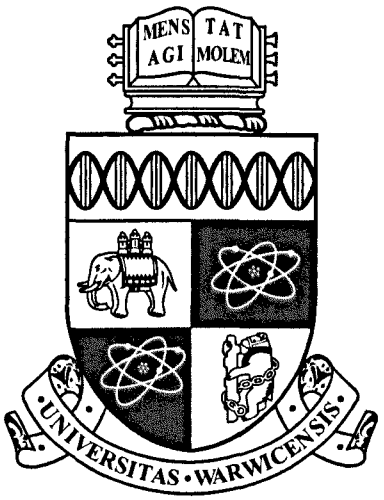


The Valuation of Exotic Barrier Options and American Options using Monte Carlo Simulation

by

Pokpong Chirayukool

Thesis

Submitted to the University of Warwick

for the degree of

Doctor of Philosophy

Warwick Business School

September 2011

THE UNIVERSITY OF
WARWICK

Contents

List of Tables v

List of Figures viii

Acknowledgments x

Declarations xi

Abstract xii

Chapter 1 Introduction 1

I Contour bridge Simulation 4

Chapter 2 Literature Review on Barrier Options and Hitting Times 5

2.1 An overview of barrier options 5

2.2 Valuation of barrier options 7

2.2.1 Valuation methods with GBM and constant barrier 7

2.2.2 Non-GBM processes 13

2.2.3 Non-constant barriers 16

2.3 Hitting times 18

2.3.1 z_t is a Brownian motion 19

2.3.2 z_t is a non-Brownian motion 20

2.4 Conclusion 21

Chapter 3 Valuing Exotic Barrier Options using the Contour Bridge Method 22

3.1 Simulation methods for barrier options 23

3.1.1 Dirichlet Monte Carlo method 23

3.1.2 Brownian bridge Monte Carlo method 25

3.2 The contour bridge simulation method 26
3.2.1 The choice of contour 27
3.2.2 Hitting time sampling method 29
3.2.3 The contour bridge method algorithm 35
3.2.4 The stopping conditions 37
3.2.5 Computing the value of S_T 40
3.2.6 Single-hit method barriers 42
3.2.7 The biggest-bite variant 43
3.2.8 The vertical contours variant 44
3.2.9 Valuation of a book of options 44
3.3 Numerical results 45
3.3.1 The benchmark option 46
3.3.2 The exotic options 51
3.3.3 Numerical results 53
3.4 Conclusion 64

II American Put Options and Control Variates **65**

Chapter 4 Literature Review on American Option Valuation **66**
4.1 American put option valuation problems 66
4.2 Approximation and numerical methods 68
4.2.1 Analytical approximation methods 68
4.2.2 PDE and lattice methods 70
4.3 Simulation methods 71
4.3.1 Mesh methods 72
4.3.2 State-space partitioning methods 74
4.3.3 Duality methods 76
4.3.4 Functional form methods 77
4.3.5 Overview of the control variate method 80
4.3.6 Control variate method for American option valuation 82
4.4 American option valuation in stochastic volatility, Lévy and jump processes, and stochastic interest rate models 85
4.4.1 Stochastic volatility 85
4.4.2 Lévy and jump processes 87
4.4.3 Stochastic interest rate 89
4.5 American barrier and power options 89
4.5.1 American barrier 90

4.5.2 Power option 90
4.6 Conclusion 90

Chapter 5 Valuing Bermudan and American Put Options with Bermudan Put Option Control Variates **92**
5.1 Exercise at fixed times: Bermudan options 93
5.2 Bermudan option control variate 93
5.2.1 Bermudan put-terminal control variate 96
5.2.2 Bermudan put-tau control variate 97
5.2.3 Obtaining values of τ_j from simulation 100
5.3 Two-phase simulation method 100
5.3.1 First phase: Estimating the early exercise boundary 100
5.3.2 Second phase: Computing the option value 103
5.4 Approximate American put options using Richardson extrapolation . 105
5.4.1 Two-point scheme 105
5.4.2 Three-point scheme 106
5.4.3 Implementing Richardson extrapolation with Monte Carlo and control variates 107
5.5 Numerical results 108
5.5.1 First phase results 109
5.5.2 Second phase results 110
5.6 Conclusion 119

Chapter 6 Valuing American Put Options using the Sequential Contour Monte Carlo Method **121**
6.1 The sequential contour Monte Carlo (SCMC) Method 121
6.1.1 Sequential contour (SC) options 123
6.1.2 Sequential contour path construction 124
6.1.3 The SCMC algorithm: the generalisation of the LSLS algorithm 129
6.1.4 Using SC put options to approximate American put options: Richardson extrapolation 130
6.2 Sequential contour construction 131
6.2.1 Choice of g^{N-1} 131
6.2.2 Choices of α^0, α^{N-1}, and $g^i, i = 1, \ldots, N-2$ 132
6.3 Control variates from the SCMC method 136
6.3.1 Valuing barrier options by using hitting time simulation . . . 136
6.3.2 Hitting times and options 137
6.3.3 Rebate options from the SCMC method 138

6.3.4 Rebate option control variate 139
6.4 Numerical Results 140
6.4.1 Choice of contours 140
6.4.2 Early exercise boundary of sequential contour options 143
6.4.3 Valuing standard American put options 144
6.5 Conclusion 153

III Projection Techniques and Exotic American Options 154

Chapter 7 Valuing Exotic American Options using the Sequential Contour Monte Carlo Method 155
7.1 Sequential contour bridge (SCB) method 156
7.1.1 Contour construction for the SCB method 156
7.1.2 The SCB algorithm 158
7.2 Different projection techniques 158
7.2.1 Generalised LSLS Algorithm with projection operator 160
7.2.2 Hitting times projection ($\mathcal{T}$-projection) 161
7.2.3 Contour distance projection ($\mathcal{D}$-projection) 162
7.3 Application to American fractional power call options 163
7.3.1 Valuation of an American fractional power call option 164
7.3.2 Contour construction for an American fractional power call option 166
7.4 Application to American linear barrier fractional power call options . 169
7.5 Numerical Results 172
7.5.1 Sequential contour bridge (SCB) method with American put options 173
7.5.2 American fractional power call option results 177
7.5.3 American knock-in option results 179
7.5.4 Benchmark options: flat barrier with power $\kappa = 1$ 179
7.5.5 Exotic American options 180
7.6 Conclusion 185

Chapter 8 Conclusion 187

Appendix A The derivation of the hitting time bridge distribution 192

Appendix B Compute Arc Length of an exponential contour 196

List of Tables

2.1 Examples of τ-option and ι-option 6

3.1 Benchmark valuation, rebate, flat barrier. $\sigma = 0.1$ 48
3.2 Benchmark valuation, rebate, flat barrier. $\sigma = 0.2$ 49
3.3 Benchmark valuation, rebate, flat barrier. $\sigma = 0.3$ 50
3.4 Exotic barrier types and parameters 53
3.5 Non-constant barriers 54
3.6 Rebate option $\mathcal{O}^R$, $\sigma = 0.2$ 55
3.7 Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.2$ 55
3.8 Knock-in put option $\mathcal{O}^{IP}$, $\sigma = 0.2$ 56
3.9 Recovery option $\mathcal{O}^{rec}$, $\sigma = 0.2$ 56
3.10 Knock-in call option $\mathcal{O}^{IC}$ and knock-out put option $\mathcal{O}^{OP}$, single-hit ι, $\sigma = 0.2$ 58
3.11 Rebate option $\mathcal{O}^R$, close to the barrier, $\sigma = 0.2$ 59
3.12 Knock-in call option $\mathcal{O}^{IC}$, close to the barrier, $\sigma = 0.2$ 59
3.13 Rebate option $\mathcal{O}^R$, far from the barrier, $\sigma = 0.2$ 60
3.14 Knock-in call option $\mathcal{O}^{IC}$, far from the barrier, $\sigma = 0.2$ 60
3.15 Rebate option $\mathcal{O}^R$, $\sigma = 0.1$ 62
3.16 Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.1$ 62
3.17 Rebate option $\mathcal{O}^R$, $\sigma = 0.3$ 63
3.18 Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.3$ 63

5.1 Four cases of the Bermudan put control variate, c^{b2} 97
5.2 Laguerre polynomial characteristics 102
5.3 RMSEs comparison between the put-tau, c_τ^p, and a combination of the put-tau and the Bermudan put-tau, c_τ^{p+b2}, rollback method . . . 109
5.4 Summary of control variates used in table 5.5 111
5.5 Bermudan put values and variance reduction gains from various kinds of Bermudan control variates by using Lattice boundary 112

5.6 Two-phase method to value Bermudan put options. Put+Bermudan tau roll back with S_0-dispersion. Pricing control variates are put-tau and put-tau + Bermudan early. Benchmark price is 4.00646. 114
5.7 Two-phase method to value Bermudan put options. $X = 100$. Benchmark value is 6.08118. 115
5.8 Two-phase method to value Bermudan put options. Benchmark value is 8.72811 116
5.9 Extrapolated American put values with pricing control variate only . 118
5.10 Extrapolated American put values with pricing and $v(0)$ control variate 118
5.11 Efficiency gains from adding $v(0)$ control 119

6.1 T-modified path illustration, $T = 1$ and $N = 6$ 127
6.2 Summary of SC option types 132
6.3 Options defined by hitting times 137
6.4 Extrapolated American put values from type 4 SC option with different values of $k_{\max}$ and $k_{\min}$. $\beta^{N-1}(0) = 5$ and $g^{N-1} = 2.99$. Benchmark value is 6.09037 142
6.5 Contour parameter values 142
6.6 Extrapolated American put option values from the SCMC method with different types, $\sigma = 0.2$ 143
6.7 Extrapolated American put values from SC option values without the $v(0)$ control 145
6.8 Extrapolated American put values from SC option values with $v(0)$ control 147
6.9 Correlations between extrapolated American put and $v(0)$ controls, $\rho\left(\widehat{v}^{\infty}, \widehat{v}^{N}\right)$ 148
6.10 Extrapolated American put values from the sequential contour options with only pricing control 151
6.11 Extrapolated American put values from the sequential contour options with pricing and $v(0)$ control 152

7.1 Comparison of American and European fractional power call options 163
7.2 American fractional call values with different $\beta^{N-1}(T)$ 167
7.3 American put, $\sigma = 0.2$ 174
7.4 Bias in extrapolated American put values using $\mathcal{T}$-projection 176
7.5 American fractional power call option with forward evolution (standard SCMC) method. $\sigma = 0.2$ 178

7.6 Benchmark case: knock-in flat barrier American call options with $\kappa = 1$. 181
7.7 Knock-in linear barrier fractional power American call options, $S_0 = 105$ 182
7.8 Knock-in linear barrier fractional power American call options, $S_0 = 100$ 183
7.9 Knock-in linear barrier fractional power American call options, $S_0 = 98$.184

List of Figures

3.1 Illustration of a bridge hitting time $\tau^{\alpha} \mid \tau^{\alpha_1}, \tau^{\alpha_2}$ 28
3.2 Hitting time densities where $\nu > 0$ and $\nu < 0$ 32
3.3 Contour bridge algorithm illustration 37
3.4 Intersection time illustration 39
3.5 Bimodal bridge hitting time density; $f(\tau^{\alpha} \mid \tau^{\alpha_1}, \tau^{\alpha_2})$. $\tau^{\alpha_1} = 0.25$, $\tau^{\alpha_2} = 0.26$, $\frac{1}{\alpha_1} = 70.2$, $\frac{1}{\alpha} = 70$, $\frac{1}{\alpha_2} = 69.8$ 41
3.6 Biggest bite illustration 44

4.1 American put option's early exercise boundary (EEB) 68

5.1 Convergence of a Bermudan option value to an American option value from a trinomial lattice method 94
5.2 A B-spline defined by a set of knot points $\{0, 2, 4, 8, 16\}$ 104

6.1 Bermudan and SC option's exercise opportunities 125
6.2 Illustration of path construction 126
6.3 Iteration of the SCMC algorithm 130
6.4 The four types of sequential contours 135
6.5 Asset value simulation from the SCMC method 136
6.6 Contour shape with different $k_{\max}$ and $k_{\min}$. $N = 50$, $\beta^{N-1}(0) = 5$ and $g^{N-1} = 2.99$. 141
6.7 Early exercise boundaries of the sequential contour puts and the American put 144
6.8 Convergence of SC put option values to American put values for various S_0 146
6.9 Correlations between $c_T^{R,i}$ with $\widehat{v}^{64}$, their covariances, and standard deviations for different contours $\beta^i(t)$ 149

7.1 Illustration of different projection techniques 159

7.2 Convergence of Bermudan fractional call option values to the American value from a lattice method 165
7.3 American fractional power call options' EEB 166
7.4 Contour construction for an American fractional power call with its EEB from Richardson extrapolation 168
7.5 The values of S_T (area (a), (b) and (c)) 169
7.6 Continuation values, regression values and exercise values from the $\mathcal{S}$-projection and the $\mathcal{T}$-projection method, for a SC put option, on the 62th contour 175
7.7 Continuation values, regression values and exercise values from the $\mathcal{T}$-projection method, for a SC fractional call option, on the 62th contour 179

A.1 Illustration of bridge hitting times 193

Acknowledgments

I would like to first show my gratitude to my supervisor, Dr. Nick Webber, for his continual supports, advices, detailed comments, and patience for the past four years. I have learnt a lot from him. This thesis would not have been completed without his help.

