of strikes. It is also significantly faster to compute. The JRMQ algorithm took 5.3 seconds to price all strikes, whereas the Monte Carlo simulation took 13.4 seconds per strike, due to the much larger number of time steps.

In Figure 5.8, European call option pricing differences and corresponding implied Bachelier volatilities are displayed for the RMQ algorithm, the Hagan implied volatility approximation, and an Euler Monte Carlo simulation. The parameters chosen are $\beta = 0$, $\nu = 0.3691$, $\rho = -0.0286$, $X_0 = 0.68\%$, $Y_0 = 4.35\%$, with the maturity of the option set at one year. With $\beta = 0$, this case is specialized to the *Bachelier* or *normal* SABR model. This parameter set is Test Case I from Korn and Tang [2013] and it describes a challenging simulation environment with a low initial forward rate which is very volatile.

For the JRMQ algorithm, $n = 24$ time steps were used with $N^x = 10$ codewords at each step for the independent process and $N^y = 90$ codewords for the dependent process. The JRMQ algorithm took 5.5 seconds to price all strikes, whereas the Monte Carlo simulation took 5.6 seconds per strike.

Despite the extreme parameter set, all but two of the JRMQ prices fall well within the three

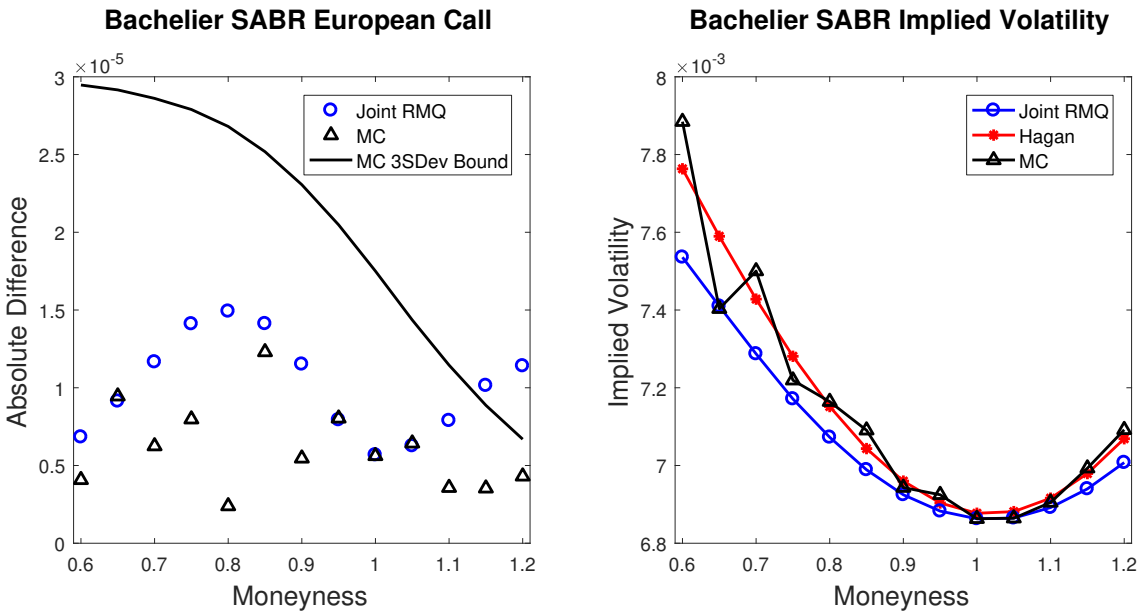

Figure 5.8: Implied Bachelier volatility and pricing error for the Bachelier SABR model.

standard deviation bound of the much higher resolution Monte Carlo simulation.

5.3.2 Exotic Option Pricing

An advantage of the JRMQ algorithm, similar to binomial and trinomial tree methods, is the ability to price many options off the grid that results from a single run. This is demonstrated in this section by using a single pass of the JRMQ algorithm to price European, Bermudan and barrier options, and volatility corridor swaps.

The SABR model parameters for all the examples in this section are $\beta = 0.9$, $\nu = 0.4$, $\rho = -0.3$, $X_0 = 0.4$ and $Y_0 = S_0 \exp(rT)$, where Y now describes the T-forward price of an equity asset with $S_0 = 100$, $r = 0.05$ and the maturity T is equal to one year. The JRMQ algorithm used $n = 24$ time steps with $N^x = 30$ codewords for the volatility process and $N^y = 60$ codewords for the forward price process. The Monte Carlo simulations are executed using a fully-truncated Euler scheme with 500 000 paths and 120 time-steps.

To generate the quantization grid, the JRMQ algorithm took 7.8 seconds for these parameters. The computational cost of generating derivative prices using the resulting grid is negligible in comparison.

The left graph in Figure 5.9 illustrates the difference in the prices of European put options using JRMQ and the prices using the implied volatility formula of Hagan et al. [2016]. The right graph shows the prices for a Bermudan put with monthly exercise opportunities using JRMQ and a least-squares Monte Carlo simulation. For each strike, computing an option price using Monte Carlo simulation takes approximately 14.5 seconds for the European options and 16.9 seconds for the Bermudan options. The high-level algorithm for pricing Bermudan options using a quantization grid is outlined in Section 4.3.2.

The left graph in Figure 5.10 shows the JRMQ and Monte Carlo prices for a discrete up-and-out put option, with monthly monitoring, where the barrier level is expressed as a multiple of the at-the-money strike. The right graph shows the prices for a series of volatility corridor swaps. The

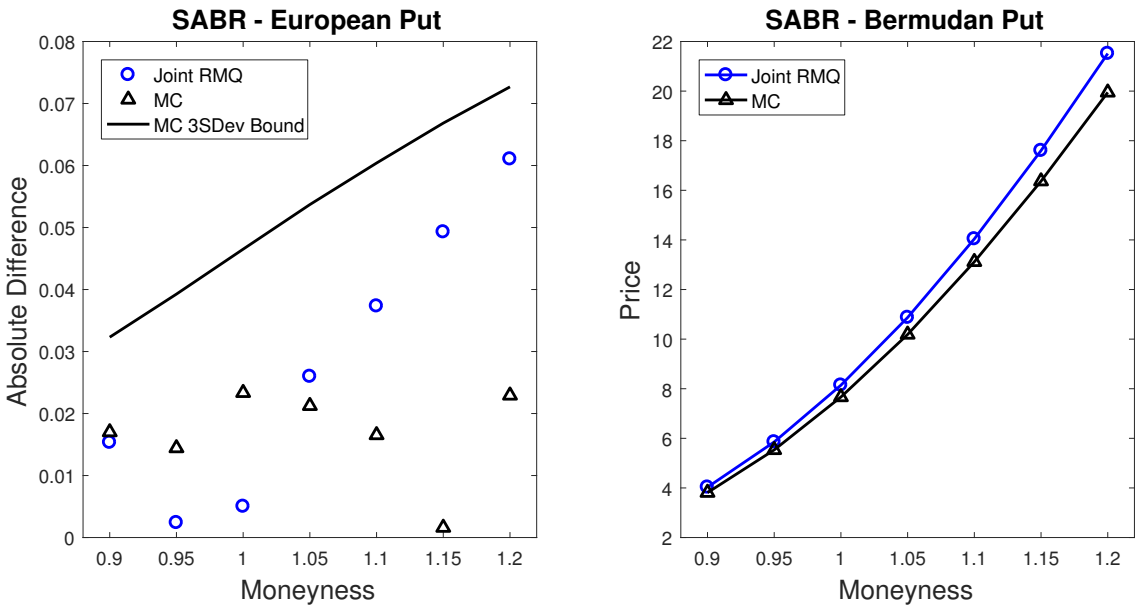

Figure 5.9: European and Bermudan put option prices for the SABR model.

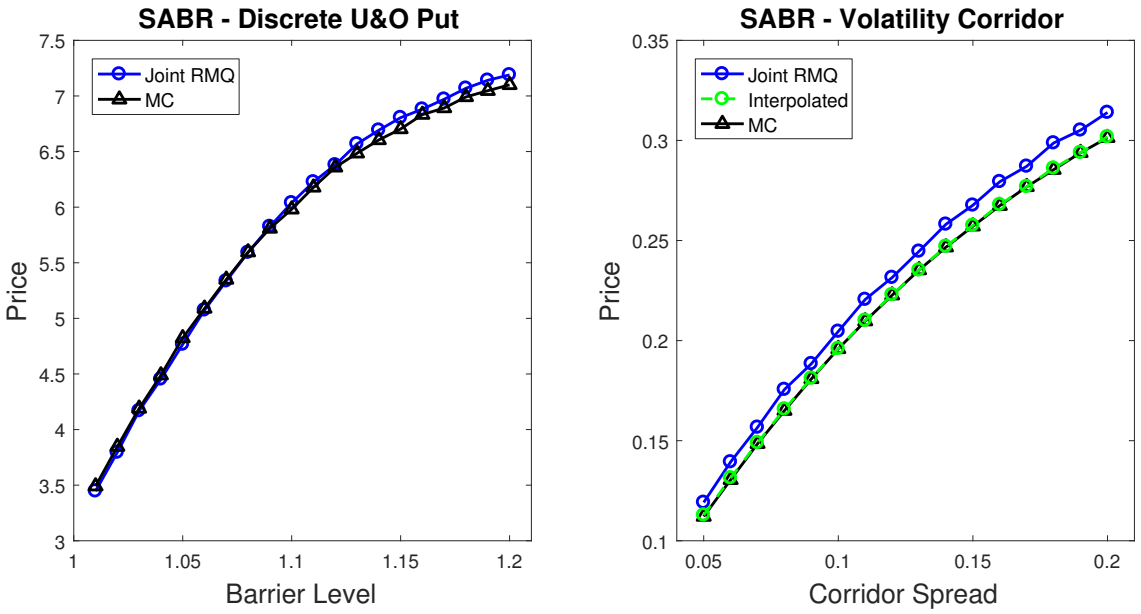

Figure 5.10: Price comparison for discrete up-and-out put options and volatility corridor swaps in the SABR model.

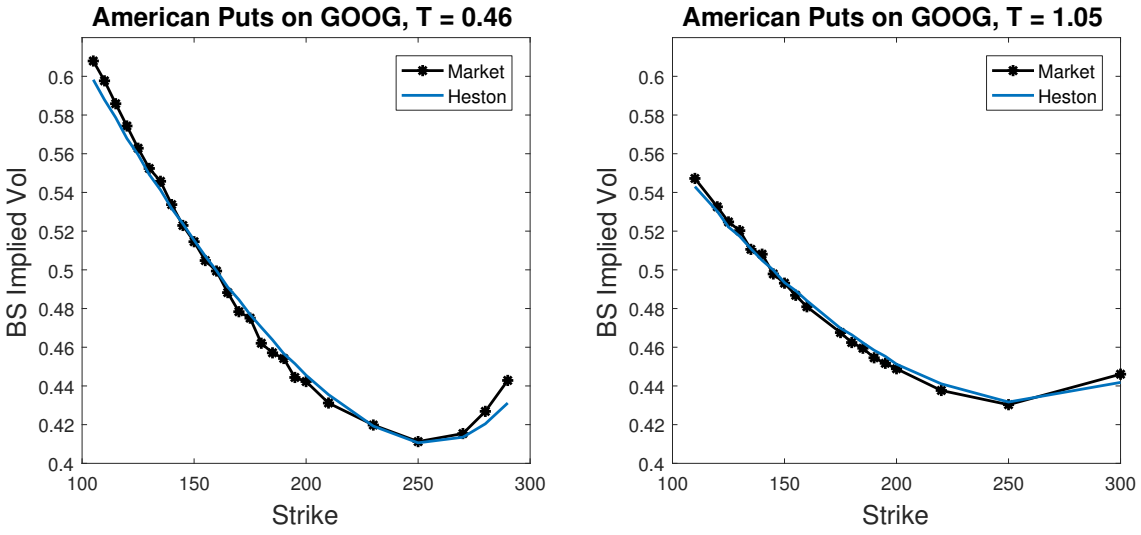

Figure 5.11: Calibration of the Heston model directly to American put options on GOOG on 03/01/2005.

payoff of a volatility corridor swap is given by

$$\frac{1}{T}\int_0^T X_z \mathbb{I}_{\{L<S_z<H\}}\, dz, \tag{5.27}$$

where $S_t = Y_t \exp(-r(T-t))$ is the asset price in our deterministic interest-rate framework and L and H specify the corridor of the asset price in which the volatility is accumulated. The algorithm for pricing volatility corridor swaps on a quantization grid in the stochastic volatility setting is presented in Callegaro et al. [2016] and uses a left-endpoint approximation to the integral in (5.27)

The corridor spreads on the x-axis represent a percentage bound around the initial asset price value, that is, the lower bound of the corridor is given by $L = S_0(1-c)$ and the upper bound by $H = S_0(1+c)$, where c is the corridor spread. The vertical gap between the prices generated by the Monte Carlo simulation and the RMQ algorithm is partially due to the increased accuracy of the Monte Carlo simulation when using simple quadrature to approximate (5.27), as a result of the large number of time steps used. For a single barrier value or a single corridor spread, the Monte Carlo simulation takes approximately 15.2 seconds and 16.3 seconds to price these derivatives.

The accuracy of JRMQ volatility corridor swap prices can be improved without using additional time steps. An increase in the accuracy of the approximation to the integral (5.27) is achieved by interpolating both the asset price and the volatility over each interval, see Appendix C. The improved accuracy of this interpolated JRMQ price is displayed in the right graph of Figure 5.10.

5.4 Calibration

Following the approach of Section 4.4, as a proof-of-concept example the Heston model is calibrated using the JRMQ algorithm directly to American put options on GOOG on 03/01/2005. Two different maturities are considered and the calibrated and market Black-Scholes implied volatilities are displayed in Figure 5.11. The parameter set is given by

$$\Theta = \{X_0,\ \sigma,\ \kappa,\ \theta,\ \rho\}$$

	X_0	σ	κ	θ	ρ	$F(\Theta)$
T_1	0.2	0.898	0.1	0.1	-0.63	0.0031
T_2	0.204	0.599	0.155	0.171	-0.582	0.0005

Table 5.1: Summary of the results for the Heston model calibrated to American put options on GOOG on 03/01/2005.

and the calibrated parameters are summarized in Table 5.1, along with the final value of the objective function. The option data is the same as was used for Section 4.4 and summarized in the left of Table 4.1.

Figure 5.11 illustrates how the Heston model better captures the shape of the market implied volatility curve as compared to the CEV model, see Figure 4.9. Intuitively, there are more degrees of freedom in the calibration of the model, so this is to be expected. The final value of the objective function is smaller than that for the CEV model for both maturities, although only slightly so for the first maturity. The calibrated parameters are fairly stable across the two maturities, indicating that the Heston model could be a candidate for calibration to the volatility surface.

The purpose of this simple calibration example is to illustrate the robustness of the JRMQ algorithm, especially when modified to handle the zero boundary. As noted in Section 4.4, the Hessian of the distortion function can become ill-conditioned and thus fail to be invertible. Heuristically, this tends to happen when the considered parameter set is far away from the minimum of the objective function and thus does not have a serious effect on the minimization procedure.

Although all five parameters of the Heston model are calibrated using minimization in this example, it must be emphasized that for several of the parameters it would be more appropriate to attempt to recover them using filtering techniques. An unscented Kalman filter would be a candidate to recover the hidden state, X_0, as well as parameters that have a stable interpretation through time, like the rate of mean reversion, κ, and the mean reversion level, θ.

Chapter 6

Valuation of Long-dated Contracts Under the Real-world Measure

6.1 Overview

It has been argued that requiring the existence of a risk-neutral measure is an unrealistic modelling constraint, severe enough to limit the efficacy of long-term pricing and hedging strategies [Hulley and Platen, 2012].

It is, however, possible to derive a unified framework for portfolio optimization, derivative pricing and risk management without the need for an equivalent martingale measure, as described by Platen [2006], the seminal work on the *benchmark approach*. Under this framework, the Law of One Price no longer holds [Platen, 2008], and the Fundamental Theorems of Asset Pricing [Delbaen and Schachermayer, 1994, 1998] do not apply, as the traditional no-arbitrage concept is relaxed. As a consequence, certain long-dated contracts can be less expensively replicated than suggested by the classical theory.