I am truly indebted to my parents, Mrs. Vipada and Mr. Jitti Chirayukool for their encouragements, understandings and financial supports throughout my study. Thanks are due to them for giving me an opportunity to pursue this PhD. I could not have finished this thesis without them.

Many thanks go to Ms Somsawai Kuldiloke for always being encouraging, considerate, and supportive. I deeply thank her for being an excellent listener. Without her, I could not have made it.

Declarations

I hereby declare that this thesis is the result of my original research work and effort. When other sources of information have been used, they have been acknowledged. This thesis has not been submitted anywhere for any award.

Abstract

Monte Carlo simulation is a widely used numerical method for valuing financial derivatives. It can be used to value high-dimensional options or complex path-dependent options. Part one of the thesis is concerned with the valuation of barrier options with complex time-varying barriers. In Part one, a novel simulation method, the contour bridge method, is proposed to value exotic time-varying barrier options. The new method is applied to value several exotic barrier options, including those with quadratic and trigonometric barriers.

Part two of this thesis is concerned with the valuation of American options using the Monte Carlo simulation method. Since the Monte Carlo simulation can be computationally expensive, variance reduction methods must be used in order to implement Monte Carlo simulation efficiently. Chapter 5 proposes a new control variate method, based on the use of Bermudan put options, to value standard American options. It is shown that this new control variate method achieves significant gains over previous methods. Chapter 6 focuses on the extension and the generalisation of the standard regression method for valuing American options. The proposed method, the sequential contour Monte Carlo (SCMC) method, is based on hitting time simulation to a fixed set of contours. The SCMC method values American put options without bias and achieves marginal gains over the standard method. Lastly, in Part three, the SCMC method is combined with the contour bridge method to value American knock-in options with a linear barrier. The method can value American barrier options very well and efficiency gains are observed.

Chapter 1

Introduction

Barrier options are well-known exotic options that are traded in the market. They are popular among investors since they are cheaper than the corresponding European options and they can be used to gain or to reduce an exposure. They also occur in debt instrument and arise in credit risk application. Although there are several standard valuation methods that can be used to value barrier option accurately, these methods often work when the shape of the barrier is simple such as a flat or an exponential barrier, and can value only a single option at a time.

The more complex versions of barrier options are characterised by conditional or non-constant barriers. Even though standard simulation methods can be used to value these complex options, they are often slow and prone to simulation biases. The first part of this thesis is concerned with the development of a novel simulation method to value complex-shape barrier options accurately and to improve on the standard simulation method in term of computational costs.

American options and American style derivatives have become one of the most common financial instruments traded both in exchange and in the over-the-counter market. These types of options are difficult to value because of their early exercise feature. This special feature of American options complicates the pricing problem because one needs to solve for the exercise boundary simultaneously with the option value. The standard Black-Scholes partial differential equation (PDE) becomes a free boundary value problem. Because in several settings partial different equation (PDE) methods may be numerically difficult to use, Monte Carlo simulation can be used as an alternative numerical method to price American options. However, Monte Carlo simulation can be computationally expensive. Therefore, variance reductions techniques must be used in order to implement Monte Carlo simulation efficiently. The development of a new control variate technique and

a new simulation method to value standard and exotic American options are the main focus of part two and part three of the thesis.

There are three main contributions of this thesis. The first contribution is the development of a novel simulation method, based on a hitting time simulation, to value exotic barrier options and to improve upon the standard simulation method. The second contribution is the development of a new control variate technique, based on the use of a Bermudan option, to value Bermudan and American put options. The new control variate method can be used to reduce substantially computational costs of a simulation method. The third contribution is the extension and the generalisation of the standard Least Squares Monte Carlo method of Longstaff and Schwartz (1995) [177] (LSLS) to value standard and exotic American options.

Throughout this thesis, the time t asset value S_t is assumed to follow the geometric Brownian motion. That is,

$$\mathrm{d}S_t = rS_t\mathrm{d}t + \sigma S_t\mathrm{d}z_t, \tag{1.0.1}$$

where z_t is a Brownian motion, r is a risk free rate and σ is volatility. A risk free rate and volatility are assumed to be constant.

At the beginning of the thesis, Chapter 2 provides a literature review of valuation methods that have been used to value barrier options. It also provides a general review of the hitting time, a concept that is crucial to the barrier option valuation problem and the development of the new simulation technique.

In Chapter 3 of the thesis, the novel simulation method, the contour bridge method, is introduced and is described. The method is applied to the valuation of exotic barrier options. Several complex non-constant barrier options are introduced. These range from a not very complicated linear barrier option to a more complex trigonometric-shape barrier option. The method is based on a hitting time simulation and a bridge hitting time simulation to a specific type of contour. Numerical results show that biases in resulting barrier option values are negligible and most of the time the contour bridge method achieves substantial efficiency gains over standard simulation methods.

Part two of the thesis begins by reviewing existing American valuation methods. Chapter 4 reviews semi-analytical and numerical methods used in valuing standard American options both in the geometric Brownian motion model and in other models. The chapter also reviews the valuation of American options by using simulation techniques. Different simulation methods are reviewed and described. The standard regression method, the LSLS method, is also analysed. Several control

variate techniques for valuing American options are reviewed and discussed. The chapter lastly describes the pricing of American barrier options and a power option; the two options that will be considered in the last part of the thesis.

In Chapter 5, a new control variate technique is proposed. The new method makes use of the twice-exercise Bermudan put option as a control variate. The Bermudan control variate is implemented to value Bermudan put options using a two-phase Monte Carlo simulation method. The results suggest that the Bermudan control variate method gives accurate values for Bermudan options and achieves significant efficiency gains over the plain LSLS method. It is also shown that American put option values can be approximated accurately by employing the three-point Richardson extrapolation method. The standard errors of an extrapolated American value are also reduced by using a value of an individual Bermudan put as a control variate.

Chapter 6 extends and generalises the standard LSLS method for valuing American put options. The new method, the sequential contour Monte Carlo (SCMC) method, simulates hitting times to a fixed set of exponential contours. The method generalises the LSLS method to a more general family of contours and instead of iterating backward on a fixed time step, the method iterates backward on each contour with a varied time step. It is shown that sampled hitting times to contours can be used to value barrier options and these options can be used as control variates. Numerical examples show that the method can value American put options without evidence of bias and it achieves marginal efficiency gains over the standard method.

Chapter 7 extends the SCMC method by introducing two new projection techniques to approximate an option's continuation values. The new techniques are the $\mathcal{T}$- and the $\mathcal{D}$-projections. These techniques are applied to value American put and American fractional call options. The second part of Chapter 7 demonstrates the application of these techniques to value exotic American options; the American linear knock-in fractional power call option. To value this option, the SCMC method is modified by incorporating the contour bridge method introduced in Chapter 3. The method can value American knock-in options without any evidence of bias and also achieve respectable efficiency gains over the standard LSLS method.

Lastly, Chapter 8 presents the conclusion and observations of the complete thesis. Recommendations for future research are pointed out. There is an appendix consisting of two parts. The first part is concerned with the derivation of the bridge hitting time density that is used in Chapter 3. The second part shows the derivation of a distance on an exponential contour that is used in Chapter 7.

Part I

Contour bridge Simulation

Chapter 2

Literature Review on Barrier Options and Hitting Times

Barrier options have become popular exotic options that are traded in the market. They enable investors to avoid or obtain the exposure and they are cheaper than the corresponding European options. They can be monitored either continuously or discretely. There are explicit solutions for barrier options for a simple case; single or double constant barriers with single underlying asset that follows a geometric Brownian motion process in (1.0.1). For a more complicated barrier option or process, numerical methods must be used. A concept that is closely related and is important to barrier options is that of a hitting time. It is the first time that a stochastic process crosses a certain barrier level. The hitting time has been used in several applications in option pricing such as the valuation of barrier options and American options.

This chapter provides an overview of barrier options, their valuation methods and hitting times to several types of barrier. An overview of barrier options is discussed in section 2.1. Section 2.2 reviews several valuation methods for barrier options. The review of hitting times is provided in section 2.3.

2.1 An overview of barrier options

Barrier options are a modified form of standard plain-vanilla European options. The payoff of barrier options depends on whether or not an asset value hits a pre-specified barrier level during the life of an option. An in-option will pay out only if an asset value crosses a barrier. On the other hand, an out-option will pay out only if an asset value does not cross a barrier during the life of an option. If the initial asset

value at time t_0 is above the initial value of the barrier, an option is referred to as a down-option. If the initial asset value is below the time t_0 value of the barrier, an option is referred to as an up-option. A barrier option can have either single or double barriers.

Let $b(t), t \in [0,T]$ be a barrier level at time t for a barrier option maturing at time T and S_t be an asset value at time t. If a barrier is constant, one writes $b(t) \equiv b$. Write τ for the first hitting time of $S = (S_t)_{t\geq 0}$ to the option barrier. The first hitting time for a barrier option is defined as

$$\tau = \min\{t \in [0,T] : S_t = b(t)\}. \tag{2.1.1}$$

Let $\iota = \mathbb{I}_{\tau \leq T}$, where $\mathbb{I}_{\tau \leq T}$ is the indicator function, be a variable determining whether or not the barrier is hit before time T. Barrier options can be categorised into two main groups. The first group is an option whose payoff depends directly on τ. The second group is an option whose payoff depends on τ only through ι. The former is referred to as a τ-option and the latter is referred to as an ι-option. There is also a special case of an ι-option called a bare-ι option. Examples are shown in table 2.1. Usually, to value an ι-option, one needs to compute additional

Type	Example
τ-option	rebate option, paying at τ
ι-option	knock-in option knock-out option
bare-ι option	knock-out option, paying rebate at T

Table 2.1. Examples of τ-option and ι-option

quantity such as the asset value of maturity time T, S_T, to compute an option's payoff. However, if a payoff depends only on ι, for instance a knock-out option that pays a fixed rebate R at time T, this type of option is referred to as a bare-ι option.

The concept of barrier option can also be used in the context of credit application. For instance, in a structural model of default, the default time is the first hitting time of an asset value to a certain barrier level. A bond can be viewed as a τ-option. In reduced form credit models, default may be modelled as occurring when a count-down process hits a barrier level, so bond prices again resemble τ-options.

2.2 Valuation of barrier options

Even though there are explicit solutions to value barrier options, most of the analytical formulae are for the simplest types. Merton (1973) [182] first presented an explicit solution of a down-and-out call option with one flat barrier when an underlying process follows a geometric Brownian motion (GBM). More analytical formulae, with this simple case, were provided in Reiner and Rubinstein (1991) [205] and Rich (1994) [211] and Kunitomo and Ikeda (1992) [162]with exponential barriers.

For more complicated types of barrier options, perhaps those with non-constant barriers or with non-geometric Brownian underlying processes, numerical methods must be applied.

2.2.1 Valuation methods with GBM and constant barrier

There are several numerical methods used to value barrier options in the literature. These are lattice methods, partial differential equation (PDE) methods, analytical approximation and quadrature methods and Monte Carlo simulation methods. When an option has more than one barrier, these standard methods have to be modified to obtain values with reasonable accuracy.

2.2.1.1 Lattice methods

One of the most popular methods for pricing barrier options is the lattice method. Without any modification, the standard lattice method converges very slowly when valuing barrier options. When the lattice nodes do not lie exactly on the barrier, a lattice will price an option with a wrong barrier, resulting in a bias in option values. To reduce the bias, a large number of time steps must be used. This results in greater computational times. There are several methods whose purpose is to correct bias without using a large number of time steps.

Boyle and Lau (1994) [42] suggested a method where the number of time steps N is chosen such that the barrier lies on lattice nodes. This method is often referred to as the 'shifting the node' method. Even though this method can reduce biases substantially, it cannot be applied when there is more than one barrier. This idea was also adopted by Ritchken (1995) [212] and Dai and Lyuu (2007) [80]. Ritchken (1995) [212] pointed out that a trinomial lattice may have an advantage over a binomial lattice because a trinomial lattice enables one to have more flexibility in terms of ensuring that lattice nodes lie on the barrier. Ritchken (1995) [212] proposed a method that makes use of a stretching parameter that adjusts an asset

value so that nodes can be placed on a barrier. A trinomial lattice is also applied to value discrete barrier options by Broadie et al. (1999) [51]. Dai and Lyuu (2007) [80] constructed a lattice that has the barrier lying on lattice nodes and applied a combinatorial algorithm. The method where the option value is interpolated from two values computed from nodes lying above and below the barrier was proposed by Derman et al. (1995) [84]. Recently, a method where the drift term of the underlying asset can be fitted dynamically at each time step so that an asset value will coincide with the barrier, was suggested by Wöster (2010) [248].

Another lattice method incorporates conditional hitting probabilities into branching probabilities of a lattice. This method was described in Kuan and Webber (2003) [160] and Barone-Adesi et al. (2008) [28]. The method requires the conditional hitting probabilities to a barrier to be known. However, even though they are not known, a linear piecewise approximation can be used. Kuan and Webber (2003) [160] showed that the method works well both for plain vanilla barrier options and for barrier Bermudan options.

Ahn et al. (1999) [2] considered a discrete barrier option in the case where an initial asset value is close to to the barrier and proposed an adaptive mesh method. This is essentially a trinomial lattice in which more refined lattice branchings are constructed in areas where more accuracy is needed. This idea is adopted from Figlewski and Gao (1999) [91] who applied an adaptive mesh method to continuous barrier options. Steiner et al. (1999) [229] extended the interpolation technique of Derman et al. (1995) [84] to value discrete knock-out options when an initial asset value lies close to the barrier. The method also takes into account barrier hitting probabilities of nodes near the barrier. The convergence of the lattice method with a barrier option with arbitrary discrete monitoring dates was studied by Hörfelt (2003) [126].

2.2.1.2 PDE methods

PDE-based methods have also been applied to value barrier options. Suppose an asset value S_t follows a GBM in (1.0.1). Let v_t be the value at time t of an option written on S_t maturing at time T. The method applies the Black-Scholes PDE:

$$\frac{\partial v}{\partial t} + rS\frac{\partial v}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 v}{\partial S^2} = rv \tag{2.2.1}$$

with boundary conditions that are suited for a barrier option. Different ways used to discretise partial derivatives in (2.2.1) give rise to different variants of the technique.

The explicit finite difference method for valuing both continuous and discrete barrier options was proposed by Boyle and Tian (1998) [43]. They modified the

partition of grid points on the y-axis such that the barrier is always on the grid and implemented quadratic interpolation to compute an option value. The method employs a more refined grid near the barrier to eliminate problems that arise when an initial asset value S_0 is close to the barrier. When S_0 is not close to the barrier, the method's performance is similar to that of the Ritchken (1995) [212] method. This comes as no surprise since an explicit finite difference method can be viewed as a trinomial lattice method.

The implicit finite difference method for valuing barrier options was suggested Zvan et al. (2000) [253]. They showed that their implicit method converges faster than the explicit method because the explicit method requires a much smaller grid spacing. They also pointed out that the Crank-Nicolson scheme (thought of as a method that is in between the explicit and implicit methods), even though stable, can produce oscillating results when valuing barrier options if a certain condition is not satisfied. Wade et al. (2007) [242] suggested a technique for smoothing high-order Crank-Nicolson schemes for discrete barrier options. An adaptive finite element technique to value barrier options was discussed by Foufas and Larson (2004) [93]. They show that the resulting barrier option's values are in agreement of those computed from the implicit finite difference method of Zvan et al. (2000) [253]. The implicit method was also used by Ndogmo and Ntwiga (2007) [191]. They applied coordinate transformation to restrict asset values to be in a particular range. To handle valuation problems in areas near the barrier, adaptive grids are used. Their main idea is to reduce the solution domain by incorporating conditional hitting probabilities to a boundary condition. This method is similar to the method of Kuan and Webber (2003) [160] and Barone-Adesi et al. (2008) [28] in a lattice method context and also to the Dirichlet Monte Carlo method that will be discussed later.

Although lattice and PDE-based methods can be used to value barrier options effectively, they may be difficult to use to value simultaneously a book of barrier options with different strikes and barriers. For instance, an implicit PDE method has to be used to value one option at a time.

2.2.1.3 Semi-analytical methods

Other types of method that can be applied to value barrier options are semi-analytical methods. They are semi-analytical because usually a solution is in an integral form and hence numerical integration is required. Mijatović (2010) [187] presented a semi-analytical solution to value double barriers options with time-dependent parameters. The method provides an integral representation of the bar-

rier premium, which is the difference between the barrier option value and the corresponding European option value.

A quadrature technique for valuing discrete barrier options was applied by Sullivan (2000) [233] and Fusai and Recchioni (2007) [98]. Sullivan (2000) [233] presented a method that combines numerical (multi-dimensional) integration with function approximation. This technique is applied at the monitoring dates of the option. Fusai and Recchioni (2007) [98] employed a similar idea to value discrete barrier options in the constant elasticity of variance (CEV) model of Cox and Ross (1976) [75]) and Variance Gamma (VG) model of Madan et al. (1998) [179].

There are a number of papers concerning the use of Laplace transform to value barrier options, with both single and double barriers. These include Lin (1998) [166], Pelsser (2000) [197], Hulley and Platen (2007) [129], Davydov and Linetsky (2002) [82] and Wang et al. (2009) [245].

Lin (1998) [166] applied the Laplace transform technique of Gerber and Shiu (1994, 1996) [101, 102] to the double barrier hitting distribution. The option value is expressed as an infinite sum. Pelsser (2000) [197] applied contour integration to invert the Laplace transform in order to obtain the double hitting time density. Option values are then obtained by integrating with respect to the density. Hulley and Platen (2007) [129] presented the Laplace transform of the option value, using numerical quadrature to evaluate integral terms and then inverting the Laplace transform of the option value.

Wang et al. (2009) [245] proposed a hybrid method that combines the Laplace transform with the finite difference method to value both single and double barrier options. The method eliminates the t-dependent term in the Black-Scholes PDE in (2.2.1) by using the Laplace transform. Then the method applies finite difference method to discretise S-dependent terms. They show that the method converges faster than lattice methods discussed earlier. This is because when the t-dependent term is eliminated, there is no problem arising from partitioning time steps unlike other numerical methods.

A path counting method to obtain the double hitting time distributions was employed by Sidenius (1998) [226], and with these distributions, an analytical solution is obtained.

Semi-analytical methods often require that the hitting distribution to the barrier is known. When working with more complex barriers, such as those that will be described in chapter 3, it will be difficult for these methods to work well. This encourages the use of simulation methods.

2.2.1.4 Simulation methods

Simulation methods have several advantages. For instance, they can be used in a situation where more state variables are required or where an underlying process is not amenable to other methods. However, when valuing barrier options, naive simulation methods can be slow and may suffer from biases in option values. The standard Monte Carlo simulation method will be described in detail in section 3.1.

There is a well known problem when valuing barrier options using Monte Carlo simulation called the barrier breaching problem. Consider a down and out option, this is a situation where, for each two consecutive times t_i and t_{i+1}, both asset values S_{t_i} and $S_{t_{i+1}}$ lie above the barrier. However, there is a possibility that the barrier may have been hit in the interval (t_i, t_{i+1}) but the simulation technique cannot detect this, resulting in a bias in the option value. One simple way to reduce the bias is to increase the number of time steps, but this will increase the computational cost considerably.

Bias correction methods

To correct the biases in Monte Carlo simulation without too much computational cost, several methods have been proposed in the literature. Two general approaches have been made. The first method is to sample from the distribution of the minimum (for 'down'-option) or the maximum (for 'up'-option) of asset values. This method is sometimes referred to as the Brownian bridge approach since an asset value is sampled conditional on two end points. To implement this method, the distribution of the maximum or the minimum of an asset value must be known. Beaglehole et al. (1997) [30] applied this method to value barrier options when an underlying process follows a geometric Brownian motion. The method is as follows:

Consider a random stock price path S, starting from an initial value S_{t_i} at time t_i and ending to a value $S_{t_{i+1}}$ at time t_{i+1}. One is interested in computing the conditional probability of S hitting a constant barrier b during the time interval $[t_i, t_{i+1})$ given the values of S_{t_i} and $S_{t_{i+1}}$. S_{t_i} is assumed to be greater than b.

Denote a minimum of a process $Z = (Z_t)_{t \geq 0}$ as

$$m_{t,T}^{Z} \equiv m_{t,T}^{Z}(S) = \min_{k \in [t,T]} \{Z_k(S)\}. \tag{2.2.2}$$

Let $\left\{\widehat{S}_{t_i}\right\}_{i=1,..,N}$ be a stock price path generated from the Monte Carlo method with N number of time steps and $\{\widehat{m}_{t_i,t_{i+1}}\}_{i=1,..,N}$ be a sample of $\{m_{t_i,t_{i+1}}\}_{i=1,..,N}$ which

is generated from the Brownian bridge method. Write

$$m\left(\widehat{S}\right) = \min_{i=1,..,N}\left\{\widehat{S}_{t_i}\right\} \tag{2.2.3}$$

$$m\left(\widehat{m}\right) = \min_{i=1,..,N-1}\left\{\widehat{m}_{t_i,t_{i+1}}\right\}. \tag{2.2.4}$$

The conditional distribution of the minimum (for simplicity, write $\widehat{S} = S$) is given as:

$$\mathbb{P}\left[m^S_{t_i,t_{i+1}} \mid S_{t_i}, S_{t_{i+1}}\right] = \exp\left(\frac{-2(\ln S_{t_i} - \ln b)(\ln S_{t_{i+1}} - \ln b)}{\sigma^2(t_{i+1} - t_i)}\right), \tag{2.2.5}$$

and, with a univariate $U \sim \mathrm{U}[0,1]$,

$$\ln S^{\min}_{t_i} = \frac{\ln S_{t_{i+1}} + \ln S_{t_i} - \sqrt{(\ln S_{t_{i+1}} - \ln S_{t_i})^2 - 2\sigma^2(t_{i+1} - t_i)\ln(U)}}{2} \tag{2.2.6}$$

is a draw from $m^S_{t_i,t_{i+1}}$.

Beaglehole et al. (1997) [30] showed that this method can remove simulation bias for a down-and-out call option with one flat barrier. This method was applied to construct a lattice to value American options by Sidenius (1998) [226]. Shevchenko (2003) [225] adopted the method of Beaglehole et al. (1997) [30] to value multi-asset knock-out options.

The second bias correction method for the Monte Carlo simulation is to compute the conditional hitting probabilities and use them to weight an option's payoff. This idea is described in Baldi et al. (1999, 1999) [20, 21]. Suppose a knock-out option has a payoff $h(S_T)$ where S_T is an asset value at time T. The method computes a conditional hitting probability between time t_i and t_{i+1}, p_i. If there are N time steps, then the payoff of a knock-in option can be computed as

$$h\left(S_T\right)\left(1 - \prod_{i=0}^{N-1}\left(1 - p_i\right)\right). \tag{2.2.7}$$

This method is referred to as the Dirichlet method and will be discussed further in the next chapter. Baldi et al. (1999) [20] applied a large deviation technique to obtain analytical forms for p_i for flat and exponential barriers both single and double. The formula for the single exponential barrier will be shown and used in chapter 3. The valuation of rebate option and parisian option using this method was described in Baldi et al. (1999) [21].