Central to the benchmark approach is the concept of the growth-optimal portfolio (GOP), first explored by Kelly [1956]. This portfolio strategy is such that, when denoted in units of the GOP, every asset becomes a supermartingale. This allows the derivation of the real-world pricing theorem; yielding the least expensive prices under the real-world probability measure.

The goal of this chapter is to provide a simple numerical toolbox for the pricing of long-dated contracts under the benchmark approach by using recursive marginal quantization (RMQ). RMQ was introduced by Pagès and Sagna [2015] and is a numerical technique for the optimal approximation of functionals of solutions to stochastic differential equations (SDEs).

The main contribution of this chapter is two-fold. First, analytic European option prices are derived for the time-dependent constant elasticity of variance (TCEV) model for the GOP. This is an extension of formulae presented by Baldeaux et al. [2014], which generalize the previous option pricing formulae from Miller and Platen [2008, 2010]. Secondly, RMQ and JRMQ are shown to be highly effective tools for pricing both European options on the GOP, under the assumption of constant interest rates, and zero-coupon bond options, under a stochastic short-rate model, namely the 3/2-model from Ahn and Gao [1999]. This efficient pricing mechanism allows for the wider application of the benchmark approach.

The chapter proceeds as follows. In Section 6.2 the benchmark approach is briefly reviewed in the context of a two-asset scalar diffusion market. The form of the growth-optimal portfolio strategy is specified, and the real-world pricing theorem is derived. The TCEV model is introduced

and its probabilistic features are detailed.

In the first part of Section 6.3, analytic European option pricing formulae are derived for the TCEV model under the assumption of a constant interest rate. The pricing efficiency of these formulae is compared with the approximate prices obtained using RMQ. The latter part of Section 6.3 deals with stochastic interest rates, specifically the hybrid model introduced by Baldeaux et al. [2015]. Analytic zero-coupon bond prices are presented under the assumption of independence between the stochastic short rate and the GOP. The influence of correlation is briefly explored. Finally, JRMQ is used to efficiently price European options on zero-coupon bonds in this framework.

6.2 The Benchmark Approach

This section presents a one-dimensional version of the continuous financial market presented by Platen [2006], similar to the introduction given by Platen and Heath [2006, Chap. 9]. For full mathematical rigor see the original derivation of real-world pricing by Platen [2002], and the extension to the jump-diffusion framework [Platen and Heath, 2006, Chap. 14]. The most general derivation of the "Law of Minimal Price", of which real-world pricing is a result, is presented by Platen [2008].

Assume the existence of a filtered probability space, $(\Omega, \mathscr{F}, \underline{\mathscr{F}}, \mathbb{P})$. The filtration $\underline{\mathscr{F}} = (\mathscr{F}_t)_{t\in[0,\infty)}$ is assumed to satisfy the usual conditions.

Consider a continuous financial market consisting of two assets: a risk-free savings account, $S^0 = \{S^0_t,\ t \in [0,\infty)\}$, and a risky primary security, $S^1 = \{S^1_t,\ t \in [0,\infty)\}$. They respectively obey the SDEs

$$dS^0_t = S^0_t r_t\, dt \tag{6.1}$$

and

$$dS^1_t = S^1_t(a_t\, dt + b_t\, dW_t), \tag{6.2}$$

with $r = \{r_t,\ t \in [0,\infty)\}$, the adapted short rate process. Here $b = \{b_t,\ t \in [0,\infty)\}$ is a predictable and strictly positive process known as the diffusion coefficient of S^1 with respect to the standard Brownian motion, W, and is assumed to satisfy

$$\int_0^T b_s^2\, ds < \infty$$

almost surely for $T \in [0,\infty)$, a finite time-horizon. It is also assumed that $a = \{a_t,\ t \in [0,\infty)\}$, known as the drift, is a predictable process satisfying

$$\int_0^T |a_s|\, ds < \infty,$$

almost surely. It is assumed that the SDE (6.2) has a unique strong solution, see, for example, Platen and Heath [2006].

In the market considered, the number of risky securities is the same as the number of Wiener processes. Refer to Platen [2004] for the case where there are more sources of uncertainty than traded securities.

Defining the market price of risk (MPOR) process as

$$\theta_t := b_t^{-1}(a_t - r_t),$$

allows the SDE (6.2) to be re-written as

$$dS_t^1 = S_t^1 \left((r_t + b_t\theta_t)\, dt + b_t\, dW_t) \right). \tag{6.3}$$

It is assumed that the absolute value of the MPOR process is always finite.

6.2.1 The Growth Optimal Portfolio

The GOP has often been attributed to Kelly [1956]. Hakansson and Ziemba [1995] state in their review of the growth optimal investment strategy that the GOP was already implied by Bernoulli's solution to the St. Petersburg Paradox as early as 1738 (see Samuelson [1977] for this interesting digression). An important application of the GOP to claim valuation and long-run portfolio growth is Long [1990], where it is shown that, under certain constraints, risk-neutral pricing is equivalent to pricing under the real-world probability measure using the GOP as the numeraire.

The GOP is the central object of the benchmark approach and hence real-world pricing. In a general semimartingale framework, Karatzas and Kardaras [2007] show that the "no unbounded profit with bounded risk" no-arbitrage condition is necessary and sufficient for the existence of the GOP. This condition is placed into its proper context alongside the range of applicable arbitrage statements by Fontana [2015].

In the market defined above, a predictable stochastic vector process $\delta = \{\delta_t = (\delta_t^0, \delta_t^1),\ t \in [0,T]\}$ is called a *strategy* if, for all $t \in [0,T]$, the stochastic Itô integrals

$$\int_0^t \delta_s^0\, dS_s^0 \qquad \text{and} \qquad \int_0^t \delta_s^1\, dS_s^1$$

exist. Denote by $S^\delta = \{S_t^\delta, t \in [0,T]\}$ the time-t value of the portfolio process associated with the strategy δ, defined as

$$S_t^\delta = \delta_t^0 S_t^0 + \delta_t^1 S_t^1.$$

A strategy and its corresponding portfolio process are said to be self-financing if

$$dS_t^\delta = \delta_t^0\, dS_t^0 + \delta_t^1\, dS_t^1,$$

for all $t \in [0,T]$. The self-financing condition ensures that instantaneous changes in the value of the portfolio are due to changes in the prices of the constituent securities and not to external deposits or withdrawals. Only self-financing portfolios are considered in the following.

At this point it is necessary to introduce the concept of *admissable* portfolios, to avoid the arbitrage opportunities generated by traditional doubling strategies. Admissable strategies are usually either constrained via an absolute lower bound or an integrability condition (see Hunt and Kennedy [2004, Chap. 7] for a discussion). Only the set, $\mathcal{V}^+$, of strictly positive portfolios will be considered, thus providing an absolute lower bound at zero. For $S^\delta \in \mathcal{V}^+$, define the portfolio fraction

$$\pi_{\delta,t}^j = \frac{\delta_t^j S_t^j}{S_t^\delta},$$

as the fraction of the total portfolio value invested in each asset, S_t^j, for $j \in \{0,1\}$. Portfolio fractions can be negative but must always sum to one. Using (6.3), the SDE for a self-financing

portfolio in $\mathcal{V}^+$ can now be written as

$$dS_t^\delta = S_t^\delta \left((r_t + \pi_{\delta,t}^1 b_t \theta_t)\, dt + \pi_{\delta,t}^1 b_t\, dW_t \right).$$

A simple application of Itô's formula provides the SDE for the logarithm of the portfolio

$$d \log S_t^\delta = g_{\delta,t}\, dt + \pi_{\delta,t}^1 b_t\, dW_t,$$

with portfolio growth rate

$$g_{\delta,t} = r_t + \pi_{\delta,t}^1 b_t \theta_t - \frac{1}{2}(\pi_{\delta,t}^1 b_t)^2. \tag{6.4}$$

The GOP is the portfolio that maximises this growth rate, that is, the drift of the log-portfolio. Mathematically, a strictly positive portfolio value process, $S^{\delta_*} = \{S_t^{\delta_*}, t \in [0,T]\}$, is a growth optimal portfolio if, for all $t \in [0,T]$ and all $S^\delta \in \mathcal{V}^+$, the inequality

$$g_{\delta_*,t} \geq g_{\delta,t}$$

holds almost surely. From the first-order condition for the maximum of the growth rate (6.4), the optimal fraction to be invested in S^1 can be found as

$$\pi_{\delta_*,t}^1 = b_t^{-1} \theta_t.$$

Consequently the SDE for the GOP is given by

$$dS_t^{\delta_*} = S_t^{\delta_*}((r_t + \theta_t^2)\, dt + \theta_t\, dW_t), \tag{6.5}$$

with $t \in [0,T]$ and $S_0^{\delta_*} > 0$. Contingent claims can now be priced under the real-world probability measure using the GOP as the numeraire or *benchmark*, as will be detailed below.

6.2.2 Real-world Pricing

Using the GOP as benchmark and numeraire, consider the evolution of a benchmarked portfolio given by the ratio

$$\hat{S}_t^\delta = \frac{S_t^\delta}{S_t^{\delta_*}}.$$

Itô's formula provides the SDE

$$d\hat{S}_t^\delta = (\delta_t^1 \hat{S}_t^1 b_t - \hat{S}_t^\delta \theta_t)\, dW_t, \tag{6.6}$$

in terms of $\hat{S}_t^1 = \frac{S_t^1}{S_t^{\delta_*}}$, the benchmarked security price process.

Because no drift is present in (6.6), it is clear that benchmarked portfolios form $(\underline{\mathscr{F}}, \mathbb{P})$-local martingales. Thus, by Fatou's lemma, all non-negative portfolios, when benchmarked, are $(\underline{\mathscr{F}}, \mathbb{P})$-supermartingales[1].

Define a non-negative contingent claim, V_τ, that matures at a stopping time $\tau \in [0,T]$, as an $\mathscr{F}_\tau$-measurable payoff that possesses a finite expectation when benchmarked. Note that V_τ need not be square-integrable. Let S_t^V denote a non-negative self-financing portfolio that replicates the claim, i.e., $S_\tau^V = V_\tau$. Then

$$\frac{S_t^V}{S_t^{\delta_*}} \geq \mathbb{E}\left[\frac{V_T}{S_T^{\delta_*}} \middle| \mathscr{F}_t \right]$$

[1] For a simple proof of this classic result, see Platen and Heath [2006, Lemma 5.2.3].

holds by the supermartingale property of benchmarked non-negative self-financing portfolios.

A security price process, equivalent to a self-financing, replicating portfolio, is called *fair* if its benchmarked value forms an $(\underline{\mathscr{F}}, \mathbb{P})$-martingale (in the classical risk-neutral setting, this notion of a fair process is equivalent to that proposed by Geman et al. [1995]). Under the benchmark approach this leads to the minimal possible price, the desired result for this section.

Theorem 6.2.1 (Real-world Pricing)**.** *For any fair security price process,* $V = \{V_t,\ t \in [0,T]\}$, $T \in (t, \infty)$*, one has the real-world pricing formula*

$$V_t = S_t^{\delta_*} \mathbb{E}\left[\frac{V_T}{S_T^{\delta_*}}\Big|\mathscr{F}_t\right]. \tag{6.7}$$

The expectation in Theorem 6.2.1 is taken under the real-world probability measure, $\mathbb{P}$. Under the assumption that one can perform an equivalent probability measure change, following Geman et al. [1995], the candidate Radon-Nikodym derivative process to move from the current real-world numeraire-measure pair, $(S^{\delta_*}, \mathbb{P})$, to the risk-neutral numeraire-measure pair, $(S^0, \mathbb{P}_\theta)$, is

$$\lambda_\theta(t) = \frac{d\mathbb{P}_\theta}{d\mathbb{P}}\Big|_{\mathscr{F}_t} = \frac{S_t^0}{S_0^0}\frac{S_0^{\delta_*}}{S_t^{\delta_*}} = \frac{\hat{S}_t^0}{\hat{S}_0^0}. \tag{6.8}$$

If $\lambda_\theta(t)$ is a strictly positive $(\underline{\mathscr{F}}, \mathbb{P})$-martingale (and not a strict local-martingale), then the probability measure change can indeed be performed, yielding

$$V_t = S_t^{\delta_*}\mathbb{E}\left[\frac{V_T}{S_T^{\delta_*}}\Big|\mathscr{F}_t\right] = \mathbb{E}\left[\frac{\lambda_\theta(T)}{\lambda_\theta(t)}\frac{S_t^0}{S_T^0}V_T\Big|\mathscr{F}_t\right] = \mathbb{E}_\theta\left[\frac{S_t^0}{S_T^0}V_T\Big|\mathscr{F}_t\right],$$

by Bayes' theorem, where the last expression is the classical risk-neutral pricing formula.

In this way, risk-neutral pricing is a special case of real-world pricing, applicable only when $\lambda_\theta(t)$ describes a strictly positive $(\underline{\mathscr{F}}, \mathbb{P})$-martingale. This translates into a constraint on the market price of risk θ_t, the volatility of the GOP. This volatility must be specified so that $\lambda_\theta(t)$ is a martingale, which is, for instance, the case if θ_t satisfies Novikov's condition or, more generally, Kazamaki's condition [Revuz and Yor, 1999]. To compute the expectation in Theorem 6.2.1, the GOP must be modelled explicitly. A realistic model for the GOP, which excludes risk-neutral pricing, is presented in the next section.

6.2.3 Modelling the GOP

Consider a simple two-asset market consisting only of the risk-free bank account and the growth-optimal portfolio, obeying (6.1) and (6.5), respectively. The *discounted* GOP is then described by

$$\bar{S}_t^* = \frac{S_t^*}{S_t^0},$$

which means that

$$d\bar{S}_t^* = \bar{S}_t^*(\theta_t^2\,dt + \theta_t\,dW_t).$$

The δ_*-superscript has been dropped from the GOP notation for simplicity. To compute the expectation in (6.7), the MPOR process, θ_t, must be modelled explicitly.

Baldeaux et al. [2014] propose the TCEV model for the GOP. It is parsimonious, tractable, reliably estimated and can provide explicit formulae for various derivatives and their hedge ratios. The TCEV model is specified by

$$\theta_t := c\left(\frac{\bar{S}^*_t}{\alpha_t}\right)^{a-1} \qquad \text{and} \qquad \alpha_t := \alpha_0 e^{\eta t}.$$

Here the parameter restrictions are $c > 0$, $a \in (-\infty, 1)$, $\alpha_0 > 0$ and $\eta > 0$. Note that the TCEV model generalizes both the minimal market model (MMM), first introduced by Platen [2001] and analyzed in detail by Miller and Platen [2008], and the modified constant elasticity of variance model (MCEV) from Heath and Platen [2002].

By direct substitution, the SDE for the discounted GOP under the TCEV model is

$$d\bar{S}^*_t = c^2\alpha_t^{2(1-a)}\left(\bar{S}^*_t\right)^{2a-1}dt + c\alpha_t^{1-a}\left(\bar{S}^*_t\right)^a dW_t. \tag{6.9}$$

The behaviour of this process is described by Proposition 6.2.2 below, which appears in a slightly modified form in Baldeaux et al. [2014].

Proposition 6.2.2. *The process $\bar{S}^* = \{\bar{S}^*_t,\ t \geq 0\}$ satisfies the following equality in distribution*

$$\bar{S}^*_t \overset{(d)}{=} X_{\varphi(t)}^{\frac{1}{2(1-a)}} = X_{\varphi(t)}^{\left(\frac{\delta}{2}-1\right)},$$

where $X = \{X_\varphi,\ \varphi \geq 0\}$ is a squared Bessel process of dimension $\delta = \frac{3-2a}{1-a}$ in φ-time and the time-transformation is given by

$$\varphi(t) = \frac{(1-a)\alpha_0^{2(1-a)}c^2}{2\eta}\left(e^{2(1-a)\eta t} - 1\right).$$

Proof. Define $Y_t = \left(\frac{\alpha_0}{\alpha_t}\bar{S}^*_t\right)^{2(1-a)}$ such that

$$dY_t = \left(\alpha_0^{2(1-a)}c^2(1-a)(3-2a) - 2(1-a)\eta Y_t\right)dt + 2(1-a)\alpha_0^{1-a}c\sqrt{Y_t}\,dW_t.$$

This is a CIR-process and thus by Proposition B.1, in Appendix B,

$$Y_t = e^{-2(1-a)\eta t}X_{\varphi(t)}.$$

Since $\bar{S}^*_t = e^{\eta t}Y_t^{\frac{1}{2(1-a)}}$, this completes the proof. □

An immediate consequence of the relationship between the discounted GOP and a squared Bessel process (BESQ) of dimension $\delta = \frac{3-2a}{1-a} > 2$, is that the discounted GOP never attains zero (see Appendix B.1). A more subtle consequence is that modelling the GOP in this way precludes the existence of an equivalent risk-neutral probability measure.

As in Section 6.2.2, the candidate Radon-Nikodym derivative process to move to the risk-neutral measure is given by (6.8). However, now

$$\hat{S}^0_t = \frac{1}{\bar{S}^*_t} = X_{\varphi(t)}^{1-\frac{\delta}{2}},$$

and from the symmetry relationship derived for BESQ processes in Appendix B.1, the process on the right-hand side of the above expression is a *strict* local martingale. Thus, the 'risk-neutral measure' induced by this Radon-Nikodym derivative process will not be a probability measure.

Alternatively, consider the existence of a measure, $\mathbb{P}_\theta$, such that the discounted GOP is potentially a martingale under this measure,

$$d\bar{S}_t^* = c\alpha_t^{1-a}\left(\bar{S}_t^*\right)^a \, dW_t^\theta. \tag{6.10}$$

Using the same transformation as in the proof of Proposition 6.2.2, $Y_t = \left(\frac{\alpha_0}{\alpha_t}\bar{S}_t^*\right)^{2(1-a)}$, it is straightforward to show that under this hypothetical measure the discounted GOP is the power of a BESQ process of dimension $\delta_\theta = \frac{1-2a}{1-a} < 2$. As this process has a non-zero probability of attaining zero in finite time, the measure $\mathbb{P}_\theta$ cannot be equivalent to $\mathbb{P}$, the original real-world probability measure under which X has dimension $\delta = \frac{3-2a}{1-a} > 2$ and never hits zero.

6.3 Option Pricing

In this section, the extended RMQ algorithm from Chapter 4 is used to provide fast and accurate pricing for the benchmark approach.

Initially, analytic European option prices are derived under the assumption of constant interest rates. These formulae generalize those found in Miller and Platen [2008, 2010] for the minimal market and modified constant elasticity of variance models, respectively. Although these formulae are analytic, they can be numerically expensive to compute and are contrasted to the fast, but approximate, prices obtained via RMQ. Furthermore, Bermudan options on the GOP are priced using traditional Monte Carlo methods, with their accuracy, speed and efficiency compared with that of recursive marginal quantization.

The second subsection deals with the hybrid model introduced by Baldeaux et al. [2015]. The hybrid model combines the TCEV model for the GOP with a 3/2 stochastic short-rate model. Baldeaux et al. [2015] derive analytic zero-coupon bond prices under the assumption that the GOP is independent of the short rate. In this section, numerical experiments show that these prices are well approximated with RMQ. The effect of the independence assumption on the zero-coupon bond prices is also investigated. Lastly, European options on zero-coupon bonds are priced using both traditional Monte Carlo methods as well as RMQ.

All simulations were performed using MATLAB 2016b on a computer with a 2.00 GHz Intel i-3 processor and 4 GB of RAM.

6.3.1 Constant Short Rates

Expressions similar to those derived in Propositions 6.3.1 and 6.3.2 below appear in Baldeaux et al. [2014], where the strike was selected to be a constant multiple of the savings account. In this way, Baldeaux et al. [2014] were able to avoid specifying a model for the short rate by restricting the class of strikes they considered.

Proposition 6.3.1. *Assuming a constant interest rate r, the real-world price, $p_{T,K}(t, S_t^*)$, of a European put option on the GOP at time t with expiry T and strike K is given by*

$$\begin{aligned} p_{T,K}(t,S_t^*) = &-\bar{S}_t^*\beta(t)\chi'^2\left(\frac{\widetilde{K}}{\Delta\varphi};\delta,\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi}\right) \\ &+ K\frac{\beta(t)}{\beta(T)}\left[\chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi};\delta-2,0\right) - \chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi};\delta-2,\frac{\widetilde{K}}{\Delta\varphi}\right)\right], \end{aligned}$$

where

$$\widetilde{K} = \left(\frac{K}{\beta(T)}\right)^{2(1-a)}, \qquad \beta(t) = \exp(rt),$$
$$\delta = \frac{3-2a}{1-a}, \qquad \Delta\varphi = \varphi(T) - \varphi(t),$$

with $\chi'^2(x;\delta,\lambda)$ *representing the noncentral chi-squared distribution, evaluated at* x *with degrees of freedom* δ *and non-centrality parameter* λ*, and* $\varphi(t)$ *is defined in Proposition 6.2.2.*

Proof. When the short rate is constant, the savings account is deterministic, with

$$S_t^0 = \exp(rt) =: \beta(t).$$

As is standard, the expectation of the numeraire denominated payoff can be expressed as the difference of two expectations,

$$\begin{aligned} p_{T,K}(t,S_t^*) &= \mathbb{E}\left[\frac{S_t^*}{S_T^*}(K-S_T^*)^+\middle|\mathscr{F}_t\right] \\ &= \mathbb{E}\left[\frac{\bar{S}_t^*}{\bar{S}_T^*}\beta(t)\left(\frac{K}{\beta(T)}-\bar{S}_T^*\right)^+\middle|\mathscr{F}_t\right] \\ &= \bar{S}_t^*\frac{\beta(t)}{\beta(T)}K\mathbb{E}\left[\frac{1}{\bar{S}_T^*}\mathbb{I}_{\{\frac{K}{\beta(T)}>\bar{S}_T^*\}}\middle|\mathscr{F}_t\right] - \bar{S}_t^*\beta(t)\mathbb{E}\left[\mathbb{I}_{\{\frac{K}{\beta(T)}>\bar{S}_T^*\}}\middle|\mathscr{F}_t\right]. \end{aligned}$$

Using Proposition 6.2.2, the first expectation can be rewritten in terms of the power of a squared Bessel process of dimension $\delta = \frac{3-2a}{1-a} > 2$,

$$\begin{aligned} \mathbb{E}\left[\frac{1}{\bar{S}_t^*}\mathbb{I}_{\{\frac{K}{\beta(T)}>\bar{S}_T^*\}}\middle|\mathscr{F}_t\right] &= \mathbb{E}\left[X_{\varphi(T)}^{-\left(\frac{\delta}{2}-1\right)}\mathbb{I}_{\{\widetilde{K}>X_{\varphi(T)}\}}\middle|\mathscr{F}_t\right] \\ &= \int_0^{\widetilde{K}} X^{-\left(\frac{\delta}{2}-1\right)}p_{\delta>2}(X,\varphi(T);X_{\varphi(t)})\,dX, \end{aligned}$$

with the transition density, $p_{\delta>2}$, given by (B.8) in Appendix B. Now, the symmetry relationship, (B.11) in Appendix B, can be applied to yield

$$\int_0^{\widetilde{K}} X^{-\left(\frac{\delta}{2}-1\right)}p_{\delta>2}(X,\varphi(T);X_{\varphi(t)})\,dX = X_{\varphi(t)}^{-\left(\frac{\delta}{2}-1\right)}\int_0^{\widetilde{K}} p_{4-\delta}(X,\varphi(T);X_{\varphi(t)})\,dX,$$

where the final density to be integrated is the norm-decreasing density given by (B.2). The bounds can be rewritten as

$$\begin{aligned} X_{\varphi(t)}^{-\left(\frac{\delta}{2}-1\right)}&\int_0^{\widetilde{K}} p_{4-\delta}(X,\varphi(T);X_{\varphi(t)})\,dX = \\ &\frac{1}{\bar{S}_t^*}\left[\int_0^{\infty} p_{4-\delta}(X,\varphi(T);X_{\varphi(t)})\,dX - \int_{\widetilde{K}}^{\infty} p_{4-\delta}(X,\varphi(T);X_{\varphi(t)})\,dX\right]. \end{aligned}$$

Using (B.4) and (B.5) from the Appendix yields

$$\mathbb{E}\left[\frac{1}{\bar{S}_t^*}\mathbb{I}_{\{\frac{K}{\beta(T)}>\bar{S}_T^*\}}\middle|\mathscr{F}_t\right] = \frac{1}{\bar{S}_t^*}\left[\chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi};\delta-2,0\right) - \chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi};\delta-2,\frac{\widetilde{K}}{\Delta\varphi}\right)\right].$$

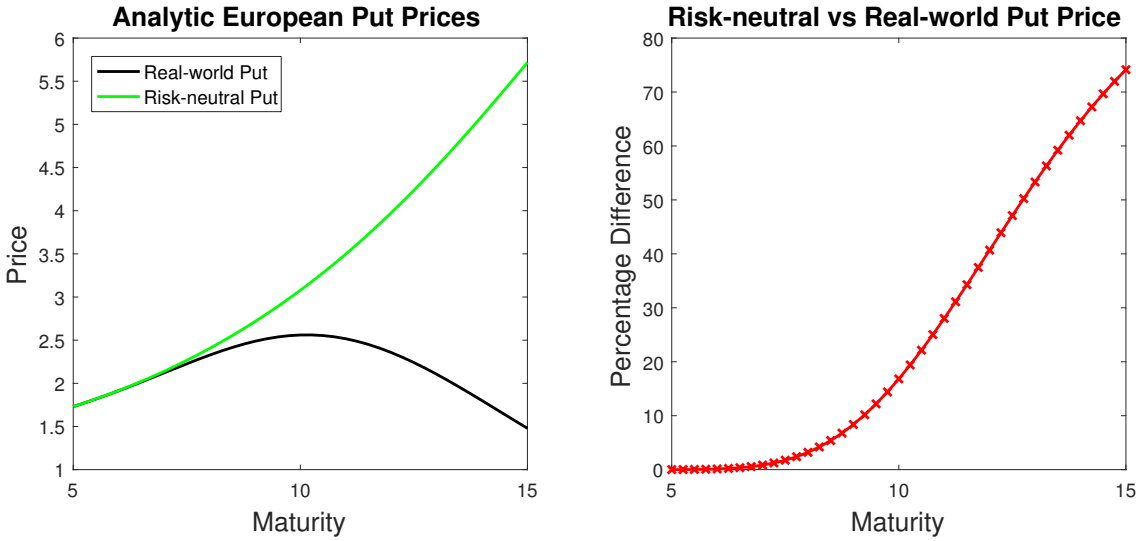

Figure 6.1: Comparison of risk-neutral and real-world prices obtained for at-the-money European put options with maturities out to 15 years.

Similarly, the second expectation can be computed directly from the transition density (B.8),

$$\begin{aligned}\mathbb{E}\left[\mathbb{I}_{\left\{\frac{K}{\beta(T)}>\bar{S}_T^*\right\}}\middle|\mathscr{F}_t\right] &= \int_0^{\widetilde{K}} p_{\delta>2}(X,\varphi(T);X_{\varphi(t)})\,dX \\ &= \chi'^2\left(\frac{\widetilde{K}}{\Delta\varphi};\delta,\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi}\right).\end{aligned}$$

□

The analytic expression for the European call option can be derived in the same way, but the derivation avoids the norm-decreasing density required above.

Proposition 6.3.2. *Assuming a constant interest rate r, the real-world price, $c_{T,K}(t,S_t^*)$, of a European call option on the GOP at time t with expiry T and strike K is given by*

$$c_{T,K}(t,S_t^*) = \bar{S}_t^*\beta(t)\left[1-\chi'^2\left(\frac{\widetilde{K}}{\Delta\varphi};\delta,\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi}\right)\right] - K\frac{\beta(t)}{\beta(T)}\chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi};\delta-2,\frac{\widetilde{K}}{\Delta\varphi}\right),$$

with all definitions as in Proposition 6.3.1.

For comparison, assume a hypothetical risk-neutral measure, $\mathbb{P}_\theta$, with the discounted GOP dynamics under this measure given by (6.10). Although this measure is not equivalent to $\mathbb{P}$, when the strike $K>0$, the risk-neutral call price, denoted $c_{T,K}^{\mathrm{RN}}$, corresponds to the real-world call price given above for $\delta_\theta = \frac{1-2a}{1-a}$. This occurs because the measures differ only for values of S^* around 0, which the integral in the call pricing problem avoids for positive strikes. However, the risk-neutral put option, denoted $p_{T,K}^{\mathrm{RN}}$, is significantly more expensive than the real-world put option for long time horizons.

To see this, consider the mathematical basis for put-call parity,

$$(K-S_T^*)^+ = (S_T^*-K)^+ - S_T^* + K. \tag{6.11}$$

Taking the expectation under the real-world measure, with the GOP as the numeraire, provides

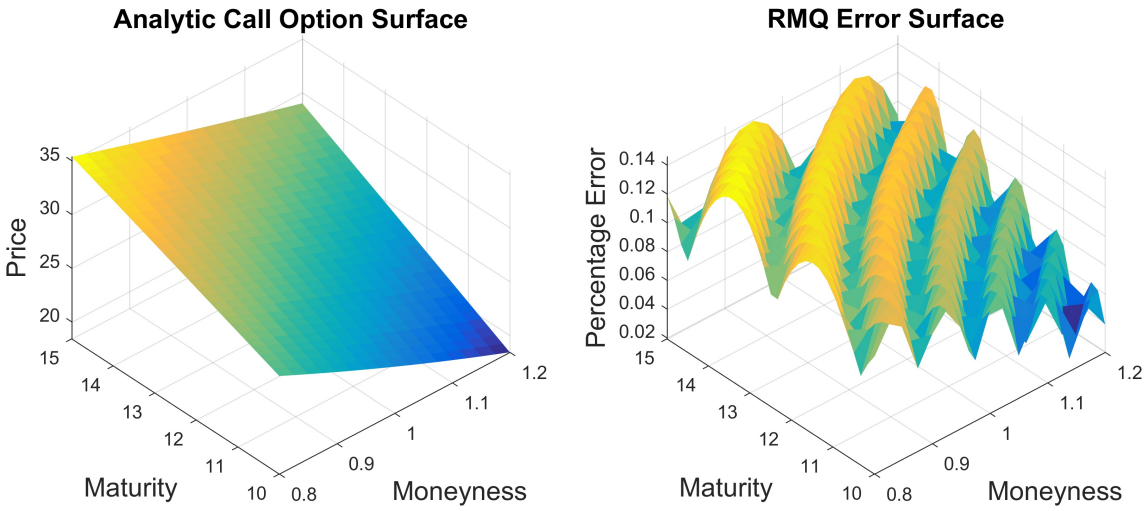

Figure 6.2: European call option price surface with RMQ pricing error.

the *fair* put-call parity relationship,

$$p_{T,K}(t, S_t^*) = c_{T,K}(t, S_t^*) - S_t^* + KP_T(t, S_t^*),$$

where the fair or real-world zero coupon bond price is given by

$$P_T(t, S_t^*) = \mathbb{E}\left[\frac{S_t^*}{S_T^*}\middle|\mathscr{F}_t\right] = \frac{\beta(t)}{\beta(T)}\mathbb{E}\left[\frac{\bar{S}_t^*}{\bar{S}_T^*}\middle|\mathscr{F}_t\right] = \frac{\beta(t)}{\beta(T)}\chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi}; \delta - 2, 0\right).$$

Of course, taking the discounted risk-neutral expectation of (6.11) provides the classical put-call parity relationship,

$$p_{T,K}^{\mathrm{RN}}(t, S_t^*) = c_{T,K}^{\mathrm{RN}}(t, S_t^*) - S_t^* + K\frac{\beta(t)}{\beta(T)}.$$

Since the hypothetical risk-neutral call prices and real-world call prices coincide, and

$$P_T(t, S_t^*) \leq \frac{\beta(t)}{\beta(T)},$$

the real-world put option prices must be less than or equal to the risk-neutral put option prices.

Figure 6.1 illustrates the difference between long-dated at-the-money put options priced using the classical risk-neutral pricing theory and the prices provided by Proposition 6.3.1, obtained using real-world pricing. The parameters used are taken from Baldeaux et al. [2015], where they were estimated from empirical data, with $\alpha_0 = 51.34$, $\eta = 0.1239$, $c = 0.1010$ and $a = 0.2868$. The initial discounted GOP was set at 50 and the constant short rate at 5%. Maturities are set at bi-monthly intervals from 5 out to 15 years.