Even though these correction methods can be applied to correct biases in

barrier options, they are limited to cases where the probability is available. These are cases where a barrier is flat or exponential.

Variance reduction with simulation methods

A method that can be used with the Monte Carlo simulation to value barrier option is importance sampling. In this case, the method is used to reduce the variance of the possibility of a barrier breaching at each monitoring date. The technique involves changing the probability measure from which sample paths are generated. Glasserman (2004) [110] provided a general background of the application of the importance sampling in option pricing.

The importance sampling technique to value barrier options was suggested by Glasserman and Staum (2001) [111]. The method is to incorporate survival (not hitting) probabilities p' into the generation of uniform random variable, U'. Then the asset value is sampled using U'. Since the method reduces only the variance of a knock-out event, not the variance of the payoff at maturity time T, the method may not perform well in a situation when the payoff variance is high. An example is an option with large volatility and a long time to maturity. In this case, the method should be used in conjunction with other variance reduction techniques. Joshi and Leung (2007) [145] use the importance sampling method to value barrier options in a jump-diffusion model. The technique modifies the jump size drawn by incorporating the hitting probability between each time step. The modified jump size ensures that a sampled asset value already reflects the possibility of the barrier being breached. The use of stratified sampling on hitting times to value discrete barrier options is investigated by Joshi and Tang (2010) [146].

Control variate methods for barrier option valuation are described in Kim and Henderson (2007) [158], Ehrlichman and Henderson (2007) [89], Fouque and Han (2004) [95] and Fouque and Han (2006) [96]. Their methods are based on the use of constructed martingales as control variates. Kim and Henderson (2007) [158] and Ehrlichman and Henderson (2007) [89] proposed the adaptive control variate method in which the iteration procedure is employed to approximate the martingale. The use of martingale control variates in valuing barrier options in stochastic volatility models was discussed by Fouque and Han (2004, 2006) [95, 96].

2.2.2 Non-GBM processes

There are a number of papers that are concerned with the valuation of barrier options in the non-GBM context. Ribeiro and Webber (2006) [208] employed the method of

Beaglehole et al. (1997) [30] to correct simulation bias in barrier option values where the underlying asset follows Lévy processes. Carr and Hirsa (2007) [58] derived the valuation equation for barrier options in a general class of Lévy processes. Because of the existence of jumps, there will be an integral term in the differential equations analogous to (2.2.1) and thus one obtains the partial integro-differential equation (PIDE). To solve the PIDE, Carr and Hirsa (2007) [58] employed a finite difference method.

The transform approach to value barrier options in Lévy models has been proposed by Jeannin and Pistorius (2010) [139]. They applied the Weiner-Hopf factorisation and then used the Laplace transform to obtain expressions for the option value. Boyarchenko and Levendorskiĭ (2010) [41] employed the randomisation method of Carr (1998) [56] to value double barrier options with a wide class of Lévy models. The valuation of double barrier options under a flexible jump diffusion model was investigated by Cai et al. (2009) [54] This is a model where jump sizes are assumed to be hyper-exponentially distributed. They presented a double Laplace transform and then employ a numerical inversion method to obtain the option value. The valuation of barrier options in jump-diffusion models using a simulation method with importance sampling is described in Joshi and Leung (2007) [145]. Metwally and Atiya (2002) [184] also applied a concept of importance sampling to estimate the density of hitting times for jump processes. Bernard et al. (2008) [33] developed a valuation approach to value barrier options in the stochastic interest rate model of Vasicek (1977) [240]. They found a semi-analytical formula to value a single constant barrier option. The pricing problem of barrier options in the stochastic volatility model of Heston (1993) [122] was investigated in Griebsch and Wystup (2011) [117]. They expressed an option value as an n-dimensional integral. To evaluate this integral, they employed a fast Fourier transform (FFT) method and a multidimensional numerical integration.

The valuation of a barrier option in which the underlying asset has stochastic dividends was discussed by Graziano and Rogers (2006) [116]. They derived a semi-analytical solution that can be used to value single and double barrier options.

The valuation of barrier options in the CEV model has been investigated by several authors. Boyle and Tian (1999) [44] employed a trinomial lattice to develop a discrete approximation for the CEV process. An asset value is transformed such that the new process has constant volatility. A trinomial lattice is constructed based on the transformed process and the lattice branching is modified using the stretching method of Ritchken (1995) [212]. A PDE-based method to price barrier options in the CEV model was investigated by Lo et al. (2001) [172]. They employed an

eigenvalue expansion approach to obtain the solution.

Another method is a Lie-algebra approach used by Lo et al. (2000) [175]. The method is applied to value barrier options with time-dependent parameters. The option value involves an infinite sum with integral terms which may have to be computed by numerical integration methods. A similar idea was also used by Lo and Hui (2006) [170] and Lo et al. (2009) [174]. This technique was extended to valuing barrier options with time dependent parameters by Lo and Hui (2006) [170]. Lo et al. (2009) [174] incorporated time-dependent volatilities into the price process of moving barrier options.

Models with time-dependent parameters were also investigated by Novikov et al. (2003) [194], Rapisarda (2005) [202], Roberts and Shortland (1997) [214] and Lo et al. (2003) [173]. Roberts and Shortland (1997) [214] valued a barrier option when the risk-free rate is a deterministic function. They make use of the approximation technique proposed by Roberts and Shortland (1995) [213]. Numerical integration is required in order to obtain the option price. Novikov et al. (2003) [194] proposed a method that is based on a piecewise linear approximation and repeated integration. The method of images was employed by Lo et al. (2003) [173] to obtain upper and lower bounds of barrier options with time dependent parameters. Rapisarda (2005) [202] extended the results of Lo et al. (2003) [173] and applied a perturbation expansion method to obtain a system of PDEs which yields an expression of the option value involving integrals and the sum of infinite series.

Barrier option pricing in the variance-gamma (VG) and the normal inverse Gaussian (NIG) models was investigated in Ribeiro and Webber (2003, 2004) [206, 207] and Becker (2010) [31]. Ribeiro and Webber (2003) [206] proposed the use of a gamma bridge in conjunction with a stratified sampling method (see Glasserman (2004) [110] for a general review of a stratified sampling method in the Monte Carlo application) to value barrier options in the variance-gamma model. Their results show that substantial speed-ups can be obtained. The use of simulation techniques to value barrier options in the VG model was also investigated by Becker (2010) [31]. He applied the difference-of-gamma bridge sampling method of Avramidis et al. (2003) [18] and Avramidis and L'Ecuyer (2006) [17]. The technique provides bounds of simulated VG paths, which can be used to obtain information about a path hitting a barrier in each time interval. Ribeiro and Webber (2004) [207] investigated the valuation of average rate options in the NIG model using simulation. They suggested the use of an inverse Gaussian bridge with stratified sampling. Even though their paper does not provide results for barrier options, their work is important and is related to the contour bridge method described in chapter 3. In particular, one part

of the method that will be presented in the next chapter uses a result of Ribeiro and Webber (2004) [207], with a slight modification.

2.2.3 Non-constant barriers

A type of barrier that is relatively more tractable than others is an exponential barrier. It is of the form

$$b(t) = b(0)\mathrm{e}^{gt}, \qquad 0 \leq t \leq T, \tag{2.2.8}$$

where $g \in \mathbb{R}$ is a barrier growth rate. The valuation problem for exponential barrier options was investigated by Kunitomo and Ikeda (1992) [162]. They presented the solution of the double exponential barrier options as an infinite series by using the known hitting time densities of a Brownian motion to a linear barrier. (A correction to one of the equations in their paper was made by Kunitomo and Ikeda (2000) [161].) Buchen and Konstanatos (2009) [52] investigated double exponential barrier options with arbitrary payoff. They employed the method of images and obtained the solution that involves the sum of an infinite series. Thompson (2002) [235] studied bounds on values of barrier options with exponential barriers.

Another type of method that is used to value barrier options with exponential barriers is lattice methods. This is done by Rogers and Stapleton (1998) [217], Costabile (2002) [74] and Kuan and Webber (2003) [160]. Rogers and Stapleton (1998) [217] modified the CRR binomial method of Cox et al. (1979) [76] to value barrier options with exponential barriers. The method is based on interpreting the random walk on the lattice to be equally-spaced in space rather than equally-spaced in time. The probability of moving to the consecutive nodes is modified.

Costabile (2002) [74] extended the binomial lattice to value exponential barrier options. The method is based on shifting nodes such that they lie on the exponential barrier. The method is shown to yield more accurate option values than the trinomial method proposed by Ritchken (1995) [212]. Kuan and Webber (2003) [160] considered barrier options in which barriers are in the form $b(t) = c_1 \exp(c_2 t^2 + c_3 t)$ and $b(t) = c_4 - c_1 \exp(c_2 t^2 + c_3 t)$, where c_1, c_2, c_3 and c_4 are constants. They valued non-constant barriers in the stochastic Dirichlet framework. Their method uses a Brownian bridge hitting time distribution to construct a lattice. Their method is shown to be superior to those of plain lattice and Monte Carlo methods.

Rogers and Zane (1997) [218] implemented a state space transformation method together with a trinomial lattice to value double exponential barrier op-

tions. They transformed a log price process $X_t = \ln(S_0) + \mu t + \sigma Z_t$ to

$$\widetilde{X}_t = \frac{X_t - l_t}{u_t - l_t}, \tag{2.2.9}$$

where u_t and l_t are continuously once differentiable barrier functions.[1] Then the process $\widetilde{X}_t$ is time-transformed to a process $\bar{X}_t$ with unit volatility and deterministic drift term. They then applied a trinomial lattice with the transformed process $\bar{X}_t$to value barrier options. Their results were similar (maximum of 0.04% difference) to the semi-analytical solutions of Kunitomo and Ikeda (1992) [162].

Rogers and Zane (1997) [218] investigated barrier options with a linear barrier of a form:

$$b(t) = b(0) + mt \qquad 0 \leq t \leq T. \tag{2.2.10}$$

In this case there is no explicit solution because the hitting time density to a linear barrier of a geometric Brownian motion is not known and hence the probabilistic approach fails. Rogers and Zane (1997) [218] also applied their transformation method to value double linear barrier options in which an upper and a lower barrier takes the form (2.2.10). However, they did not compare the results with benchmark values. Ballestra and Pacelli (2009) [25] applied boundary element methods to value linear double barrier options with barriers of the form (2.2.10) where $m = 1$. They presented an integral representation of the barrier option value and obtain a solution by solving a system of integral equations. The method is extended to value a quadratic barrier and a kinked barrier option.

Morimoto and Takahashi (2002) [190] investigated barrier options with a square root barrier of forms

$$b(t) = b(0)\sqrt{(t)}, \tag{2.2.11}$$

$$b^U(t) = S_0 \exp\left(at + g\sigma\sqrt{t}\right), \tag{2.2.12}$$

$$b^L(t) = S_0 \exp\left(at - g\sigma\sqrt{t}\right), \tag{2.2.13}$$

where (2.2.12) and (2.2.13) are upper and lower barriers for double barrier options. The method makes use of an asymptotic expansion to estimate the hitting probability. A numerical integration is required in order to obtain the option value.

Hui (1997) [127] investigated the partial barrier option. This is an option that the barrier exists for only a certain part of an option's life. Since the barrier is constant, Hui (1997) [127] provided explicit solutions for several types of partial

[1]For example, an exponential barrier gives $u_t = \ln(u_0) + gt$.

barrier options. Dorfleitner et al. (2008) [87] applied a Green's function method to solve the PDE of barrier options with time dependent parameters. They employed their method to value a quadratic barrier option and also a flat barrier option with power payoff.

Kijima and Suzuki (2007) [156] investigated a barrier option valuation problem in a credit application. They employed a change of measure technique to derive a closed form solution for a knock-out exchange option whose barrier is a fraction of the value of other security.