The left panel of Figure 6.1 shows how the prices correspond for short maturities, with the risk-neutral put becoming more and more expensive as the maturities lengthen. The right panel of Figure 6.1 shows the difference between the risk-neutral put and the real-world put as a percentage of the classical risk-neutral option price. At a maturity of 15 years, the real-world put option is 70% less expensive to purchase.

As a first example of RMQ, Figure 6.2 shows an analytic European call option pricing surface along with the RMQ pricing error. Moneyness is varied by changing the strike over the initial GOP value, and maturities are set at monthly intervals from 10 to 15 years.

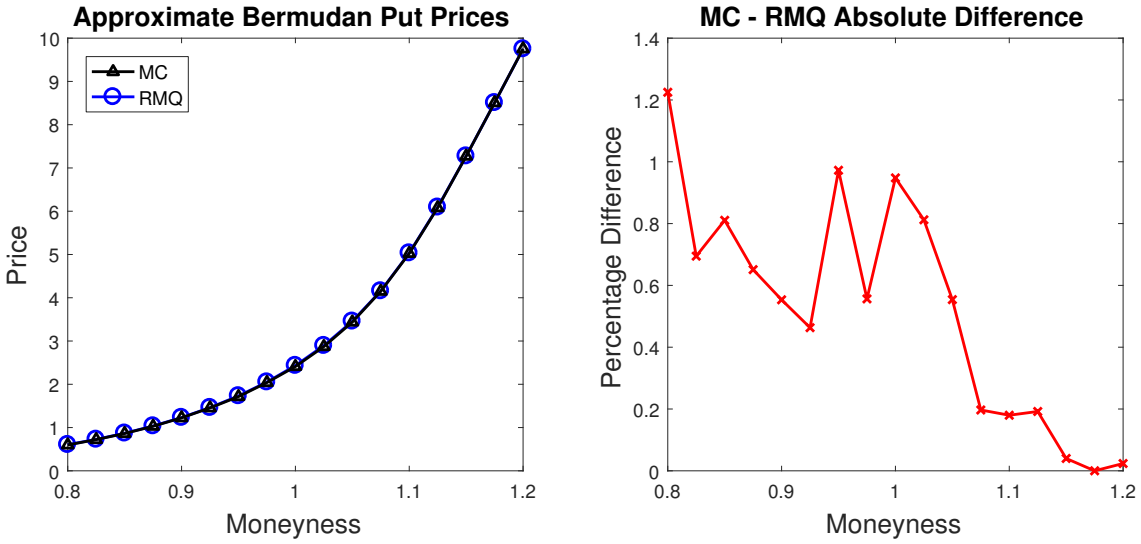

Figure 6.3: Bermudan put options priced using least-squares Monte Carlo and RMQ.

The weak-order 2.0 RMQ scheme was used, see Section 4.1.2, with 12 time-steps per year and 50 codewords held constant across time. The final error is under 0.15% irrespective of maturity and moneyness. The error oscillates across moneyness, as is expected for a tree-type method. Computing the RMQ grid out to 15 years takes less than 1 second.

In Figure 6.3, Bermudan put options on the GOP with a maturity of 5 years and monthly exercise opportunities are priced using both a least-squares Monte Carlo simulation and RMQ. The Monte Carlo simulation is a 500 000 path long-step simulation, using the exact transition density of the discounted GOP. The RMQ algorithm is again the weak-order 2.0 scheme with 12 time-steps and 100 codewords. The Monte Carlo algorithm takes approximately 14.9 seconds per strike, whereas the RMQ algorithm takes only 0.5 seconds to price all strikes. The maximum difference between the two methods is 1.4% of the price in the worst case. Thus, the methods agree very well across strikes with the RMQ algorithm being significantly faster. Note that the RMQ results may well be more accurate than the least-squares Monte Carlo results for low moneyness, as the Monte Carlo simulation may be unreliable for deep out-the-money options.

6.3.2 Stochastic Short Rates

In this subsection, the hybrid model, investigated by Baldeaux et al. [2015], is extended by relaxing the assumption of independence between the short rate and the discounted GOP. It is shown how the real-world zero-coupon bond prices become significantly less than risk-neutral prices as maturities increase. Fast and accurate numerical pricing for European put options on zero-coupon bonds is also provided.

Baldeaux et al. [2015] undertake an empirical investigation to determine which short-rate model, combined with the TCEV model for the discounted GOP, performs best when pricing and hedging long-dated zero-coupon bonds. Their investigation concluded that the 3/2 short-rate model of Ahn and Gao [1999] outperforms the competing models in terms of capturing the dynamics of the real-world short-term interest rate, as well as delivering the smallest prices for zero-coupon bonds.

Under the real-world measure, the 3/2 short-rate model is described by

$$dr_t = \kappa(\theta r_t - r_t^2)\,dt + \sigma r_t^{3/2}\,dW_t^r, \tag{6.12}$$

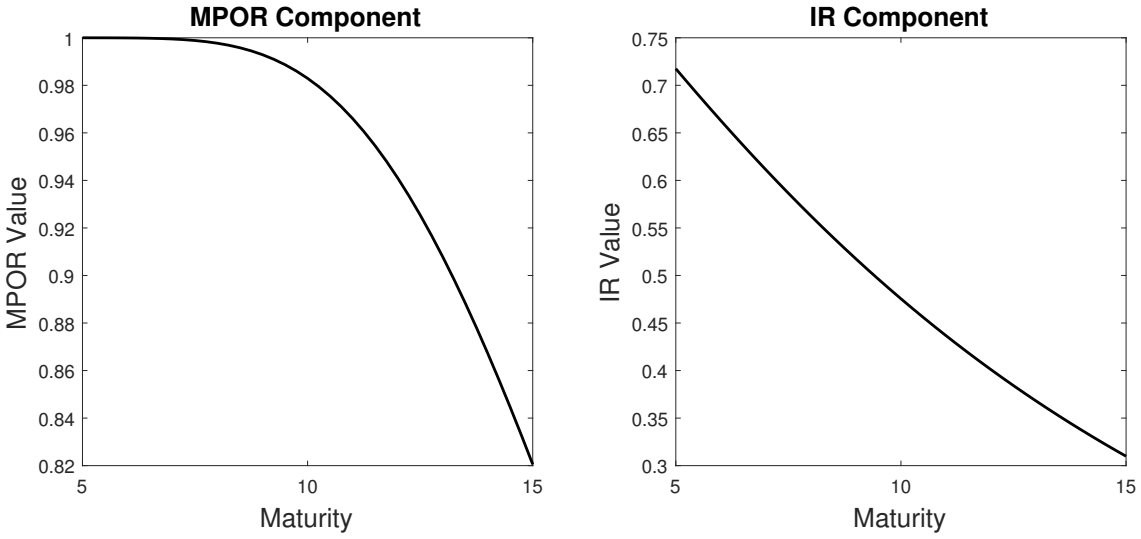

Figure 6.4: The analytic MPOR and IR components of the fair zero-coupon bond price under the hybrid model.

with $r_0 > 0$. Proposition 6.3.3 below is reproduced from Baldeaux et al. [2015].

Proposition 6.3.3. *If the Brownian motion, W^r, driving the short rate is independent of the Brownian motion, W, driving the GOP then the time-t price of a fair zero-coupon bond maturing at T is given by*

$$P_T(t, r_t, \bar{S}_t^*) = \mathbb{E}\left[\frac{S_t^*}{S_T^*}\middle|\mathscr{F}_t\right] = \mathbb{E}\left[\frac{\bar{S}_t^*}{\bar{S}_T^*}\frac{S_t^0}{S_T^0}\middle|\mathscr{F}_t\right] = M(\bar{S}_t^*, t, T)G(r_t, t, T), \tag{6.13}$$

with

$$M(\bar{S}_t^*, t, T) = \mathbb{E}\left[\frac{\bar{S}_t^*}{\bar{S}_T^*}\middle|\mathscr{F}_t\right] = \chi'^2\left(\frac{(\bar{S}_t^*)^{2(1-a)}}{\Delta\varphi}; \delta - 2, 0\right) \tag{6.14}$$

denoting the market price of risk component and

$$G(r_t, t, T) = \mathbb{E}\left[\frac{S_t^0}{S_T^0}\middle|\mathscr{F}_t\right] = \mathbb{E}\left[\exp\left(\int_t^T r_s\, ds\right)\middle|\mathscr{F}_t\right] \tag{6.15}$$

denoting the interest-rate (IR) component.

Proof. The independence of the GOP from the short rate, and thus the savings account, allows the expectation in (6.13) to be separated into the product of (6.14) and (6.15). The right-hand side of (6.14) follows directly from the known transition density of the discounted GOP, provided by Proposition 6.2.2, and the right-hand side of (6.15) follows from the definition of the savings account, (6.1). □

As a result of the above proposition, if the expectation in (6.15) was taken under the risk-neutral measure, the fair zero-coupon bond price could be interpreted as the product of the traditional risk-neutral bond price, $G(r_t, t, T)$, and the market price of risk component, $M(\bar{S}_t^*, t, T)$. Note that the MPOR component is given explicitly as the probability that all paths of the inverse discounted GOP process have not attained zero by time T. This probability goes to 0 as T goes to infinity; eventually the growth-optimal portfolio dominates any other traded asset, ensuring that the expected value of the asset, expressed in terms of the GOP, goes to zero.

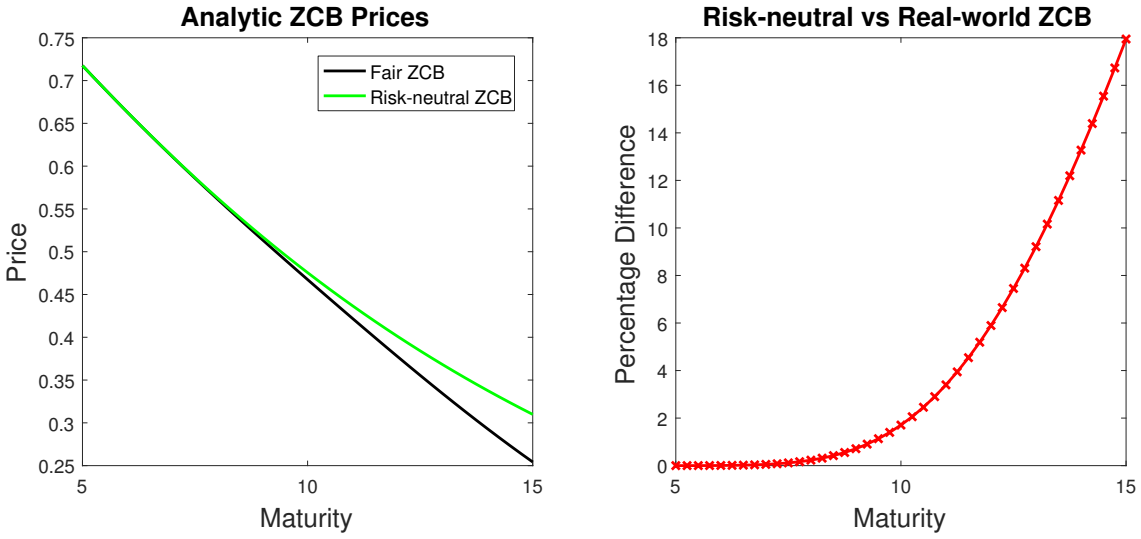

Figure 6.5: The analytic fair zero-coupon bond price in the hybrid model compared to the hypothetical risk-neutral bond price for maturities ranging from 5 to 15 years.

This behaviour can be seen in the left panel of Figure 6.4, where the MPOR component is plotted out to 15 years using the model parameters in the previous section. Note that the MPOR component only becomes significantly less than one well after the 5-year mark. This indicates that theoretical real-world bond prices and those calculated using classical risk-neutral pricing would coincide for maturities out to roughly 8 years for these parameters.

Proposition 6.3.4. *Under the* 3/2 *short-rate model,*

$$G(r_t, t, T) = \frac{\Gamma(\alpha - \gamma) x(r_t, t, T)^{\gamma}}{\Gamma(\alpha)} \, {}_1F_1(\gamma, \alpha, -x(r_t, t, T))$$

with

$$\alpha = \frac{2}{\sigma^2}\left(\kappa + (1+\gamma)\sigma^2\right), \qquad \gamma = \frac{1}{\sigma^2}\left(\sqrt{\phi^2 + 2\sigma^2} - \phi\right)$$
$$x(r, t, T) = \frac{2\kappa\theta}{\sigma^2\left(e^{\kappa\theta(T-t)} - 1\right) r}, \qquad \phi = \kappa + \frac{1}{2}\sigma^2,$$

and where ${}_1F_1$ *is the confluent hypergeometric function of the first kind, or Kummer's function.*

Proof. See Ahn and Gao [1999, Sec. 3]. □

The IR component is illustrated in the right panel of Figure 6.4, using the 3/2 model parameters estimated from the market by Baldeaux et al. [2015], with $\kappa = 3.5726$, $\theta = 0.096$, $\sigma = 0.7960$ and the initial short rate selected as $r_0 = 0.05$.

Finally, the analytic fair zero-coupon bond price is displayed in Figure 6.5 for maturities out to 15 years and contrasted with the risk-neutral bond price. The right panel of Figure 6.5 shows the difference between the hypothetical risk-neutral bond and the real-world bond as a percentage of the classical risk-neutral bond price. At a maturity of 15 years, the fair bond is 18% less expensive to purchase. It is clear that the fair bond will continue to become less expensive, under the 3/2 dynamics, as the maturity lengthens further. It is beyond the scope of this work to demonstrate the hedging of these contracts. Hulley and Platen [2012] have, however, demonstrated that the theoretical real-world bond prices can be accurately hedged.

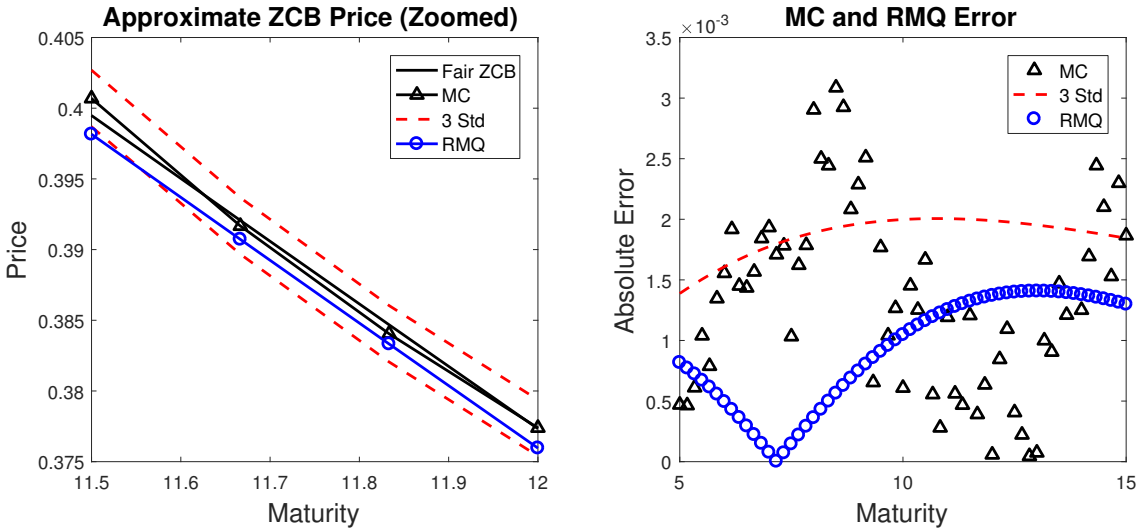

Figure 6.6: Approximating the fair zero-coupon bond price using Monte Carlo simulation and RMQ.

In Figure 6.6 the fair zero coupon bond price is approximated using Monte Carlo simulation and the RMQ algorithm. The Monte Carlo simulation for the IR and MPOR components were each computed using 100 000 paths. The MPOR component could be long-stepped to each maturity considered, as the exact transition density is known, however the 3/2 model was simulated using an Euler-Maruyama scheme with 6 time steps per year. The RMQ algorithm also used 6 time steps per year with 50 codewords for the short-rate process and 150 codewords for the discounted GOP process. The Monte Carlo simulation took 13.5 seconds to compute, whereas the RMQ algorithm was more than twice as fast at 5.1 seconds.

The absolute error for both methods is small; the left panel of Figure 6.6 has been zoomed in to focus on a 6-month period so that the difference between the approximations and the analytic value can be seen. The Monte Carlo method presents some bias over the full period, as more points lie outside the three standard deviation bounds than would be expected. The RMQ algorithm lies well within the error bounds for the full range of maturities.