2.3 Hitting times

A concept that is crucial to the problem of barrier options valuation is that of hitting times. For a continuous process z, the first hitting time of the level β for the process z is defined as

$$\tau = \inf\{t \geq 0 : z_t = \beta\}. \tag{2.3.1}$$

If $z_0 > \beta$, then

$$\tau^- = \inf\{t \geq 0 : z_t \leq \beta\}. \tag{2.3.2}$$

Similarly, if $z_0 < \beta$, then

$$\tau^+ = \inf\{t \geq 0 : z_t \geq \beta\}. \tag{2.3.3}$$

Generally β is allowed to be a function of time t. In this case, $\beta(t)$ is called a contour. Densities of τ are available in a number of cases depending on the process z_t and the contour $\beta(t)$.

Perhaps the most common hitting time density is when z_t is a Brownian motion: $z_t = \mu t + \sigma w_t, z_0 = 0$, where w_t is a standard Brownian motion. The hitting density $f(t;\mu,\sigma)$, where contour $\beta(t) = \beta$ is constant, is given by (Karatzas and Shreve (1991) [151]).

$$f(t;\mu,\sigma) = \frac{\beta}{\sigma\sqrt{2\pi t^3}} \exp\left(-\frac{(\beta-\mu t)^2}{2\sigma^2 t}\right). \tag{2.3.4}$$

$f(t;\mu,\sigma)$ is an inverse Gaussian distribution.

When $z_t = S_t$ where S_t is a geometric Brownian motion and $\beta(t)$ is an exponential contour:

$$\beta(t) = \frac{1}{\alpha}\exp(gt), \tag{2.3.5}$$

where g is a contour growth rate and $\frac{1}{\alpha} = \beta(0)$ is time t_0 value of a contour (one

can view α as a parameter that controls time t_0 value of an exponential contour), in this case the hitting time density is

$$f(t;\alpha,g,r,\sigma) = \frac{a}{\sigma\sqrt{2\pi t^3}} \exp\left(-\frac{(a-(g-\hat{\mu})\,t)^2}{2\sigma^2 t}\right), \tag{2.3.6}$$

where $\hat{\mu} = r - \frac{1}{2}\sigma^2$ and $a = \ln\left(\frac{S}{\beta(0)}\right) = \ln(S\alpha)$. The density (2.3.6) will be discussed further in section 3.2. There are a number of studies of hitting time probabilities and distributions. Examples are summarised as follows.

2.3.1 z_t is a Brownian motion

When a contour $\beta(t)$ is linear, the density is known by the Bachelier-Lévy formula (see Siegmund and Yuh (1982) [227] and Lerche (1986) [164]).

The hitting time distribution of a linear contour was investigated by Hobson et al. (1999) [125]. They used Taylor expansions to approximate the hitting probability of a Brownian motion starting both above and below a contour. Durbin and Williams (1992) [88] studied a hitting probability in a curved contour case. The hitting distribution of quadratic contours was investigated by Alili and Patie (2005) [8]. Wang and Pötzelberger (1997) [244] discussed a case where a contour is piecewise linear. A hitting time density of a two-dimensional Brownian motion to a constant contour was investigated by Iyengar (1985) [135].

Daniels (2000) [81] used a numerical method to approximate a hitting time distribution of a contour of the form $\beta(t) = c_1 + c_2 t + c_3 \sin t$ where c_1, c_2 and c_3 are constants. The hitting time distribution of a similar form of contour was also studied by Roberts and Shortland (1995) [213] who applied a hazard rate tangent approximation.

There are a number of authors who investigate hitting time distributions when a contour involves a square root of time. Let $a, b, c \in \mathbb{R}$ be constants. The idea is to transform a Brownian motion to an Ornstein-Uhlenbeck (OU) process and then compute a hitting distribution of a transformed process to a constant contour. A case where a contour is in a form $\beta(t) = c\sqrt{t}$ was investigated by Breiman (1966) [45]. Sato (1977) [222] investigated the hitting distribution of a contour of the form $\beta(t) = c\sqrt{1+t}$. Jennen and Lerche (1981) [140] considered a case where $z_0 < \beta(0)$ and provided a general form of a hitting distribution of the form $\beta(t) = \sqrt{2\,(c+at)}$ and $\beta(t) = \sqrt{(2a + \ln(t+1))\,(t+1)}$. Ricciardi et al. (1984) [210] presented the hitting distribution to a contour of the form $\beta(t) = c + bt^{1/k}$, where $k \geq 2$, in an integral equation. Park and Paranjape (1976) [196] investigated hitting probabilities of a

Brownian motion to various types of contour including $\beta(t) = 1 + \sqrt{t}$.

Analytical expressions, that involve hypergeometric functions, for hitting time distributions to contours $\beta(t) = a+b\sqrt{c+t}$ and $\beta(t) = b\sqrt{c+t}$ where $z_0 < \beta(0)$ were provided by Novikov et al. (1999) [193]. The use of the method of images for approximating the hitting time distribution to a contour $\beta = a\sqrt{t \ln\left(\frac{b}{t}\right)}$ was done by Kahale (2008) [149].

The hitting distribution to a contour $\beta(t) = at^k$ where $k < \frac{1}{2}$ has been investigated by Park and Paranjape (1976) [196] and Jennen and Lerche (1981) [140]. A quadratic case where $k = 2$ was investigated by Salminen (1988) [220].

2.3.2 z_t is a non-Brownian motion

There are a number of studies of hitting time distributions to constant contours when an underlying process is not a Brownian motion. The processes considered are the Ornstein-Uhlenbeck (OU) process, the Bessel process, the CEV process and Lévy processes.

The hitting distribution of an OU process is expressed as the Laplace transform of a function of a three-dimensional Bessel bridge by Leblanc et al. (2000) [163]. A limitation of the formula of Leblanc et al. (2000) [163] was pointed out by Göing-Jaeschke and Yor (2003) [114]. The three-dimensional Bessel bridge method was corrected by Alili et al. (2005) [9]. They also provided two additional techniques used to approximate the hitting distribution to a constant contour. These are an eigenvalue expansion and an integral representational method. The OU-process hitting distribution was analysed by Ricciardi and Sato (1988) [209] and Salminen et al. (2007) [221]. Lo and Hui (2006) [171] computed an upper bound and a lower bound for the OU-process hitting time distribution using the method of images. By using a martingale technique, Novikov (2004) [192] also found bounds of OU-process hitting distributions when the OU process has a jump component. The hitting distribution of an OU process was applied to a credit application by Linetsky (2004) [168] and Yi (2010) [249].

The hitting time distribution of the Bessel process was investigated by Pitman and Yor (1982) [199], Kent (1980, 1982) [154, 155] and Pitman and Yor (2003) [198]. Kent (1980, 1982) [154, 155] employed an eigenfunction expansion to obtain the hitting distribution. When z_t is a CEV process, Linetsky (2004) [169] provided a hitting time distribution and applied it to derive an explicit solution for the value of a look-back option. The hitting probability of the CEV process was used in a credit application by Atlan and Leblanc (2006) [16] and Campi et al. (2009) [55].

Hitting time distributions in jump-diffusion models were investigated by

Kou and Wang (2003) [159], Zhang and Melnik (2007) [250] and Atiya and Metwally (2005) [15]. For a class of Lévy processes, Alili and Kyprianou (2005) [7] provided a relationship between hitting time distributions and a smoothing pasting condition of an American put option. Hitting time distributions of the generalised inverse-Gaussian process was studied by Barndorff-Nielsen et al. (1978) [26]. An application of hitting distributions in Lévy models to an optimal stopping problem was investigated by Mordecki (2002) [188]. Hitting times in Lévy models were analysed in Imkeller and Pavlyukevich (2006) [133] and Roynette et al. (2008) [219].

2.4 Conclusion

This chapter has reviewed several methods that have been used to value barrier options both in GBM and non-GBM models. This chapter points out that even though these methods can be applied to value a single barrier option, they may fail when the value of a book of options is required. Also, it is not clear how these methods can be used to value exotic barrier options with complex non-constant barriers.

This chapter also provides a review of the literature on hitting time distributions. This include a number of stochastic processes and several types of contour. The distribution that is important to this thesis is the hitting distribution of a Brownian motion to a linear barrier (which is equivalent to a hitting distribution of a geometric Brownian motion to an exponential barrier). The methods that will be developed later in this thesis will make use of this distribution.

In the next chapter, a new simulation method, based on hitting time simulation, for valuing exotic barrier options with complex non-constant barriers will be described.

Chapter 3

Valuing Exotic Barrier Options using the Contour Bridge Method

Barrier options with constant barriers are standard path-dependent options that are widely traded in financial markets. They are also used in debt instruments and in credit applications. The valuation problem of barrier options becomes more complicated when the barrier is non-constant. This feature is the focus of this chapter.

This chapter describes in detail a new simulation method, the contour bridge method, which can be applied to general barrier options, including options with highly non-linear barriers. It is based on bridge hitting times to amenable contours that can result in substantial improvements over existing simulation methods. Not only the method can be applied to non-constant barrier options, but it may also be applied to partial and discrete barrier options. However, those types of barriers are not considered in this chapter.

To implement the contour bridge method, one has to consider models and contours where the hitting time distribution is known and can be sampled. The two required distributions are

1. hitting times to suitable families of contours and

2. certain bridge hitting times.

These distributions can be found not only when the underlying process follows a geometric Brownian motion, but also when its process is variance-gamma or normal inverse Gaussian.

The contour bridge method is not restricted to valuing a single barrier option at a time. Because a set of hitting times to a sequence of contours can be generated, the method can be applied to value multiple barrier options simultaneously. Many of the methods mentioned in section 2.2 can be used to price only a single option at a time. This can be difficult when a large book of options is required. Even if a method can price a single option quite rapidly, its cost when applied to a book can be excessive.

The structure of the chapter is as follows. Section 3.1 reviews current simulation methods for barrier options. In section 3.2 the contour bridge method is presented. Different variants, tailored to specific situations, which can facilitate further speed-ups, are also discussed. Numerical results are presented in section 3.3. Finally, section 3.4 concludes.

3.1 Simulation methods for barrier options

This section reviews standard simulation methods for barrier options, namely, the Dirichlet and the Brownian bridge Monte Carlo methods. Note that the variance reduction techniques such as control variate or importance sampling will not be discussed. The effectiveness of these variance reduction techniques will be equal for both the standard simulation method and the contour bridge method. This implies that the advantage of the new method will persist when the variance reduction methods are implemented.

Let S_0 be an asset value at time t_0 and $b(0)$ be a barrier value at time t_0. In this chapter, the focus will be on the 'down'-options in which $S_0 > b(0)$. The application to the 'up'-option where $S_0 < b(0)$ can be extended by symmetry.

In this section the asset values are generated at a fixed set of times $\Upsilon = \{t_i\}_{i=0,\ldots,N}$, with $t_0 = 0$, $t_N = T$, and $t_i < t_{i+1}$ for all i. Write $b^i = b(t_i)$ for the barrier value at time t_i and S_j^i for the simulated value for time t_i on the jth sample path, with $S_j^0 \equiv S_0$.

3.1.1 Dirichlet Monte Carlo method

A plain and simple way to value a barrier option by using the Monte Carlo method is to use forward-evolution. The method generates a whole set of paths $S_j = \left\{S_j^0, \ldots, S_j^N\right\}_{j=1,\ldots,M}$, for times $t_i \in \Upsilon$. This can be done easily since the solu-

tion of the SDE in (1.0.1) is known. It is of the form

$$S_T = S_0 \exp\left(\left(r - \frac{1}{2}\sigma^2\right)T + \sigma z_T\right). \tag{3.1.1}$$

Since $z \sim N(0,T)$, where $N(\mu,\sigma^2)$ is the normal distribution with mean μ and variance σ^2, by introducing a standard normal random variable $\epsilon \sim N(0,1)$, S_T in (3.1.1) can be expressed as

$$S_T = S_0 \exp\left(\left(r - \frac{1}{2}\sigma^2\right)T + \sigma\sqrt{T}\epsilon\right). \tag{3.1.2}$$

Write $\Delta t_i = t_i - t_{i-1}$ and $S_j^0 = S_0\ \forall j = 1,\ldots,M$. On the jth path, S_j^i can be computed using

$$S_j^i = S_j^{i-1} \exp\left(\left(r - \frac{1}{2}\sigma^2\right)\Delta t_i + \sigma\sqrt{\Delta t_i}\epsilon_j^i\right), \tag{3.1.3}$$

for $i = 1,\ldots,N$. Then one can establish whether a barrier is hit at time t_i. The barrier is hit if $S_j^{i-1} > b^{i-1}$ but $S_j^i < b^i$. The plain method hitting time, τ_j^P, is set to be

$$\tau_j^P = \min\left\{t_i \in \Upsilon \mid S_j^i \leq b^i\right\}. \tag{3.1.4}$$

Unfortunately, the plain method is not accurate and is prone to severe simulation bias. In particular, if both S_j^i and S_j^{i+1} lie close to, but above, the barrier, then it is possible that the barrier was hit in the interval $[t_i, t_{i+1}]$, undetected by the method. Consequently the method tends to under-price knock-in barrier options and over-price knock-out options. This is referred to as the barrier breaching bias.

The Dirichlet method (Baldi et al. (1999) [20]) can be used to correct for simulation bias. The method makes use of the conditional hitting probability between each time interval $[t_i, t_{i+1}]$. On the jth sample path, for $S_j^i > b^i$ and $S_j^{i+1} > b^{i+1}$, let $p_j^i = \mathbb{P}\left[t_i < \tau < t_{i+1} \mid S_j^i, S_j^{i+1}\right]$ be the hitting probability in the interval $[t_i, t_{i+1}]$ conditional on the asset value taking values S_j^i and S_j^{i+1} at times t_i and t_{i+1}. If $S_j^i > b^i$ for all i, the probability that the barrier was hit in between simulated times on the jth path is given by

$$p_j = 1 - \prod_{i=0}^{N-1}\left(1 - p_j^i\right). \tag{3.1.5}$$

When p_j^i can be computed, for instance when S is a geometric Brownian motion and $b(t) \equiv b$ is a constant barrier, then p_j can be used to simulate the

intra-period hitting time, or to weight the payoff, to correct for simulation bias. To illustrate this, suppose one wants to value a plain vanilla knock-in call option that pays $(S_{t_N} - X)^+$ at time $t_N = T$ if a barrier $b(t)$ is hit any time up to maturity. Let p_j be the hitting probability on the jth path given by (3.1.5), then the knock-in payoff on the jth path is

$$(S_j^N - X)^+ p_j. \tag{3.1.6}$$

This is a very effective bias correction method.

When p_j^i cannot be computed, perhaps because the barrier is not constant, or the asset process is complex, it may be possible to approximate p_j^i, for instance by approximating $b(t)$ as piece-wise constant (or when the underlying asset process is geometric Brownian motion, piece-wise exponential-affine) over the intervals $[t_i, t_{i+1}]$.

3.1.2 Brownian bridge Monte Carlo method

The Brownian bridge method is a way to construct a sample path by using recursive bisection. Let ϵ_1 be the first standard normal variable. ϵ_1 is generally used to generate a sample point at final time $t_N = T$. Then the next normal variable is drawn to sample at an intermediate time step t_h conditional on a sample point at t_N and at t_0. Then sampling is made at further intermediate times. At step t_h, given z_{t_i} and z_{t_k} where $t_i < t_h < t_k$, ϵ_{t_h} is drawn and used to sample an asset value. The distribution of z_{t_h} is normal with mean

$$\mathbb{E}[z_{t_h} \mid z_{t_i}, z_{t_k}] = \left(\frac{t_k - t_h}{t_k - t_i}\right) z_{t_i} + \left(\frac{t_h - t_i}{t_k - t_i}\right) z_{t_k}, \tag{3.1.7}$$

and variance

$$\frac{(t_h - t_i)(t_k - t_h)}{t_k - t_i}. \tag{3.1.8}$$

Write S_{t_h} as the intermediate asset value with $t_i < t_h < t_k$. S_{t_h} can be sampled as follows.

$$S_{t_h} = S_{t_i}^c S_{t_k}^{1-c} \mathrm{e}^{\sigma\epsilon\sqrt{c(t_h - t_i)}}, \tag{3.1.9}$$

where $c = \frac{t_k - t_h}{t_k - t_i}$.

To use the Brownian bridge method to construct a path $(S^0, \ldots, S^N)$ at times $t_i \in \Upsilon$, one first constructs a value S^N for time t_N and then iteratively fills up the path in the order $S^{\lfloor N/2 \rfloor}$, $S^{\lfloor N/4 \rfloor}$, $S^{\lfloor 3N/4 \rfloor}$, $S^{\lfloor N/8 \rfloor}$, $S^{\lfloor 3N/8 \rfloor}$, *et cetera*, using a binary chop. It is straightforward to use the Brownian bridge when N is a power of two because one can divide an interval into two intervals with an integral number

of time steps.

In the Brownian bridge application, a hitting time can be constructed as follows. Given a value S^N for time t_N, one generates a value $S^{\lfloor N/2 \rfloor} \mid S^0, S^N$ by applying the Brownian bridge in (3.1.9). The one continues to generate values by binary chop, first to the left of the most recently generated value, and then to the right. At time t_i, when a value S^i is generated, one tests if S^i lies above or below the barrier. If it is below the barrier then no further iteration is required for times greater than t_i; to determine a hitting time, only asset values for times less than t_i need to be generated. If it is above the barrier then values both to the left and to the right of S^i must be generated and tested.

The computational saving in pruning away unnecessary refinements of intervals known not to contain the first hitting time can be considerable. This approach is related to a special case of the contour bridge method when contours are vertical (see section 3.2).

However, since both Dirichlet and Brownian bridge Monte Carlo generate hitting times only in the discrete set Υ, they do not value exotic barrier options rapidly. In contrast to these two standard methods, the contour bridge enables one to generate a more refined value of τ, with implicitly better sampling of both τ and ι, often with significantly faster computation times.

3.2 The contour bridge simulation method

A contour is defined as the image of a map $\bar{\beta} : \mathbb{R}^+ \longmapsto \mathbb{R}^+ \times \mathbb{R}^+$ where $\bar{\beta}$ is continuous, $\bar{\beta}(0) \in \{0\} \times \mathbb{R}^+$, and $\bar{\beta}$ is bijective onto its image.

In this chapter, contours of the form $\bar{\beta}(t) = (t, \beta(t))$ are considered. A contour $\bar{\beta}$ is determined by a choice of β and β is identified with $\bar{\beta}$. Throughout this chapter, β will be denoted as a contour. To define a vertical contour, let P be a set of points $(x, y) \in \mathbb{R}^2$ such that

$$\text{General contour:} \qquad P = \{(x, y) \mid y = \beta(x)\} \tag{3.2.1}$$

$$\text{Vertical contour:} \qquad P = \{(x, y) \mid x = t\}, \tag{3.2.2}$$

where t is a fixed time.

The option considered in this chapter is a 'down' option where the asset value at time t_0 is above the barrier at time t_0.

The method proceeds by constructing a series of hitting times to an indexed set of contours. The method is initialised by constructing a pair of contours that

bound the option barrier. Then intermediate contours are constructed. Hitting times to these contours are computed by sampling from the contour bridge density. The values of the asset at the hitting times are computed and compared to the values of the option barrier at those times, enabling efficient convergence to a hitting time to the option barrier.

The contour bridge method generalises the standard Brownian bridge Monte Carlo in two ways. First, hitting times are not constrained to lie in a pre-defined finite set Υ. This allows values of τ to be established with much greater refinement, resulting in reduced simulation bias. Second, by permitting a broader range of contours, very substantial efficiency gains can be achieved.

The method also incorporates a Dirichlet stopping condition, in which a conditional hitting probability is computed and then is used to test the exit condition (see section 3.2.4). This enables the method to also benefit from the advantages of the Dirichlet approach.

There are two main approaches to using the method which can be applied; (1) with single-hit contours and (2) with general contours. These approaches are discussed in section 3.2.6. There is a variant that can enhance the method's performance effectively. This variant can be applied when a pair of contours bracketing a hitting time $\tau < T$ has been found. This variant is referred to as the biggest-bite variate and will also be discussed.

In the next section, the general contour bridge method is described. Then an application to value both τ- and ι-options is given.

3.2.1 The choice of contour

The properties of the contour are given now. Given a model for the asset value process, suppose that, for this model, there exists an indexed set of contours,

$$\beta(t \mid \alpha) \in \mathbb{R}^+, \tag{3.2.3}$$

for t, $\alpha \in \mathbb{R}^+$, with the following four properties:

(i) **Ordered.** $\alpha_1 < \alpha_2 \rightarrow \beta(t \mid \alpha_1) > \beta(t \mid \alpha_2)\ \forall t \in \mathbb{R}^+$.

(ii) **Bounding.** There exist contours $\beta^\infty(t) = \beta(t \mid \alpha^\infty)$ and $\beta^0(t) = \beta(t \mid \alpha^0)$, with $S_0 \geq \beta^0(0)$, such that $\beta^0(t) > b(t) > \beta^\infty(t)\ \forall t \in [0, T]$.

(iii) **Hittable.** Let $\tau^\alpha \equiv \tau(S_0 \mid \alpha) = \min_t \{S_t < \beta(t \mid \alpha) \mid S_0\}$ be the hitting time to the contour $\beta(t \mid \alpha)$. It is necessary that τ^α can be sampled.

(iv) **Bridgeable.** Suppose $\alpha^0 < \alpha_1 < \alpha_2 < \alpha^\infty$, so that $\tau^{\alpha_1} < \tau^{\alpha_2}$. Given $\alpha_1 < \alpha < \alpha_2$, it must be possible to sample $\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}$. That is, conditional on τ^{α_1} and τ^{α_2}, it is possible to sample from the bridge distribution τ^α.

The illustration of the bridgeable property (property iv) is given in figure 3.1.

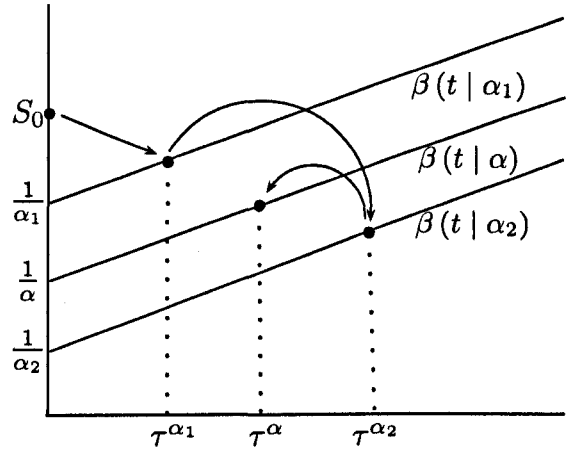


Figure 3.1. Illustration of a bridge hitting time $\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}$

It seems convenient to require $\mathbb{P}[\tau < \infty] = 1$. That is, the hitting time exists and is finite. Nonetheless, this requirement can be relaxed in some applications of the method (see section 3.2.2).

There are several families of contours that satisfy properties (i) - (iv). For example,

1. The asset price process is a geometric Brownian motion and contours are exponential

$$\beta(t \mid \alpha) = \frac{1}{\alpha} \exp(gt), \tag{3.2.4}$$

for some fixed growth rate g.

In this case the hitting time density (see section 3.2.2) is

$$f(t; \alpha, g, r, \sigma) = \frac{a}{\sigma\sqrt{2\pi t^3}} \exp\left(-\frac{(a - (g - \hat{\mu})t)^2}{2\sigma^2 t}\right), \tag{3.2.5}$$

where $\hat{\mu} = r - \frac{1}{2}\sigma^2$ and $a = \ln\left(\frac{S}{\beta(0)}\right) = \ln(S\alpha)$.

For $\tau^\alpha \in (\tau^{\alpha_1}, \tau^{\alpha_2})$, a bridge hitting time $\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}$ can be sampled. To do this, let τ^x be a hitting time from τ^{α_1} to τ^α, τ^y be a hitting time from τ^α to

τ^{α_2}, and τ^z be a hitting time from τ^{α_1} to τ^{α_2} so that $\tau^z = \tau^x + \tau^y$, where

$$\tau^z = \tau^{\alpha_2} - \tau^{\alpha_1} \tag{3.2.6}$$

$$\tau^y = \tau^{\alpha_2} - \tau^{\alpha}, \tag{3.2.7}$$

$$\tau^x = \tau^{\alpha} - \tau^{\alpha_1}. \tag{3.2.8}$$

The idea is to sample $\tau^x \mid \tau^z$ and then compute a bridge hitting time τ^α using relationship between $(\tau^{\alpha_1}, \tau^\alpha, \tau^{\alpha_2})$ and (τ^x, τ^y, τ^z). The bridge hitting density of $f(\tau^x \mid \tau^z)$ (see appendix A for the derivation) is

$$f(\tau^x \mid \tau^z) = \frac{1}{\sigma\sqrt{2\pi}} \frac{a_x a_y}{a_z} \left(\frac{\tau^x \tau^y}{\tau^z}\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{2\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)\right), \tag{3.2.9}$$

where, with a simplification, $a_x = \ln\left(\frac{\alpha}{\alpha_1}\right)$, $a_y = \ln\left(\frac{\alpha_2}{\alpha}\right)$, and $a_z = \ln\left(\frac{\alpha_2}{\alpha_1}\right)$.