Affect of Short Rate and GOP Correlation

Although the assumption of independence between the short rate and GOP seems restrictive, some empirical evidence is provided for it by Baldeaux et al. [2015]. They use the daily 3-month USD T-Bill rates as the proposed short rate and the EWI114 equi-weighted index to approximate the GOP. They find that the covariation remains close to zero and exhibits no clear trend. The theory of approximating the GOP using a well-diversified world-index is presented by Platen and Rendek [2012]. It is now demonstrated that the RMQ methodology can also efficiently handle the case of correlation between the short-rate and the GOP.

To account for correlation, the Joint RMQ algorithm, developed in Chapter 5, has been used. In the left panel of Figure 6.7 the zero-coupon bond price under the hybrid model is displayed for a range of correlation values between -0.9 and 0.9. The large range is chosen to exaggerate the effect. The impact of positive correlation is greater than that of negative correlation, but the overall impact of the correlation is small. Again, the figure has been zoomed in to focus on a 6-month period. In the right panel of Figure 6.7 the percentage difference between the zero-coupon bond price and the price with zero correlation is displayed for different correlations and maturities

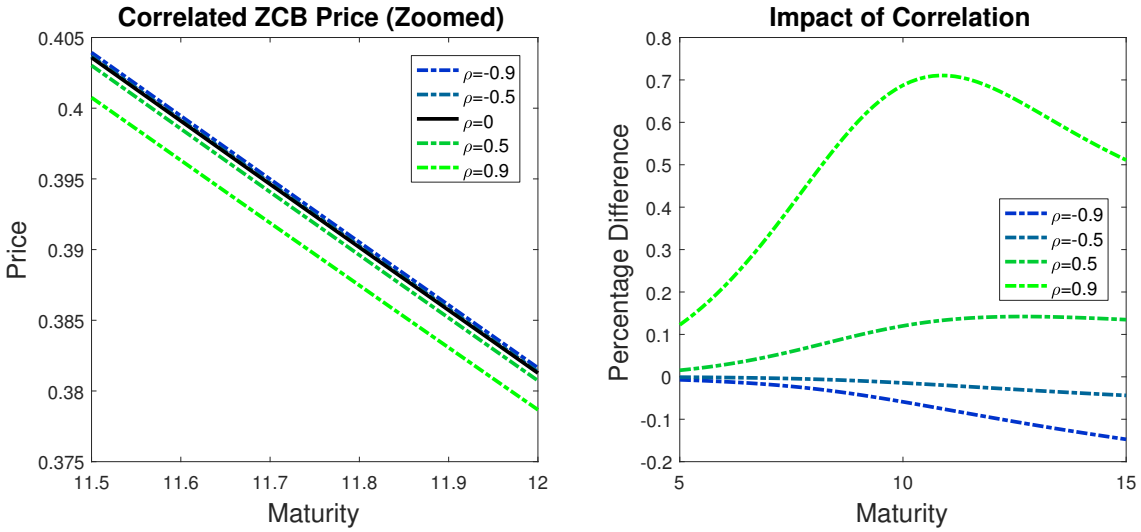

Figure 6.7: The fair zero-coupon bond price in the hybrid model for a range of correlation values.

out to 15 years.

These numerical results substantiate the original finding: even for large values of the correlation the impact on the zero-coupon bond price is less than 0.8%. Thus, for bond-pricing applications, correlation between the GOP and the short rate may be safely neglected. Intuitively, the correlation can be viewed as affecting the path of the MPOR process while leaving the path of the IR process unchanged, for example, when running an Euler Monte Carlo simulation with the covariance matrix decomposed using the Cholesky decomposition. However, the path of the MPOR process only has a minimal effect on the zero-coupon bond price for shorter maturities, as seen in the left panel of Figure 6.6. Thus, changing the correlation only has a small effect on the final zero-coupon bond price.

Zero-coupon Bond Options

To price a European put option on a zero-coupon bond, denoted ZCP, under the hybrid model with the option maturing at T and the bond maturing at $S > T$, the following expectation must be computed

$$\mathrm{ZCP}_{T,S,K}(t, r_t, \bar{S}_t^*) := \mathbb{E}\left[\frac{S_t^*}{S_T^*}\left(K - P_S(T, r_T, \bar{S}_T^*)\right)^+ \middle| \mathscr{F}_t\right].$$

In Figure 6.8, the prices obtained using Monte Carlo simulation and the RMQ algorithm for the case $T = 10$ years and $S = 15$ years are displayed. The at-the-money strike is taken as the fair forward bond. As before, the Monte Carlo simulation for the IR and MPOR components used 100 000 paths each, with 6 time steps per year for the IR component. The RMQ algorithm used 12 time steps per year with 50 codewords for the short-rate process and 150 codewords for the discounted GOP process. The Monte Carlo simulation took 2.45 seconds per strike, whereas the RMQ algorithm computed the prices for all strikes in 3.6 seconds.

There is no reference price available, but the prices obtained using the RMQ algorithm and Monte Carlo simulation lie sufficiently close together to indicate that RMQ is efficient and accurate. Barring a single, deep in-the-money point, all the prices obtained using RMQ lie within the three standard deviation bounds of the Monte Carlo simulation. The average difference between the prices across all strikes is less than 2%.

It has already been established that, under the dynamics of the hybrid model, real-world zero-

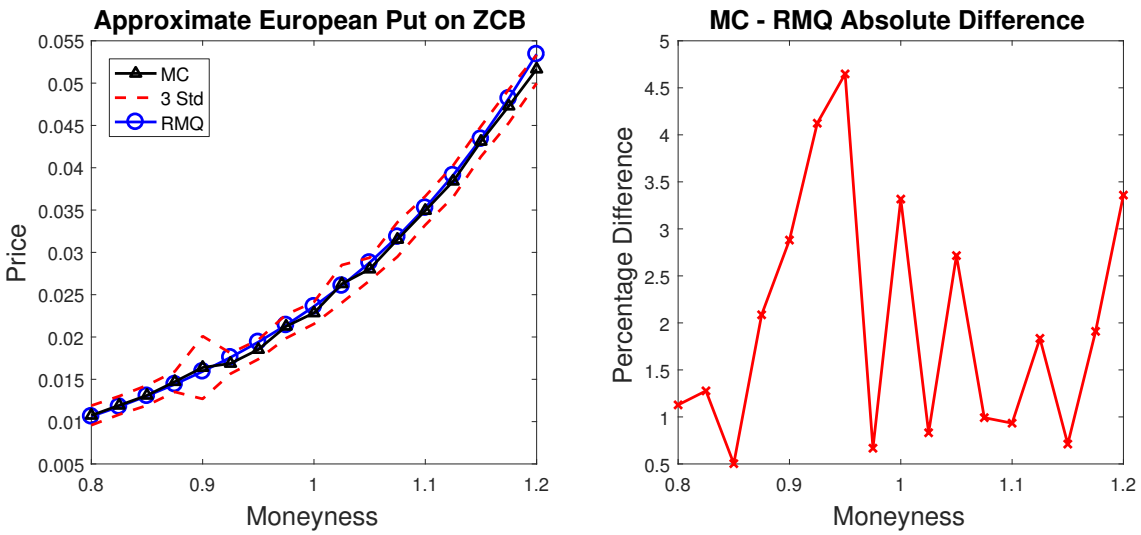

Figure 6.8: Approximating the prices of a European put on a zero-coupon bond using Monte Carlo simulation and RMQ.

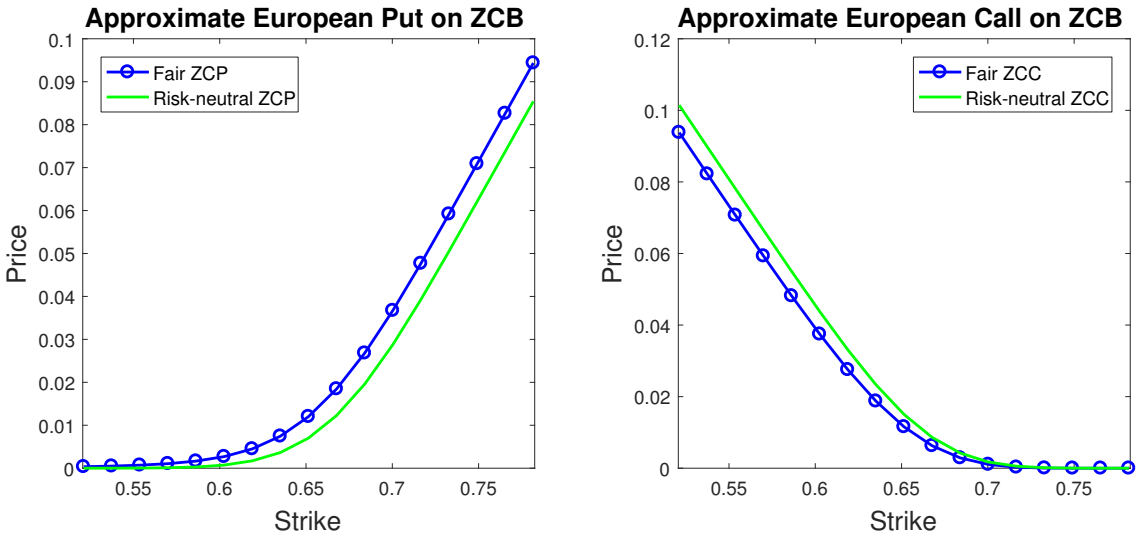

Figure 6.9: Comparison of European put and call options written on zero-coupon bonds using real-world pricing and priced under a hypothetical risk-neutral measure.

coupon bonds are less expensive than the bonds implied by traditional risk-neutral pricing. If a risk-neutral measure is assumed, this leads to an asymmetry in the prices of vanilla options on zero-coupon bonds. Real-world put options on fair bonds are more expensive than risk-neutral put options on risk-neutral bonds. The reverse is, of course, true for call options. This behaviour is illustrated in Figure 6.9 for $T = 5$ years and $S = 10$ years. For the example depicted, the fair forward bond was computed using real-world pricing and the same strikes were used for both the real-world and risk-neutral options.

Chapter 7

Conclusion

This thesis extended the recursive marginal quantization algorithm to higher-order discretization schemes, introduced a method for modelling the correct zero boundary behaviour and derived a new and efficient quantization algorithm for stochastic volatility models. The extensions were applied to several of the popular and challenging models in mathematical finance as well as outside the traditional risk-neutral pricing framework by using the benchmark approach. A chapter-by-chapter review of the primary contributions now follows.

In Chapter 3, the RMQ algorithm was presented in more generality than the original formulation of Pagès and Sagna [2015]. It was also shown how to augment the RMQ algorithm in order to implement absorption or reflection at the zero boundary, thus ensuring non-negativity of solutions. This allows RMQ to be applied in important cases, where the algorithm may previously have failed. A concise matrix formulation was provided such that the algorithm can be easily implemented.

In Chapter 4, the recursive marginal quantization methodology was extended from the standard Euler-Maruyama scheme to higher-order numerical schemes, specifically the Milstein scheme and the weak-order 2.0 Taylor scheme of Kloeden and Platen [1999]. This entailed introducing noncentral chi-squared updates and utilizing the more general form of the algorithm presented in the preceding chapter. In the case of the simplified weak-order 2.0 scheme, numerical evidence of the improved convergence was provided. For the Milstein scheme, a theoretical error bound was derived, based on the original theorem of Pagès and Sagna [2015], which can easily be extended to the weak-order 2.0 scheme.

All the schemes were used successfully to price European, Bermudan and discrete barrier options. Furthermore, the improved computational efficiency of the new schemes was demonstrated for option pricing. The pricing results show that once RMQ is implemented with second weak-order updates, it provides extremely accurate contingent claim pricing, well-suited for the fast calibration of entire derivative books.

As a proof-of-concept example, the weak-order 2.0 RMQ algorithm was used to calibrate the CEV model directly to American put option data on a single day.

In Chapter 5, the joint recursive marginal quantization algorithm for stochastic volatility models was derived, which provides a significant computational advantage over the most recent quantization developments in this area.

The central idea was to margin over, and effectively undo, the Cholesky decomposition in the two-dimensional Euler scheme when performing the quantization. It was shown how the joint probabilities could be computed exactly and using a computationally efficient approximation.

A concise matrix formulation was provided for efficient implementation. The robustness of the algorithm was demonstrated by pricing options with path dependency, early exercise and exotic

features. Parameter sets that would be appropriate to interest rate and equity environments were used to demonstrate the correct handling of the boundary behaviour.

JRMQ was shown to be accurate and fast when compared to traditional Monte Carlo methods. This allows the calibration of large derivative books, as per Callegaro et al. [2015], to be extended from considering only local volatility models to the more flexible stochastic volatility models, while retaining the efficiency of the underlying Newton-Raphson technique.

Similarly to Chapter 4, the JRMQ algorithm was used to calibrate the Heston model directly to American put option data on a single day.

In Chapter 6, the benchmark approach was reviewed and a derivation of the real-world pricing theorem was presented for a two-asset continuous market. It was demonstrated that real-world pricing may produce significantly lower prices for long-dated bonds and vanilla options than the classical risk-neutral pricing approach. The time-dependent constant elasticity of variance model was used to model the growth-optimal portfolio. Under this model and the assumption of constant interest rates, analytic European option pricing formulae were derived in detail, extending the results of Miller and Platen [2008, 2010], for the modified constant elasticity of variance model and the stylized minimal market model. Recursive marginal quantization was used to efficiently and accurately produce long-dated European option pricing surfaces as well as price Bermudan options on the growth-optimal portfolio.

The hybrid model of Baldeaux et al. [2015] was constructed by combining the TCEV model, for the growth-optimal portfolio, with the 3/2 short-rate model of Ahn and Gao [1999]. Under this combined model, RMQ was used to efficiently price long-dated zero-coupon bonds and options on zero-coupon bonds. The effect of introducing correlation between the growth-optimal portfolio and the stochastic short rate was investigated using the JRMQ algorithm, where it was shown that the correlation only has a minor impact.

This final chapter applied the RMQ algorithm outside the traditional risk-neutral framework by highlighting its effectiveness as a pricing mechanism for long-dated contracts under the benchmark approach.