2. The asset price process is a geometric Brownian motion and contours are horizontal (flat)

$$\beta(t \mid \alpha) \equiv \beta(0) = \frac{1}{\alpha}. \tag{3.2.10}$$

This is a limiting case of 1.

3. The asset price process is a geometric Brownian motion and contours are vertical. This is a standard Brownian bridge case.

4. The asset price process is variance gamma (VG) or normal inverse Gaussian (NIG) and contours are vertical. See Ribeiro and Webber (2003) [206] and Ribeiro and Webber (2004) [207] for the application to barrier and Asian options.

3.2.2 Hitting time sampling method

In this section, the hitting time simulation method is described. The variant of a method when $\mathbb{P}[\tau < \infty] < 1$ is discussed. The bridge hitting time sampling method is also discussed.

The hitting time distribution in (3.2.5) can be obtained from the hitting time distribution of a standard Brownian motion to a linear contour of a form $\beta^l(t) = a' + gt$. In this case, the density is given as (Lerche (1986) [164])

$$f(t; a') = \frac{a'}{\sqrt{2\pi t^3}} \exp\left(-\frac{(a' + gt)^2}{2t}\right), \tag{3.2.11}$$

Now for a Brownian motion with drift; $z_t = \mu t + \sigma w_t$ where w_t is a standard Brownian motion, the hitting distribution is

$$f(t; a', \mu, \sigma) = \frac{a'}{\sigma\sqrt{2\pi t^3}} \exp\left(-\frac{(a' + gt - \mu t)^2}{2\sigma^2 t}\right). \tag{3.2.12}$$

The distribution in (3.2.12) is equivalent to the hitting density of a geometric Brownian motion to an exponential contour. With $a' = \ln\left(\frac{1}{a}\right)$, one obtains the required hitting distribution

$$f\left(t; a, \hat{\mu}, \sigma\right) = \frac{a}{\sigma\sqrt{2\pi t^3}} \exp\left(-\frac{\left(a - (g - \hat{\mu})\, t\right)^2}{2\sigma^2 t}\right), \tag{3.2.13}$$

where $\hat{\mu}$ and a are given by (3.2.5).

3.2.2.1 Sampling from the hitting time distribution

The density in (3.2.13) is an inverse Gaussian distribution. The inverse Gaussian variable $X \sim IG(\nu, \lambda)$ has the canonical density $f(t; \nu, \lambda)$

$$f(t; \nu, \lambda) = \sqrt{\frac{\lambda}{2\pi t^3}} \exp\left(-\frac{\lambda(t - \nu)^2}{2\nu^2 t}\right). \tag{3.2.14}$$

With a parameterisation (Seshadri (1993) [224])

$$\nu = \frac{a}{\mu} \tag{3.2.15}$$

$$\lambda = \frac{a^2}{\sigma^2}, \tag{3.2.16}$$

where $\mu = g - \hat{\mu}$, One can see that (3.2.13) is an inverse Gaussian in (3.2.14). Since the inverse Gaussian variates can be sampled by using the transformation method of Michale et al. (1976) [186] (MSH method), the hitting time with densities (3.2.13) can be sampled.

Suppose $Y = y(X)$ where the first derivative y' of y exists, is continuous, and is non-zero, except on a closed set with probability zero. Suppose for a fixed $v = y(x_i), i = 1, \ldots, N$. The MSH method samples X by first drawing v for Y and then selecting the ith root x_i of v with probability $p_i(v)$ where

$$p_i(v) = \left(1 + \sum_{j=1, j\neq i}^{N} \left|\frac{y'(x_i)}{y'(x_j)}\right| \cdot \frac{f\left(x_j\right)}{f\left(x_i\right)}\right)^{-1} \tag{3.2.17}$$

For an inverse Gaussian distribution in (3.2.14), one has

$$y(x) = \frac{\lambda(x-\nu)^2}{x\nu^2} \sim \chi_1^2 \tag{3.2.18}$$

The two roots of (3.2.18) are

$$x_1 = \nu + \frac{\nu^2 y}{2\lambda} - \frac{\nu}{2\lambda}\sqrt{4\nu\lambda y + \nu^2 y^2}, \tag{3.2.19}$$

$$x_2 = \frac{\nu^2}{x_1}. \tag{3.2.20}$$

From (3.2.20) one has

$$\frac{y'(x_1)}{y'(x_2)} = -\left(\frac{\nu}{x_1}\right)^2 \tag{3.2.21}$$

$$\frac{f(x_2)}{f(x_1)} = \left(\frac{x_1}{\nu}\right)^3. \tag{3.2.22}$$

Then

$$p_1(v) = \frac{\nu}{\nu + x}. \tag{3.2.23}$$

The MSH algorithm proceeds as follows.

1. Generate a random variable $y \sim \chi_1^2$.
2. Compute the two roots x_1 and x_2 from (3.2.19) and (3.2.20).
3. Generate a random variable $u \sim U(0,1)$.
4. If $u \leq \frac{\nu}{\nu+x}$, return x_1.
5. Else, return x_2.

Unfortunately, this algorithm works only for the case where one has drift $\nu \geq 0$; that is, the process drifts toward the contour ($g > \mu$). To extend to the case $\nu < 0$ where the process drifts away from the contour ($g < \mu$), one can make use of the following relationship (Atiya and Metwally (2005) [15])

$$f(t; -\nu, \lambda) = f(t; \nu, \lambda) \exp\left(\frac{-2\lambda}{\nu}\right), \tag{3.2.24}$$

where $f(\nu, \lambda; x)$ is the inverse Gaussian density in (3.2.14).

However, as mentioned in Atiya and Metwally (2005) [15] the area under the function $f(-\nu, \lambda; x)$ in (3.2.24) is not equal to one because when the process drifts

away from a contour, there is a possibility that the contour may not ever be hit. The plot of hitting time densities with different ν is shown in figure 3.2.

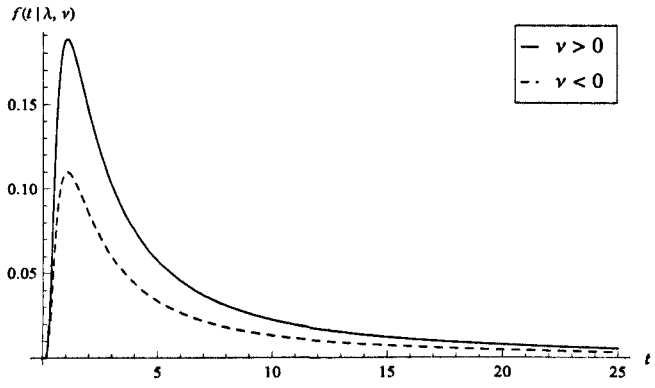


Figure 3.2. Hitting time densities where $\nu > 0$ and $\nu < 0$

Parameter values in figure 3.2 are $S_0 = 100$, $b(0) = 70$, $\sigma = 0.2$, $r = 0.05$. Hitting time densities are computed by using (3.2.5). The first density is plotted by using the contour growth rate $g = 0.06$, which yields $\nu = 11.89$. This corresponds to a case where the process drifts toward a contour and hence, by using a numerical integration, an area under the curve is equal to 1. The second density is plotted by using the contour growth rate $g = 0$. In this case, $\nu = -11.88$ and the process drifts away (upward) from a flat contour. One can see that a density where $\nu < 0$ is much smaller than that of $\nu > 0$. A numerical integration gives an area under the dotted line curve to be $\sim$0.59.

Importantly, one cannot simply implement the standard acceptance and rejection method in this case where $f(\tau; \nu, \lambda)$ in (3.2.14) is the dominating density. This is because the resulting variates τ would still come from the Inverse Gaussian distribution which integrates to one. In other words, by simply implementing a usual acceptance and rejection method, one will obtain more (finite) hitting time variates than is supposed to, hence resulting in a bias in option values.

To overcome this problem, Beskos and Roberts (2005) [34] suggested that, from (3.2.24), an asset value process will eventually hit a contour with probability $\mathbb{P}[\tau \in (0, \infty)] = \exp\left(\frac{-2\lambda}{\nu}\right)$, and then

$$\tau_{(0<\tau<\infty)} \sim IG(\nu, \lambda). \tag{3.2.25}$$

That is, conditioning on the contour being hit, the hitting time distribution is $IG(\nu, \lambda)$. Hence, to sample from a density $f(-\nu, \lambda; x)$, one samples the hitting

time variates with probability $\exp\left(\frac{-2\lambda}{\nu}\right)$ and then set variates $\tau = \tau_\infty$ with a probability $1-\exp\left(\frac{-2\lambda}{\nu}\right)$, reflecting the fact that there is non-zero probability of a contour not being hit.
The adjusted sampling algorithm proceeds as follows.

1. Generate random variable $\tau \sim IG(|\nu|, \lambda)$ by using MSH method.
2. Generate random variable $u \sim U(0,1)$.
3. If $u \leq \frac{f(-\nu,\lambda,\tau)}{f(\nu,\lambda,\tau)} = \exp(-2\lambda/\nu)$ then return τ,
4. If not, set $\tau = \infty$ so that the contour is not hit.

Step 4 means a contour is not hit. Note that one is interested only whether (and when) a contour is hit before option maturity T. Whether or when a contour is hit after time T is not taken into the computation procedure.

3.2.2.2 Sampling from the bridge distribution

In order to sample from the inverse Gaussian bridge distribution in (3.2.9), one uses the Tweedie's Theorem in Seshadri (1993) [224] and the result of Ribeiro and Webber (2003) [206]. Write $\text{IG}(\nu, \lambda)$ as an inverse gaussian variable with parameters ν and λ. Using (3.2.15) and (3.2.16), one writes $\text{IG}\left(\frac{a}{\mu}, \frac{a^2}{\sigma^2}\right) = \text{IG}(\nu, \lambda)$. Tweedie's Theorem and the result from Ribeiro and Webber (2003) [206] are shown below.

Theorem 1 (Tweedie's Theorem with $n = 2$). *Suppose* $\tau^x \sim IG\left(\frac{a_x}{\mu}, \frac{a_x^2}{\sigma^2}\right)$, $\tau^y \sim IG\left(\frac{a_y}{\mu}, \frac{a_y^2}{\sigma^2}\right)$ *and* $\tau^z \sim IG\left(\frac{a_z}{\mu}, \frac{a_z^2}{\sigma^2}\right)$ *with* $\tau^z = \tau^x + \tau^y$ *where* τ^x *and* τ^y *are independent then*

$$Q = \frac{1}{\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right) \tag{3.2.26}$$

is chi-squared with one degree of freedom. That is, $Q \sim \chi_1^2$.

Ribeiro and Webber (2003) [206] employed Monte Carlo simulation method to value path dependent options in Normal Inverse Gaussian (NIG) model. They implemented an inverse Gaussian bridge method to construct sample paths. They made use of Tweedie's Theorem and was able to sample from an inverse Gaussian bridge distribution which is similar to (3.2.9). The result of Ribeiro and Webber (2003) [206] is given in Theorem 3.2.2.2.