Appendix A

Proofs

A.1 Vector Quantization

Proof of Proposition 2.2.1. As f is Lipschitz continuous,

$$|f(X) - f(\widehat{X})| \leq [f]_{\mathrm{Lip}} |X - \widehat{X}|.$$

Taking the expectation conditional on $\widehat{X}$ yields

$$\begin{aligned} &\mathbb{E}\Big[|f(X) - f(\widehat{X})||\widehat{X}\Big] \leq [f]_{\mathrm{Lip}}\, \mathbb{E}\Big[|X - \widehat{X}||\widehat{X}\Big] \\ \implies &\Big|\mathbb{E}\Big[f(X)|\widehat{X}\Big] - f(\widehat{X})\Big| \leq [f]_{\mathrm{Lip}}\, \mathbb{E}\Big[|X - \widehat{X}||\widehat{X}\Big], \end{aligned}$$

as $|\mathbb{E}[\cdot]| \leq \mathbb{E}[|\cdot|]$ in general and $f(\widehat{X})$ is $\sigma(\widehat{X})$-measurable. Now for any real number $r \geq 1$,

$$\begin{aligned} &\Big|\mathbb{E}\Big[f(X)|\widehat{X}\Big] - f(\widehat{X})\Big|^r \leq [f]^r_{\mathrm{Lip}}\, \Big[\mathbb{E}\Big[|X - \widehat{X}||\widehat{X}\Big]\Big]^r \\ \implies &\Big|\mathbb{E}\Big[f(X)|\widehat{X}\Big] - f(\widehat{X})\Big|^r \leq [f]^r_{\mathrm{Lip}}\, \mathbb{E}\Big[|X - \widehat{X}|^r|\widehat{X}\Big], \end{aligned}$$

by the conditional Jensen's inequality. Taking the expectation and raising to the power $\frac{1}{r}$ yields

$$\Big\|\mathbb{E}\Big[f(X)|\widehat{X}\Big] - f(\widehat{X})\Big\|_r \leq [f]_{\mathrm{Lip}} \|X - \widehat{X}\|_r.$$

In the special case of $r = 1$, the left-hand side becomes

$$\mathbb{E}\Big[\Big|\mathbb{E}\Big[f(X)|\widehat{X}\Big] - f(\widehat{X})\Big|\Big] \geq \Big|\mathbb{E}[f(X)] - \mathbb{E}[f(\widehat{X})]\Big|$$

such that

$$|\mathbb{E}[f(X)] - \mathbb{E}[f(\widehat{X})]| \leq [f]_{\mathrm{Lip}} \|X - \widehat{X}\|_1.$$

Since $\mathbb{P}$ is a probability measure, Jensen's inequality provides the monotonicity of the $\mathcal{L}^p(\mathbb{P})$-norm such that $\|f\|_r \leq \|f\|_p$ for $r \leq p < \infty$ and all f. Thus,

$$\begin{aligned} |\mathbb{E}[f(X)] - \mathbb{E}[f(\widehat{X})]| &\leq [f]_{\mathrm{Lip}} \|X - \widehat{X}\|_1 \\ &\leq [f]_{\mathrm{Lip}} \|X - \widehat{X}\|_2 \\ &= [f]_{\mathrm{Lip}} \sqrt{D(\Gamma)}. \end{aligned}$$

□

Proof of Proposition 2.2.2. Consider the inequality provided by the Taylor expansion of f and the Lipschitz continuity of f',

$$|f(X) - f(\widehat{X}) - f'(\widehat{X})(X - \widehat{X})| \leq [f']_{\text{Lip}}|X - \widehat{X}|^2.$$

Taking the expectation conditional on $\widehat{X}$ yields

$$\begin{aligned} &\mathbb{E}\Big[|f(X) - f(\widehat{X}) - f'(\widehat{X})(X - \widehat{X})|\big|\widehat{X}\Big] \leq [f']_{\text{Lip}}\mathbb{E}\Big[|X - \widehat{X}|^2\big|\widehat{X}\Big] \\ \implies &\Big|\mathbb{E}[f(X)|\widehat{X}] - f(\widehat{X}) - \mathbb{E}[f'(\widehat{X})(X - \widehat{X})|\widehat{X}]\Big| \leq [f']_{\text{Lip}}\mathbb{E}\Big[|X - \widehat{X}|^2\big|\widehat{X}\Big]. \end{aligned}$$

Note that

$$\mathbb{E}[f'(\widehat{X})(X - \widehat{X})|\widehat{X}] = f'(\widehat{X})\mathbb{E}[(X - \widehat{X})|\widehat{X}] = 0,$$

since $f'(\widehat{X})$ is $\sigma(\widehat{X})$-measurable and $\mathbb{E}[X|\widehat{X}] = \widehat{X}$ because Γ is a self-consistent quantizer. Then for any real number $r \geq 1$, and using Jensen's conditional inequality,

$$|\mathbb{E}[f(X)|\widehat{X}] - f(\widehat{X})|^r \leq [f']^r_{\text{Lip}}\mathbb{E}\Big[|X - \widehat{X}|^{2r}\big|\widehat{X}\Big].$$

Taking the expectation and raising to the power $\frac{1}{r}$ yields

$$\|\mathbb{E}[f(X)|\widehat{X}] - f(\widehat{X})\|_r \leq [f']_{\text{Lip}}\|X - \widehat{X}\|_r^2.$$

For the special case when $r = 1$

$$\begin{aligned} \Big|\mathbb{E}[f(X)|\widehat{X}] - f(\widehat{X})\Big| &\leq [f']_{\text{Lip}}\|X - \widehat{X}\|_1^2, \\ &\leq [f']_{\text{Lip}}\|X - \widehat{X}\|_2^2 \\ &= [f']_{\text{Lip}}D(\Gamma). \end{aligned}$$

□

A.2 Recursive Marginal Quantization of Higher-order Schemes

Since the size of the time-step is held constant, $\Delta := \Delta t$ is adopted for notational brevity.

Proof of Lemma 4.2.1. To prove Lemma 4.2.1, a technical lemma is needed. It is adapted from the Appendix of Pagès and Sagna [2015], where a proof appears.

Lemma A.2.1. *Let $a \in \mathbb{R}$ and$p \in [2, 3]$. Then, for all $u \in \mathbb{R}$,*

$$|a + u|^p \leq |a|^p + p|a|^{p-2}(au) + \frac{p(p-1)}{2}\left(|a|^{p-2}|u|^2 + |u|^p\right). \tag{A.1}$$

The proof proceeds in three steps. First, it is shown that for any random variable $Z \in \mathcal{L}^p(\Omega, \mathscr{F}, \mathbb{P})$ with $\mathbb{E}[Z] = 0$,

$$\mathbb{E}[|c + \Delta m Z|^p] \leq \left(1 + \frac{(p-1)(p-2)}{2}\Delta^2\right)|c|^p + \Delta^2(p - 1 + \Delta^{p-2})\mathbb{E}[|Z|^p],$$

with $c,\ m \in \mathbb{R}$ and $\Delta > 0$.

From (A.1),

$$|c + \Delta mZ|^p \leq |c|^p + p\Delta|c|^{p-2}(cmZ) + \tfrac{1}{2}p(p-1)\left(\Delta^2|c|^{p-2}|mZ|^2 + \Delta^p|mZ|^p\right).$$

Applying Young's inequality with the conjugate exponents $p' = \frac{p}{p-2}$ and $q' = \frac{p}{2}$ yields

$$\Delta^2|c|^{p-2}|mZ|^2 \leq \Delta^2\left(\frac{|c|^p}{p'} + \frac{|mZ|^p}{q'}\right),$$

such that

$$\begin{aligned}|c + \Delta mZ|^p &\leq |c|^p + p\Delta|c|^{p-2}(cmZ) + \tfrac{1}{2}p(p-1)\left(\tfrac{\Delta^2}{p'}|c|^p + \left(\tfrac{\Delta^2}{q'} + \Delta^p\right)|mZ|^p\right)\\ &= |c|^p\left(1 + \tfrac{p(p-1)}{2p'}\Delta^2\right) + p\Delta|c|^{p-2}(cmZ) + \Delta^2\left(\tfrac{p(p-1)}{2q'} + \Delta^{p-2}\right)|mZ|^p.\end{aligned}$$

Now taking the expectation, using that $\mathbb{E}[Z] = 0$ and substituting in the values of p' and q' yields the desired result for step one,

$$\mathbb{E}[|c + \Delta mZ|^p] \leq \left(1 + \frac{(p-1)(p-2)}{2}\Delta^2\right)|c|^p + \Delta^2(p - 1 + \Delta^{p-2})\mathbb{E}[|Z|^p]. \tag{A.2}$$

Step two generalizes the above bound by making c and m the appropriate functions of x for the Milstein update. Let

$$c := x + \Delta a(x) \qquad \text{and} \qquad m := \tfrac{1}{2}b(x)b'(x).$$

Then, from the global Lipschitz continuity assumptions and the definition of L,

$$|c| \leq |x|(1 + L\Delta) + L\Delta \qquad \text{and} \qquad |m|^p \leq 2^pL^p(1 + |x|^p).$$

Using $L > 0$ allows the bound for the p-th power of the norm of c to be refined,

$$\begin{aligned}|c|^p &\leq (1 + 2L\Delta)^p\left(\frac{1 + L\Delta}{1 + 2L\Delta}|x|\frac{L\Delta}{1 + 2L\Delta}\right)^p\\ &\leq (1 + 2L\Delta)^p\left(\frac{1 + L\Delta}{1 + 2L\Delta}|x|^p\frac{L\Delta}{1 + 2L\Delta}\right)\\ &\leq (1 + 2L\Delta)^p|x|^p + (1 + 2L\Delta)^{p-1}L\Delta.\end{aligned}$$

Substituting these bounds into (A.2) yields

$$\begin{aligned}\mathbb{E}[|c + \Delta mZ|^p] \leq \left(1 + \tfrac{(p-1)(p-2)}{2}\Delta^2\right)(1 + 2L\Delta)^p|x|^p + \left(1 + \tfrac{(p-1)(p-2)}{2}\Delta^2\right)(1 + 2L\Delta)^{p-1}L\Delta\\ +\Delta^2 2^pL^p(p - 1 + \Delta^{p-2})(1 + |x|^p)\mathbb{E}[|Z|^p].\end{aligned}$$

Gathering the co-efficients of $|x|^p$ and Δ provides

$$\begin{aligned}\mathbb{E}[|c + \Delta mZ|^p] \leq \left[\left(1 + \tfrac{(p-1)(p-2)}{2}\Delta^2\right)(1 + 2L\Delta)^p + \Delta^2 2^pL^p(p - 1 + \Delta^{p-2})\mathbb{E}[|Z|^p]\right]|x|^p\\ + \left[\left(1 + \tfrac{(p-1)(p-2)}{2}\Delta^2\right)(1 + 2L\Delta)^{p-1}L + 2^pL^p(\Delta(p-1) + \Delta^{p-1})\mathbb{E}[|Z|^p]\right]\Delta.\end{aligned}$$

Using the inequality $1 + u \leq e^u$ for every $u \in \mathbb{R}$ yields the desired result for this step,

$$\mathbb{E}[|c + \Delta m Z|^p] \leq (e^{\kappa'_p \Delta} + K'_p \Delta)|x|^p + (e^{\kappa'_p \Delta} L + K'_p)\Delta, \tag{A.3}$$

with

$$\kappa'_p := \frac{(p-1)(p-2)}{2}\Delta + 2pL \qquad \text{and} \qquad K'_p := 2^p L^p (\Delta(p-1) + \Delta^{p-1})\mathbb{E}[|Z|^p].$$

The third and final step relates the above bound to the quantizer at time k. Consider the expectation

$$\mathbb{E}[|\widetilde{X}_k|^p] = \mathbb{E}\Big[\mathbb{E}[|\mathcal{U}(\widehat{X}_{k-1}, Z_k)|^p \big| \widehat{X}_{k-1}]\Big].$$

The Milstein update can be re-written in the required form as

$$\mathcal{U}(x, Z_k) = \Delta m(x)\overline{Z}_k + c(x),$$

with

$$\overline{Z}_k := Z_k + (1 - \lambda(x)) \qquad \text{and} \qquad \lambda(x) := (\sqrt{\Delta} b'(x))^{-2},$$

where Z_k is a noncentral chi-squared random variable with one degree of freedom and noncentrality, λ, which depends on x. Note that $\mathbb{E}[\overline{Z}_k] = 0$, as required. The distribution of $\overline{Z}_k$ depends on x, but its mean does not. Thus, A.3 can be used on the inner expectation,

$$\begin{aligned}\mathbb{E}[|\widetilde{X}_k|^p] &= \mathbb{E}\Big[\mathbb{E}\Big[|c(\widehat{X}_{k-1}) + \Delta m(\widehat{X}_{k-1})\overline{Z}_k|^p \big| \widehat{X}_{k-1}\Big]\Big] \\ &\leq (e^{\kappa'_p \Delta} + K'_p \Delta)\mathbb{E}\Big[|\widehat{X}_{k-1}|^p\Big] + (e^{\kappa'_p \Delta} L + K'_p)\Delta.\end{aligned}$$

Note that

$$\mathbb{E}\Big[|\widehat{X}_{k-1}|^p\Big] = \mathbb{E}\Big[\Big|\mathbb{E}[\widetilde{X}_{k-1}|\widehat{X}_{k-1}]\Big|^p\Big]$$

since $\widehat{X}_{k-1}$ is a self-consistent quantizer for $\widetilde{X}_{k-1}$ by construction. Jensen's inequality provides

$$\begin{aligned}\mathbb{E}\Big[\Big|\mathbb{E}[\widetilde{X}_{k-1}|\widehat{X}_{k-1}]\Big|^p\Big] &\leq \mathbb{E}\Big[\mathbb{E}\Big[|\widetilde{X}_{k-1}|^p|\widehat{X}_{k-1}\Big]\Big] \\ &= \mathbb{E}\Big[|\widetilde{X}_{k-1}|^p\Big].\end{aligned}$$

Thus,

$$\mathbb{E}[|\widetilde{X}_k|^p] \leq (e^{\kappa'_p \Delta} + K'_p \Delta)\mathbb{E}\Big[|\widetilde{X}_{k-1}|^p\Big] + (e^{\kappa'_p \Delta} L + K'_p)\Delta.$$

It is straightforward to show via induction that, for $k = 0, \ldots, n$,

$$\begin{aligned}\mathbb{E}[|\widetilde{X}_k|^p] &\leq (e^{\kappa'_p \Delta} + K'_p \Delta)^k |x_0|^p + (e^{\kappa'_p \Delta} L + K'_p)\Delta \sum_{j=0}^{k-1} (e^{\kappa'_p \Delta} + K'_p \Delta)^j \\ &= e^{\kappa'_p k\Delta}(1 + K'_p \Delta e^{-\kappa'_p \Delta})^k |x_0|^p + (e^{\kappa'_p \Delta} L + K'_p)\Delta \sum_{j=0}^{k-1} e^{\kappa'_p j\Delta}(1 + K'_p \Delta e^{-\kappa'_p \Delta})^j \\ &\leq e^{\kappa'_p k\Delta}(1 + K'_p \Delta)^k |x_0|^p + (e^{\kappa'_p \Delta} L + K'_p)\Delta \sum_{j=0}^{k-1} e^{\kappa'_p j\Delta}(1 + K'_p \Delta)^j,\end{aligned}$$

where the last step follows because $e^{-\kappa'_p\Delta} \leq 1$. Using the inequality $1 + u \leq e^u$, for every $u \in \mathbb{R}$,

$$\begin{aligned}\mathbb{E}[|\widetilde{X}_k|^p] &\leq e^{(\kappa'_p+K'_p)k\Delta}|x_0|^p + (e^{\kappa'_p\Delta}L + K'_p)\Delta\sum_{j=0}^{k-1} e^{(\kappa'_p+K'_p)j\Delta} \\ &= e^{(\kappa'_p+K'_p)t_k}|x_0|^p + (e^{\kappa'_p\Delta}L + K'_p)\Delta\frac{e^{(\kappa'_p+K'_p)t_k} - 1}{e^{(\kappa'_p+K'_p)\Delta} - 1}.\end{aligned}$$

Finally, noting that $e^{(\kappa'_p+K'_p)\Delta} - 1 \geq (\kappa'_p + K'_p)\Delta$ yields the desired result,

$$\mathbb{E}[|\widetilde{X}_k|^p] \leq e^{(\kappa'_p+K'_p)t_k}|x_0|^p + (e^{\kappa'_p\Delta}L + K'_p)\frac{e^{(\kappa'_p+K'_p)t_k} - 1}{\kappa'_p + K'_p}. \tag{A.4}$$

□

Proof of Theorem 4.2.2. From the triangle inequality, for every $k = 0, \ldots, n$,

$$\|\overline{X}_k - \widehat{X}_k\|_2 \leq \|\overline{X}_k - \widetilde{X}_k\|_2 + \|\widetilde{X}_k - \widehat{X}_k\|_2. \tag{A.5}$$

The first step of the proof concentrates on controlling the first term on the right-hand side of the above equation. With this in mind, note that the Milstein update function is Lipschitz continuous with respect to the $\mathcal{L}^2$-norm for every $k = 0, \ldots, n$. In fact, it obeys the same bound as the Euler update,

$$\begin{aligned}\mathbb{E}\Big[|\mathcal{U}(x, Z_k) - \mathcal{U}(x', Z_k)|^2\Big] &= |x - x'|^2 + \Delta^2|a(x) - a(x')|^2 + \Delta|b(x) - b(x')|^2 \\ &\quad + 2\Delta|x - x'||a(x) - a(x')| \\ &\leq \left(1 + \Delta^2[a]^2_{\mathrm{Lip}} + \Delta[b]^2_{\mathrm{Lip}} + 2\Delta[a]_{\mathrm{Lip}}\right)|x - x'|^2 \\ &= \left(1 + \Delta(2[a]_{\mathrm{Lip}} + [b]^2_{\mathrm{Lip}}) + \Delta[a]^2_{\mathrm{Lip}}\right)|x - x'|^2 \\ &\leq (1 + \Delta C_{a,b})^2|x - x'|^2 \\ &\leq e^{2\Delta C_{a,b}}|x - x'|^2,\end{aligned}$$

where $C_{a,b} = [a]_{\mathrm{Lip}} + \frac{1}{2}[b]^2_{\mathrm{Lip}}$ and the last step follows from the inequality $1 + u \leq e^u$ for $u \in \mathbb{R}$. Now for every $i = 0, \ldots, n - 1$,

$$\begin{aligned}\|\overline{X}_{i+1} - \widetilde{X}_{i+1}\|_2 &= \|\mathcal{U}(\overline{X}_i, Z_{i+1}) - \mathcal{U}(\widehat{X}_i, Z_{i+1})\|_2 \\ &\leq e^{\Delta C_{a,b}}\|\overline{X}_i - \widehat{X}_i\|_2 \\ &\leq e^{\Delta C_{a,b}}\left[\|\overline{X}_i - \widetilde{X}_i\|_2 + \|\widetilde{X}_i - \widehat{X}_i\|_2\right].\end{aligned} \tag{A.6}$$

The next step of the proof requires using (A.6) to prove

$$\|\overline{X}_k - \widetilde{X}_k\|_2 \leq \sum_{i=1}^{k-1} e^{(k-i)\Delta C_{a,b}}\|\widetilde{X}_i - \widehat{X}_i\|_2, \tag{A.7}$$

for $k = 1, \ldots, n$, by induction. First note that $\|\overline{X}_1 - \widetilde{X}_1\|_2 = 0$, as $\overline{X}_0 = \widehat{X}_0 = x_0$.

For $k = 2$,

$$\|\overline{X}_2 - \widetilde{X}_2\|_2 \leq e^{\Delta C_{a,b}}\|\widetilde{X}_1 - \widehat{X}_1\|_2,$$

using (A.6). Then for $k = 3$,

$$\begin{aligned}\|\overline{X}_3 - \widetilde{X}_3\|_2 &\leq e^{\Delta C_{a,b}} \left[e^{\Delta C_{a,b}} \|\widetilde{X}_1 - \widehat{X}_1\|_2 + \|\widetilde{X}_2 - \widehat{X}_2\|_2 \right] \\ &= e^{2\Delta C_{a,b}} \|\widetilde{X}_1 - \widehat{X}_1\|_2 + e^{\Delta C_{a,b}} \|\widetilde{X}_2 - \widehat{X}_2\|_2.\end{aligned}$$

This establishes the base case. Assuming that (A.7) holds for $k = m$, then for $k = m + 1$,

$$\begin{aligned}\|\overline{X}_{m+1} - \widetilde{X}_{m+1}\|_2 &\leq e^{\Delta C_{a,b}} \left[\|\overline{X}_m - \widetilde{X}_m\|_2 + \|\widetilde{X}_m - \widehat{X}_m\|_2 \right] \\ &\leq e^{\Delta C_{a,b}} \left[\sum_{i=1}^{m-1} e^{(m-i)\Delta C_{a,b}} \|\widetilde{X}_i - \widehat{X}_i\|_2 + \|\widetilde{X}_m - \widehat{X}_m\|_2 \right] \\ &= \sum_{i=1}^{m} e^{(m+1-i)\Delta C_{a,b}} \|\widetilde{X}_i - \widehat{X}_i\|_2,\end{aligned}$$

which completes the inductive proof and bounds the first term on the right-hand side of (A.5). Now,

$$\begin{aligned}\|\overline{X}_k - \widehat{X}_k\|_2 &\leq \sum_{i=1}^{k-1} e^{(k-i)\Delta C_{a,b}} \|\widetilde{X}_i - \widehat{X}_i\|_2 + \|\widetilde{X}_k - \widehat{X}_k\|_2 \\ &= \sum_{i=1}^{k} e^{(k-i)\Delta C_{a,b}} \|\widetilde{X}_i - \widehat{X}_i\|_2.\end{aligned}$$

Finally, using Pierce's lemma (Theorem 2.2.5), and Lemma 4.2.1 provides the desired result,

$$\begin{aligned}\|\overline{X}_k - \widehat{X}_k\|_2 &\leq K_{1,\delta} \sum_{i=1}^{k} e^{(k-i)\Delta C_{a,b}} \sigma_{2+\delta}(\widetilde{X}_i) N^{-1} \\ &\leq K_{1,\delta} \sum_{i=1}^{k} e^{(k-i)\Delta C_{a,b}} \|\widetilde{X}_i\|_{2+\delta} N^{-1} \\ &\leq K_{1,\delta} \sum_{i=1}^{k} a_i([a]_{\text{Lip}}, [b]_{\text{Lip}}, \Delta t, x_0, L, p) N^{-1}.\end{aligned}$$

□

Appendix B

Squared Bessel Processes

This appendix summarizes properties of squared Bessel processes that are relevant to real-world pricing.

Introduce $\mathbf{w} = \{\mathbf{w}_t = (w_t^1, w_t^2, \cdots, w_t^n)^\top, t \in [0, \infty)\}$, as an n-dimensional standard Brownian motion with $n \in \mathbb{N}$. Let the process $R = \{R_t = |\mathbf{w}_t|, t \in [0, \infty)\}$, be the Euclidean norm of $\mathbf{w}$, such that $(R_t)^2 = \sum_{i=1}^n (w_t^i)^2$. Itô's formula provides

$$d(R_t)^2 = n\, dt + \sum_{i=1}^{n} 2w_t^i\, dw_t^i.$$

For any $t > 0$, $\mathbb{P}(R_t = 0) = 0$, such that

$$dW_t = \frac{1}{R_t} \sum_{i=1}^{n} w_t^i dw_t^i$$

is a real-valued Brownian motion via Levý's characterization[1] and R satisfies

$$d(R_t)^2 = n\, dt + 2R_t\, dW_t.$$

Set $X_t = (R_t)^2$. Then for every $\delta \in \mathbb{N}$ and $X_0 = x \geq 0$, the unique, strong solution to the stochastic differential equation

$$dX_t = \delta\, dt + 2\sqrt{|X_t|}\, dW_t,$$

is known as a *squared Bessel process* of dimension δ, denoted BESQ_t^δ. Although this intuitive derivation accounts only for squared Bessel processes of positive and integer dimension, this can be extended to $\delta \in \mathbb{R}$, see, e.g., Revuz and Yor [1999].

Definition B.1 (BESQ_t^δ)**.** *For every $\delta \in \mathbb{R}$ and $x \in \mathbb{R}$, the unique strong solution to*

$$dX_t = \delta\, dt + 2\sqrt{|X_t|}\, dW_t, \tag{B.1}$$

with $X_0 = x$, is called a squared Bessel process of dimension δ, starting at x and denoted BESQ_t^δ.

To relate the stochastic differential equations that arise when modelling the growth-optimal portfolio to squared Bessel processes, Proposition 6.3.1.1 from Jeanblanc et al. [2009] is reproduced below.

[1] It is a continuous local martingale starting from zero and its quadratic variation is easily verified.

Proposition B.1. *Let* $S = \{S_t,\ t \in [0, \infty)\}$ *be a Cox-Ingersoll-Ross process satisfying*

$$dS_t = \kappa(\theta - S_t)\, dt + \sigma\sqrt{S_t}\, dW_t,$$

with $S_0 = x \geq 0$ *and* $\kappa,\ \theta > 0$ *and define* $\varphi(t) = \frac{\sigma^2}{4\kappa}(e^{\kappa t} - 1)$. *Then*

$$S_t = e^{-\kappa t} X_{\varphi(t)},$$

where $X_{\varphi(t)}$, $\varphi(t) \geq 0$, *is a* $\text{BESQ}^{\delta}_{\varphi(t)}$ *process with dimension* $\delta = \frac{4\kappa\theta}{\sigma^2}$.

This allows the square-root process to be expressed as a time-transformed squared Bessel process, for which the transition density is well-understood.

B.1 The Transition Density of the Squared Bessel Process

Lindsay and Brecher [2012] investigate the transition densities of squared Bessel processes under three different regimes, categorized by the dimension, δ. They follow the classic analysis of Feller [1951], who proceeded by solving the Fokker-Planck equation associated with a more general version of (B.1).

The case $\delta \leq 0$

When $\delta \leq 0$, the $X = 0$ boundary is attainable and absorbing. The fundamental solution to the associated Fokker-Plank equation is the transition density

$$p_{\delta\leq 0}(X_T, T; X_0) = \frac{1}{2T}\left(\frac{X_T}{X_0}\right)^{\frac{1}{2}\left(\frac{\delta}{2}-1\right)} \exp\left(-\frac{X_T + X_0}{2T}\right) I_{1-\frac{\delta}{2}}\left(\frac{\sqrt{X_T X_0}}{T}\right), \tag{B.2}$$

where $I_\nu(x)$ is a modified Bessel function of the first kind with index ν. By inspection, the above is related to the noncentral chi-squared density,

$$p_{\delta\leq 0}(X_T, T; X_0) = \frac{1}{T} p_{\chi'^2}\left(\frac{X_0}{T}; 4 - \delta, \frac{X_T}{T}\right), \tag{B.3}$$

expressed as a function of the noncentrality parameter, such that

$$\begin{aligned}\int_x^\infty p_{\delta\leq 0}(X, T; X_0)\, dX &= \int_x^\infty p_{\chi'^2}\left(\frac{X_0}{T}; 4 - \delta, \frac{X}{T}\right) dX, \\ &= \chi'^2\left(\frac{X_0}{T}; 2 - \delta, \frac{x}{T}\right).\end{aligned} \tag{B.4}$$

The final step above was proven by Schroder [1989].

This density is, however, *norm-decreasing*,

$$\int_0^\infty p_{\delta\leq 0}(X, T; X_0)\, dX = \chi'^2\left(\frac{X_0}{T}; 2 - \delta, 0\right) \leq 1, \tag{B.5}$$

as it does not include the probability of the process being absorbed at zero. Lindsay and Brecher [2012] propose constructing a full, norm-preserving density by adding a Dirac mass at the origin,

$$p^{\text{full}}_{\delta\leq 0}(X_T, T; X_0) := 2\left(1 - \chi'^2\left(\frac{X_0}{T}; 2 - \delta, 0\right)\right)\bar{\delta}(X_T) + p_{\delta\leq 0}(X_T, T; X_0), \tag{B.6}$$

where $\bar{\delta}(x)$ is the Dirac delta function. Then the distribution of X is given by

$$\mathbb{P}(X \leq X_T | X_0) = \int_0^{X_T} p^{\text{full}}_{\delta \leq 0}(X, T; X_0)\, dX = 1 - \chi'^2\left(\frac{X_0}{T}; 2-\delta, \frac{X_T}{T}\right). \tag{B.7}$$

The case $0 < \delta < 2$

When $0 < \delta < 2$, the $X = 0$ boundary is attainable and can be either absorbing or reflecting. If an absorbing boundary is selected, the analysis of the previous section holds,

$$p^{\text{A}}_{0<\delta<2}(X_T, T; X_0) := p^{\text{full}}_{\delta \leq 0}(X_T, T; X_0).$$

If a reflecting boundary is selected, the sign of the index of the Bessel function in the density changes,

$$\begin{aligned} p^{\text{R}}_{0<\delta<2}(X_T, T; X_0) &= \frac{1}{2T}\left(\frac{X_T}{X_0}\right)^{\frac{1}{2}\left(\frac{\delta}{2}-1\right)} \exp\left(-\frac{X_T + X_0}{2T}\right) I_{\frac{\delta}{2}-1}\left(\frac{\sqrt{X_T X_0}}{T}\right), \\ &= \frac{1}{T} p_{\chi'^2}\left(\frac{X_T}{T}; \delta, \frac{X_0}{T}\right), \end{aligned} \tag{B.8}$$

such that it is directly related to the non-central chi-squared density without the reversal of the roles of X_T and X_0 seen in the $\delta \leq 0$ case in (B.3). This density is clearly norm-preserving.

The case $\delta > 2$

When $\delta > 2$, the process can not attain zero and thus no boundary conditions can be specified. The transition density is of the same type as in the reflecting case,

$$p_{\delta>2}(X_T, T; X_0) = \frac{1}{T} p_{\chi'^2}\left(\frac{X_T}{T}; \delta, \frac{X_0}{T}\right). \tag{B.9}$$

Symmetry Relationships

When $\delta > 2$, the transition density is given by (B.9). When considering only an absorbing boundary, for all $\delta < 2$, the norm-decreasing part of the transition density is given by (B.2). A simple symmetry relationship exists between these two expressions,

$$p_\delta(X_T, T; X_0) = p_{4-\delta}(X_0, T; X_T), \tag{B.10}$$

that aids in the option pricing problems considered in this paper.

Furthermore,

$$X_T^{\left(1-\frac{\delta}{2}\right)} p_\delta(X_T, T; X_0) = X_0^{\left(1-\frac{\delta}{2}\right)} p_\delta(X_0, T; X_T) = X_0^{\left(1-\frac{\delta}{2}\right)} p_{4-\delta}(X_T, T; X_0). \tag{B.11}$$

The first equation follows from simple arithmetic using either (B.8) or (B.2) and the second equation follows from the first symmetry relationship above, (B.10).

An important implication of the above result is that if X is a BESQ$^\delta$ process with $\delta > 2$ and $X_0 > 0$, the process $Z_t = X_t^{\left(1-\frac{\delta}{2}\right)}$ is a *strict* local martingale[2]. This follows because the integral

[2] In fact it is a supermartingale, by Fatou's lemma, as it is bounded below

of the last term above is strictly less than one:

$$\begin{aligned}
\mathbb{E}[Z_T] &= \int_0^\infty X^{\left(1-\frac{\delta}{2}\right)} p_{\delta>2}(X,T;X_0)\,dX \\
&= \int_0^\infty X_0^{\left(1-\frac{\delta}{2}\right)} p_{(4-\delta)<2}(X_0,T;X)\,dX, \\
&= X_0^{\left(1-\frac{\delta}{2}\right)} {\chi'}^2\left(\frac{X_0}{T};\delta-2,0\right), \\
&< Z_0.
\end{aligned}$$

Appendix C

Volatility Corridor Swap Interpolation

Consider the payoff of a volatility corridor swap

$$\begin{aligned}\frac{1}{T}\int_0^T X_z \mathbb{I}_{\{L<S_z<H\}}\,dz &= \frac{1}{T}\sum_{k=0}^{K-1}\int_{t_k}^{t_{k+1}} X_z \mathbb{I}_{\{L<S_z<H\}}\,dz \\ &\approx \frac{1}{T}\sum_{k=0}^{K-1}\int_{t^*(S_{t_k})}^{t^*(S_{t_{k+1}})} \frac{X_{t_{k+1}}-X_{t_k}}{\Delta t}(z-t_k)+X_{t_k}\,dz, \end{aligned} \tag{C.1}$$

where on the last line the volatility process X_t has been approximated by a linear interpolation on the interval $t \in [t_k, t_{k+1}]$ with

$$t^*(s) = \begin{cases} t_k & \text{if } L \le s \le H \text{ and } s = S_{t_k}, \\ t_{k+1} & \text{if } L \le s \le H \text{ and } s = S_{t_{k+1}}, \\ \frac{H-S_{t_k}}{S_{t_{k+1}}-S_{t_k}}\Delta t + t_k & \text{if } s > H, \\ \frac{L-S_{t_k}}{S_{t_{k+1}}-S_{t_k}}\Delta t + t_k & \text{if } s < L, \end{cases}$$

providing the intercepts of the line connecting S_{t_k} and $S_{t_{k+1}}$ with the corridor. This interpolation is illustrated in Figure C.1 and accounts for the indicator function by constraining the integration

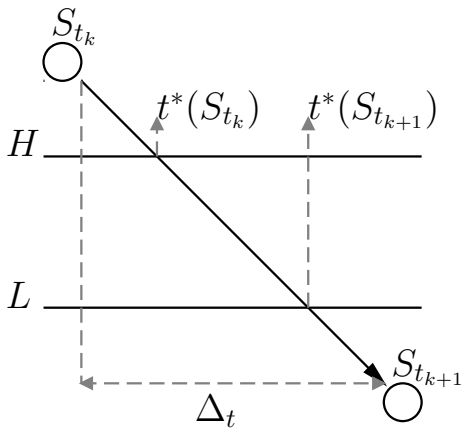

Figure C.1: Linear interpolation of the asset price provides the bounds for the integration.

to where the asset price is in the corridor. Explicitly computing a single term from (C.1) gives

$$G(t_k, t_{k+1}, X_{t_k}, S_{t_k}, X_{t_{k+1}}, S_{t_{k+1}}) := \\ \frac{X_{t_{k+1}} - X_{t_k}}{2\Delta t}\left[(t^*(S_{t_{k+1}}) - t_k)^2 - (t^*(S_{t_k}) - t_k)^2\right] + X_{t_k}\left[t^*(S_{t_{k+1}}) - t^*(S_{t_k})\right].$$

The value of a volatility corridor swap can now be computed as the expectation under the risk-neutral measure approximated using the quantization grids for X and S,

$$\mathbb{E}\left[\frac{1}{T}\int_0^T X_z \mathbb{I}_{\{L<S_z<H\}}\, dz\right] \approx \frac{1}{T}\sum_{k=0}^{K-1}\sum_{i=1}^{N^x}\sum_{j=1}^{N^x}\sum_{u=1}^{N^y}\sum_{v=1}^{N^y} G(t_k, t_{k+1}, x_k^i, s_k^u, x_{k+1}^j, s_{k+1}^v) \\ \times \mathbb{P}(\widehat{X}_k = x_k^i, \widehat{S}_k = s_k^u, \widehat{X}_{k+1} = x_{k+1}^j, \widehat{S}_{k+1} = s_{k+1}^v),$$

with the probability

$$\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{S}_k = s_k^u, \widehat{X}_{k+1} = x_{k+1}^j, \widehat{S}_{k+1} = s_{k+1}^v) = \\ \mathbb{P}(\widehat{X}_{k+1} = x_{k+1}^j, \widehat{S}_{k+1} = s_{k+1}^v | \widehat{X}_k = x_k^i, \widehat{S}_k = s_k^u)\mathbb{P}(\widehat{X}_k = x_k^i, \widehat{S}_k = s_k^u)$$

computed as part of the matrix formulation in Section 5.2.2.

Appendix D

Analytical Pricing Formulae

The GBM Model

At time t, the risk-neutral price of a European call option written on a stock, $S = \{S_t,\ t \in [0,T]\}$, with dynamics given by (3.20), is

$$c^{\mathrm{BS}}_{T,K}(t,S_t) = S_t\Phi(d_1) - Ke^{-r(T-t)}\Phi(d_2),$$

with

$$d_1 = \frac{\ln\left(\frac{S_t}{K}\right) + \left(r + \frac{1}{2}\sigma^2\right)(T-t)}{\sigma\sqrt{T-t}} \quad \text{and} \quad d_2 = d_1 - \sigma\sqrt{T-t},$$

where the option has strike K and maturity T. Put-call parity provides the price of the corresponding put option as

$$p^{\mathrm{BS}}_{T,K}(t,S_t) = -S_t\Phi(-d_1) + Ke^{-r(T-t)}\Phi(-d_2).$$

The CEV Model

At time t, the risk-neutral prices of European call and put options written on a stock, $S = \{S_t,\ t \in [0,T]\}$, with dynamics given by (3.21), are

$$c^{\mathrm{CEV}}_{T,K}(t,S_t) = S_t\left[1 - \chi'^2(y;z,x)\right] - Ke^{-r(T-t)}\chi'^2(x;z-2,y)$$

and

$$p^{\mathrm{CEV}}_{T,K}(t,S_t) = -S_t\chi'^2(y;z,x) + Ke^{-r(T-t)}\left[1 - \chi'^2(x;z-2,y)\right]$$

with

$$x = \kappa S_t^{2(1-\alpha)}e^{2r(1-\alpha)(T-t)}, \qquad y = \kappa K^{2(1-\alpha)},$$
$$z = \frac{1-2\alpha}{1-\alpha}, \qquad \kappa = \frac{2r}{\sigma^2_{\mathrm{CEV}}(1-\alpha)\left(e^{2r(1-\alpha)(T-t)} - 1\right)},$$

and the options each have strike K and maturity T. Note that here $\infty < \alpha < 1$ and an absorbing boundary is selected for $\alpha < 0.5$.

Bibliography

D. H. Ahn and B. Gao. A parametric nonlinear model of term structure dynamics. *The Review of Financial Studies*, 12(4):721–762, 1999.

H. Albrecher, P. Mayer, W. Schoutens, and J. Tistaert. The little Heston trap. *KU Leuven Section of Statistics Technical Report*, 06(05), 2006.

J. Baldeaux, K. Ignatieva, and E. Platen. A Tractable Model for Indices Approximating the Growth-optimal Portfolio. *Studies in Nonlinear Dynamics and Econometrics*, 18(1):1–21, 2014.

J. Baldeaux, M. C. Fung, K. Ignatieva, and E. Platen. A Hybrid Model for Pricing and Hedging of Long-dated Bonds. *Applied Mathematical Finance*, 22(4):366–398, 2015.

V. Bally, J. Printems, et al. A quantization tree method for pricing and hedging multidimensional American options. *Mathematical Finance*, 15(1):119–168, 2005.

O. Bardou, S. Bouthemy, and G. Pages. Optimal quantization for the pricing of swing options. *Applied Mathematical Finance*, 16(2):183–217, 2009.

G. Bormetti, G. Callegaro, G. Livieri, and A. Pallavicini. A backward Monte Carlo approach to exotic option pricing. *European Journal of Applied Mathematics*, pages 1–42, 2017.

O. Burkovska, M. Gaß, K. Glau, M. Mahlstedt, W. Schoutens, and B. Wohlmuth. Calibration to American Options: Numerical Investigation of the de-Americanization. *arXiv:1611.06181*, 2016.

G. Callegaro, L. Fiorin, and M. Grasselli. Quantized Calibration in Local Volatility. *Risk Magazine*, 28:62–67, 2015.

G. Callegaro, L. Fiorin, and M. Grasselli. Pricing via recursive quantization in stochastic volatility models. *Quantitative Finance*, 17(6):855–872, 2016.

B. Chen, C. Oosterlee, and J. van der Weide. Efficient unbiased simulation scheme for SABR stochastic volatility model. *International Journal of Theoretical and Applied Finance*, 15(2), 2012.

S. Corlay and G. Pagès. Functional quantization-based stratified sampling methods. *Monte Carlo Methods and Applications*, 21(1):1–32, 2015.

S. Delattre, J.-C. Fort, and G. Pagès. Local distortion and μ-mass of the cells of one dimensional asymptotically optimal quantizers. *Communications in Statistics-Theory and Methods*, 33(5): 1087–1117, 2004.

F. Delbaen and W. Schachermayer. A general version of the fundamental theorem of asset pricing. *Mathematische Annalen*, 300(1):463–520, 1994.

F. Delbaen and W. Schachermayer. The fundamental theorem of asset pricing for unbounded stochastic processes. *Mathematische Annalen*, 312(2):215–250, 1998.

F. Delbaen and W. Schachermayer. *The Mathematics of Arbitrage*. Springer Science & Business Media, 2006.

F. Diener and M. Diener. Asymptotics of the price oscillations of a European call option in a tree model. *Mathematical Finance*, 14(2):271–293, 2004.

Q. Du, V. Faber, and M. Gunzburger. Centroidal Voronoi tessellations: Applications and algorithms. *SIAM Review*, 41(4):637–676, 1999.

M. Emelianenko, L. Ju, and A. Rand. Nondegeneracy and weak global convergence of the Lloyd algorithm in $\mathbb{R}^d$. *SIAM Journal on Numerical Analysis*, 46(3):1423–1441, 2008.

W. Feller. Two singular diffusion problems. *Annals of Mathematics*, 54(1):173–182, 1951.

L. Fiorin, G. Pagès, and A. Sagna. Product Markovian quantization of an $\mathbb{R}^d$-valued Euler-scheme of a diffusion process with applications to finance. Technical report, May 2017. URL https://hal.archives-ouvertes.fr/hal-01034196.

C. Fontana. Weak and strong no-arbitrage conditions for continuous financial markets. *International Journal of Theoretical and Applied Finance*, 18(01):1550005, 2015.

H. Geman, N. El Karoui, and J.-C. Rochet. Changes of numeraire, changes of probability measure and option pricing. *Journal of Applied probability*, 32(2):443–458, 1995.

P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, 2003.

S. Graf and H. Luschgy. *Foundations of Quantization for Probability Distributions*. Springer, 2000.

R. M. Gray and D. L. Neuhoff. Quantization. *IEEE transactions on information theory*, 44(6): 2325–2383, 1998.

C. Gschnaidtner and M. Escobar. Parameters Recovery via Calibration in the Heston model. 2015.

P. S. Hagan, D. Kumar, A. S. Lesniewski, and D. E. Woodward. Managing smile risk. *The Best of Wilmott*, 1:249–296, 2002.

P. S. Hagan, D. Kumar, A. S. Lesniewski, and D. E. Woodward. Universal Smiles. *Wilmott*, 2016 (84):40–55, 2016.

N. H. Hakansson and W. T. Ziemba. Capital growth theory. *Handbooks in Operations Research and Management Science*, 9:65–86, 1995.

D. Heath and E. Platen. Consistent pricing and hedging for a modified constant elasticity of variance model. *Quantitative Finance*, 2(6):459–467, 2002.

P. Heckbert. *Color image quantization for frame buffer display*, volume 16. ACM, 1982.

S. L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies*, 6(2):327–343, 1993.

Y.-L. Hsu, T. Lin, and C. Lee. Constant elasticity of variance (CEV) option pricing model: Integration and detailed derivation. *Mathematics and Computers in Simulation*, 79(1):60–71, 2008.

H. Hulley and E. Platen. Hedging for the long run. *Mathematics and Financial Economics*, 6(2): 105–124, 2012.

P. Hunt and J. Kennedy. *Financial Derivatives in Theory and Practice*. John Wiley & Sons, 2004.

A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., 1988.

M. Jeanblanc, M. Yor, and M. Chesney. *Mathematical Methods for Financial Markets*. Springer, 2009.

I. Karatzas and C. Kardaras. The numeraire portfolio in semimartingale financial models. *Finance and Stochastics*, 11(4):447–493, 2007.

J. Kelly. A new interpretation of information rate. *IRE Transactions on Information Theory*, 3 (2):185–189, 1956.

P. Kloeden and E. Platen. *Numerical Solution of Stochastic Differential Equations*, volume 23. Springer, 1999.

R. Korn and S. Tang. Exact analytical solution for the normal SABR model. *Wilmott*, (66):64–69, 2013.

H. J. Kushner and P. Dupuis. *Numerical methods for stochastic control problems in continuous time*, volume 24. Springer, 2001.

A. Lindsay and D. Brecher. Simulation of the CEV process and the local martingale property. *Mathematics and Computers in Simulation*, 82(5):868–878, 2012.

S. P. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28 (2):129–137, 1982.

J. B. Long. The numeraire portfolio. *Journal of Financial Economics*, 26(1):29–69, 1990.

R. Lord, R. Koekkoek, and D. V. Dijk. A comparison of biased simulation schemes for stochastic volatility models. *Quantitative Finance*, 10(2):177–194, 2010.

H. Luschgy and G. Pagès. Functional quantization rate and mean regularity of processes with an application to Lévy processes. *The Annals of Applied Probability*, 18(2):427–469, 2008.

G. Maruyama. Continuous Markov processes and stochastic equations. *Rendiconti del Circolo Matematico di Palermo*, 4(1):48–90, 1955.

T. A. McWalter, R. Rudd, J. Kienitz, and E. Platen. Recursive marginal quantization of higher-order schemes. *Available at SSRN 2894753*, 2017.

T. A. McWalter, R. Rudd, J. Kienitz, and E. Platen. Recursive marginal quantization of higher-order schemes. *Quantitative Finance*, 2018. doi: 10.1080/14697688.2017.1402125. URL https://doi.org/10.1080/14697688.2017.1402125.

S. M. Miller and E. Platen. Analytic pricing of contingent claims under the real-world measure. *International Journal of Theoretical and Applied Finance*, 11(08):841–867, 2008.

S. M. Miller and E. Platen. Real-world pricing for a modified constant elasticity of variance model. *Applied Mathematical Finance*, 17(2):147–175, 2010.

G. Milstein. Approximate integration of stochastic differential equations. *Theory of Probability and Its Applications*, 19(3):557–562, 1975.

H. Niederreiter. *Random number generation and quasi-Monte Carlo methods*. SIAM, 1992.

J. Oblój. Fine-tune your smile: Correction to Hagan et al. *arXiv:0708.0998*, 2007.

G. Pagès. A space quantization method for numerical integration. *Journal of Computational and Applied Mathematics*, 89(1):1–38, 1998.

G. Pagès. Introduction to numerical probability for finance. *Master Course Notes, University Paris VI*, 2008a.

G. Pagès. Quadratic optimal functional quantization of stochastic processes and numerical applications. In *Monte Carlo and Quasi-Monte Carlo Methods*, pages 101–142. Springer, 2008b.

G. Pagès. Introduction to optimal vector quantization and its applications for numerics. Technical report, July 2014. URL https://hal.archives-ouvertes.fr/hal-01034196.

G. Pagès and J. Printems. Functional quantization for numerics with an application to option pricing. *Monte Carlo Methods and Applications mcma*, 11(4):407–446, 2005.

G. Pagès and J. Printems. Optimal quantization for finance: from random vectors to stochastic processes. *Handbook of Numerical Analysis*, 15:595–648, 2009.

G. Pagès and A. Sagna. Recursive marginal quantization of the Euler scheme of a diffusion process. *Applied Mathematical Finance*, 22(5):463–498, 2015.

G. Pagès and B. Wilbertz. Optimal Delaunay and Voronoi quantization methods for pricing American options. In R. Carmona, P. Hu, P. Del Moral, and N. Oudjane, editors, *Numerical Methods in Finance*, pages 171–217. Springer, 2009.

G. Pagès, H. Pham, and J. Printems. Optimal quadratic quantization for numerics: the Gaussian case. *Monte Carlo Methods and Applications*, 9(2):135–165, 2003.

G. Pagès, H. Pham, and J. Printems. Optimal quantization methods and applications to numerical problems in finance. In *Handbook of Computational and Numerical Methods in Finance*, pages 253–297. Springer, 2004.

L. Paulot. Asymptotic implied volatility at the second order with application to the SABR model. In *Large Deviations and Asymptotic Methods in Finance*, pages 37–69. Springer, 2015.

E. Platen. A minimal financial market model. Discussion Papers, Interdisciplinary Research Project 373: Quantification and Simulation of Economic Processes 2000,91, Berlin, 2001. URL http://hdl.handle.net/10419/62176. urn:nbn:de:kobv:11-10048178.

E. Platen. Arbitrage in continuous complete markets. *Advances in Applied Probability*, 34(3): 540–558, 2002.

E. Platen. Pricing and hedging for incomplete jump diffusion benchmark models. In *AMS-IMS-SIAM Joint Summer Research Conference on Mathematics of Finance*. American Mathematical Society, 2004.

E. Platen. A benchmark approach to finance. *Mathematical Finance*, 16(1):131–151, 2006.

E. Platen. Law of the minimal price. Technical report, University of Technology, Sydney. QFRC Research Paper 215, 2008.

E. Platen and D. Heath. *A Benchmark Approach to Quantitative Finance*. Springer, 2006.

E. Platen and R. Rendek. Approximating the numeraire portfolio by naive diversification. *Journal of Asset Management*, 13(1):34–50, 2012.

D. Revuz and M. Yor. *Continuous Martingales and Brownian Motion*. Springer, 1999.

R. Rudd, T. A. McWalter, J. Kienitz, and E. Platen. Fast quantization of stochastic volatility models. *Available at SSRN 2956168*, 2017.

R. Rudd, T. A. McWalter, J. Kienitz, and E. Platen. Quantization under the real-world measure: Fast and accurate valuation of long-dated contracts. *arXiv:1801.07044*, 2018.

A. Sagna. Pricing of barrier options by marginal functional quantization. *Monte Carlo Methods and Applications*, 17(4):371–398, 2011.

P. A. Samuelson. St. Petersburg paradoxes: Defanged, dissected, and historically described. *Journal of Economic Literature*, 15(1):24–55, 1977.

R. Schöbel and J. Zhu. Stochastic volatility with an Ornstein–Uhlenbeck process: an extension. *European Finance Review*, 3(1):23–46, 1999.

M. Schroder. Computing the constant elasticity of variance option pricing formula. *Journal of Finance*, 44(1):211–219, 1989.

E. M. Stein and J. C. Stein. Stock price distributions with stochastic volatility: an analytic approach. *Review of Financial Studies*, 4(4):727–752, 1991.

P. Zador. Topics in the asymptotic quantization of continuous random variables. *Unpublished Bell Laboratories Memorandum*, 1966.