Theorem 2 (Ribeiro and Webber (2003) [206]'s result). *The bridge hitting time density*

$$f\left(\tau^x \mid \tau^z\right) = \frac{1}{\sigma\sqrt{2\pi}} \frac{a_x a_y}{a_z} \left(\frac{\tau^x \tau^y}{\tau^z}\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{2\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)\right), \quad (3.2.27)$$

where, $a_x = \ln\left(\frac{\alpha}{\alpha_1}\right)$, $a_y = \ln\left(\frac{\alpha_2}{\alpha}\right)$, *and* $a_z = \ln\left(\frac{\alpha_2}{\alpha_1}\right)$, *can be sampled by the MSH method of Michale et al. (1976) [186].*

Proof. Let $q = \frac{1}{\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)$ be an exponent in equation (3.2.27). Set $s = \frac{\tau^y}{\tau^x}$, $\hat{\lambda} = \frac{a_y^2}{\sigma^2 \tau^z}$ and $\hat{\nu} = \frac{a_y}{a_x}$. Then one has

$$q = \hat{\lambda}\frac{(s-\hat{\nu})^2}{s\hat{\nu}^2} \equiv g(s) \sim \chi_1^2. \quad (3.2.28)$$

The two solutions to (3.2.28) are

$$s_1 = \hat{\nu} + \frac{\hat{\nu}^2 q}{2\hat{\lambda}} - \frac{\hat{\nu}}{2\hat{\lambda}}\sqrt{4\hat{\nu}\hat{\lambda}q + \hat{\nu}^2\hat{\lambda}^2 q^2} \quad (3.2.29)$$

$$s_2 = \frac{\hat{\nu}^2}{s_1}. \quad (3.2.30)$$

Now change variables in (3.2.27) to $s = \frac{\tau^y}{\tau^x} \mid \tau^{\alpha_2}$, the density of s is

$$f(s;\hat{\lambda},\hat{\nu}) = \sqrt{\frac{\hat{\lambda}}{2\pi}}\frac{1}{1+\hat{\nu}} s^{-\frac{3}{2}}(1+s)\exp\left(-\frac{1}{2}\hat{\lambda}\frac{(s-\hat{\nu})^2}{s\hat{\nu}^2}\right). \quad (3.2.31)$$

The MSH result now can be applied. Let $Q = g(S)$ and suppose for a fixed $q = g(s_i)$ for $i = 1, \ldots, N$. In this case (3.2.17) becomes

$$p_i(q) = \left(1 + \sum_{j=1, j\neq i}^{K} \left|\frac{g'(s_i)}{g'(s_j)}\right| . \frac{f(s_j)}{f(s_i)}\right)^{-1}. \quad (3.2.32)$$

The ratio of first derivative of $g(s)$ and of $f(s)$ can be computed from (3.2.28) and (3.2.31). By substituting (3.2.30) and (3.2.29), one obtains

$$\frac{g'(s_1)}{g'(s_2)} = -\left(\frac{\hat{\nu}}{s_1}\right)^2 \quad (3.2.33)$$

$$\frac{f(s_2)}{f(s_1)} = \frac{s_1^2}{\hat{\nu}^3}\frac{\hat{\nu}+s_1}{1+s_1} \quad (3.2.34)$$

Hence, the smaller root, s_1, should be chosen with probability

$$p_1(q) = \frac{\hat{\nu}(1+s_1)}{(1+\hat{\nu})(\hat{\nu}+s_1)}. \tag{3.2.35}$$

□

By Theorem 3.2.2.2, one essentially samples $s = \frac{\tau^y}{\tau^x}$. Then a bridge hitting time $\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}$ can be computed by using (3.2.6) to (3.2.8). To write s in term of τ^α, one has

$$s = \frac{\tau^y}{\tau^x} \tag{3.2.36}$$

$$s = \frac{\tau^{\alpha_2} - \tau^\alpha}{\tau^\alpha - \tau^{\alpha_1}} \tag{3.2.37}$$

$$s+1 = \frac{\tau^{\alpha_2} - \tau^{\alpha_1}}{\tau^\alpha - \tau^{\alpha_1}} \tag{3.2.38}$$

$$\tau^\alpha = \tau^{\alpha_1} + \frac{\tau^{\alpha_2} - \tau^{\alpha_1}}{s+1}. \tag{3.2.39}$$

The algorithm to sample a bridge hitting time $\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}$ is as follows.

1. Generate a random variable $q \sim \chi^2$.
2. Compute the roots s_1 and s_2 in (3.2.29) and (3.2.30) respectively.
3. Generate a random variable $u \sim U(0,1)$.
4. If $u \leq \frac{\hat{\nu}(1+s_1)}{(1+\hat{\nu})(\hat{\nu}+s_1)}$, set $s = s_1$.
5. Else set $s = s_2$.
6. Set $\tau^\alpha = \tau^{\alpha_1} + \frac{\tau^{\alpha_2} - \tau^{\alpha_1}}{s+1}$.

3.2.3 The contour bridge method algorithm

In this section the contour bridge method is described in detail. First, the algorithm to compute a hitting time τ to the option barrier, or to exit if no $\tau < T$ is found, is described. If only ι is required, the algorithm can be modified. For the case of an ι-option which requires the value of S_T, section 3.2.5 discusses how a value for $S_T \mid \iota$ can be simulated.

Write $\hat{\tau}$ for a simulated first hitting time of S to the option barrier. The algorithm proceeds as follows:

1. Given a pair of bounding contours, $\beta^0(t) = \beta(t \mid \alpha^0)$ and $\beta^\infty(t) = \beta(t \mid \alpha^\infty)$; construct hitting times τ^0 and τ^∞ to these contours.

   If $\tau^0 > T$ then the option barrier is not hit and the algorithm stops. If $\tau^\infty < T$ then $\hat{\tau} \in (\tau^0, \tau^\infty)$ and if an ι-option is being valued the algorithm exits.

   Otherwise set $i = 1$, $\bar{\tau}^i = \tau^0$, $\underline{\tau}^i = \tau^\infty$. Apply step 2 to the interval $(\bar{\tau}^i, \underline{\tau}^i)$.

2. At the ith step, suppose that one has hitting times $\bar{\tau}^i < \underline{\tau}^i$ to bounding contours $\bar{\beta}^i(t) = \beta(t \mid \bar{\alpha}^i)$ and $\underline{\beta}^i(t) = \beta(t \mid \underline{\alpha}^i)$, $\bar{\alpha}^i < \underline{\alpha}^i$, with $S(\bar{\tau}^i) = \beta(\bar{\tau}^i \mid \bar{\alpha}^i) > b(\bar{\tau}^i)$, and one has to determine whether there is a hitting time in the interval $(\bar{\tau}^i, \underline{\tau}^i)$.

   (2.a) Test if the algorithm halts (stopping rules are discussed in section 3.2.4).

   If the algorithm halts and $S(\underline{\tau}^i) = \beta(\underline{\tau}^i \mid \underline{\alpha}^i) < b(\underline{\tau}^i)$ then the option barrier must be hit in the interval $(\bar{\tau}^i, \underline{\tau}^i)$. Return an approximation $\hat{\tau} = \frac{1}{2}(\bar{\tau}^i + \underline{\tau}^i)$ to the hitting time; otherwise if $S(\underline{\tau}^i) > b(\underline{\tau}^i)$ return a condition indicating that no hitting time has been found.

   If the algorithm does not halt then iterate. Set $\alpha^{i+1} = \frac{1}{2}(\bar{\alpha}^i + \underline{\alpha}^i)$ and construct τ^{i+1} as the hitting time to $\beta(t \mid \alpha^{i+1})$ conditional on $\bar{\tau}^i$ and $\underline{\tau}^i$.

   (2.b) If $\tau^{i+1} > T$ then the option barrier cannot be hit in the interval $(\tau^{i+1}, \underline{\tau}^i)$. Set $\bar{\tau}^{i+1} = \bar{\tau}^i$, $\underline{\tau}^{i+1} = \tau^{i+1}$, increment $i \leftarrow i + 1$ and apply step 2 to the new interval $(\bar{\tau}^i, \underline{\tau}^i)$.

   (2.c) If $S(\tau^{i+1}) = \beta(\tau^{i+1} \mid \alpha^{i+1}) < b(\tau^{i+1})$ then the option barrier has been hit in the interval $(\bar{\tau}^i, \tau^{i+1})$. If a bare-$\iota$ option is being valued the algorithm exits immediately.

   Otherwise set $\bar{\tau}^{i+1} = \bar{\tau}^i$, $\underline{\tau}^{i+1} = \tau^{i+1}$, increment $i \leftarrow i + 1$ and apply step 2 to the new interval $(\bar{\tau}^i, \underline{\tau}^i)$.

   (2.d) If $S(\tau^{i+1}) = \beta(\tau^{i+1} \mid \alpha^{i+1}) > b(\tau^{i+1})$ then establish whether the option barrier may be hit in $(\bar{\tau}^i, \tau^{i+1})$ or in $(\tau^{i+1}, \underline{\tau}^i)$.

   i. If there exists $t \in [\bar{\tau}^i, \tau^{i+1}]$ such that $\beta(t \mid \alpha^{i+1}) < b(t)$ then it is possible that the option barrier is hit in the interval $(\bar{\tau}^i, \tau^{i+1})$. Set $(\bar{\tau}^{i+1}, \underline{\tau}^{i+1}) = (\bar{\tau}^i, \tau^{i+1})$, increment $i \leftarrow i + 1$, and apply step 2 to the new interval $(\bar{\tau}^i, \underline{\tau}^i)$. If this finds a first hitting time $\hat{\tau} \in (\bar{\tau}^i, \underline{\tau}^i)$ return $\hat{\tau}$ as the first hitting time in the interval $(\bar{\tau}^{i-1}, \underline{\tau}^{i-1})$.

   ii. If step 2(d)i fails to return a hitting time, then check if it is possible that the option barrier is hit in the interval $(\tau^{i+1}, \underline{\tau}^i)$.

If $\tau^{i+1} > T$, or if $\beta\left(t \mid \underline{\alpha}^i\right) > b(t)$ for all $t \geq \tau^{i+1}$, then the option barrier cannot be hit in this interval. Otherwise set $\left(\bar{\tau}^{i+1}, \underline{\tau}^{i+1}\right) = \left(\tau^{i+1}, \underline{\tau}^i\right)$, increment $i \leftarrow i+1$, and apply step 2 to the new interval $\left(\bar{\tau}^i, \underline{\tau}^i\right)$. If this finds a hitting time $\hat{\tau} \in \left(\bar{\tau}^i, \underline{\tau}^i\right)$ return $\hat{\tau}$ as the first hitting time in the interval $\left(\bar{\tau}^{i-1}, \underline{\tau}^{i-1}\right)$.

If steps 2(d)ii also fails to return a hitting time then the algorithm returns a value indicating that no hitting time exists in the interval $\left(\bar{\tau}^{i-1}, \underline{\tau}^{i-1}\right)$.

The contour bridge algorithm is illustrated in figure 3.3.

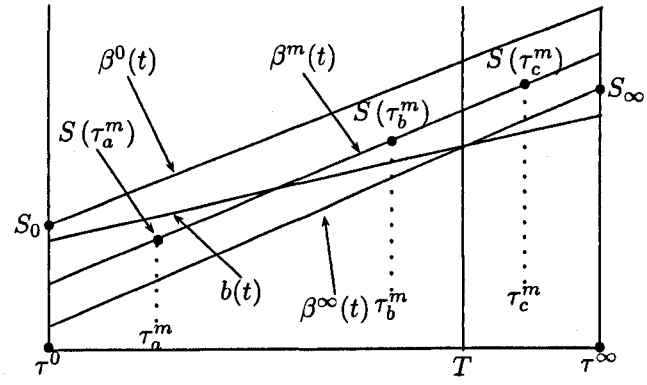


Figure 3.3. Contour bridge algorithm illustration

In figure 3.3, bounding contours are $\beta^0(t)$ and $\beta^\infty(t)$. In this case, $\tau^0 < T < \tau^\infty$. Then one sets $\bar{\tau}^1 = \tau^0$ and $\underline{\tau}^1 = \tau^\infty$. The intermediate contour $\beta^m(t)$ is constructed such that $\bar{\beta}^1(t) > \beta^m(t) > \underline{\beta}^1(t)\ \forall t > 0$. Figure 3.3 shows three scenarios of a bridge hitting time; τ_a^m, τ_b^m and τ_c^m. First, τ_a^m corresponds to step (2c) in which $S(\tau_a^m) < b(\tau_a^m)$. In this case, the algorithm will be applied only on the interval $\left(\bar{\tau}^1, \tau_a^m\right)$. Second, τ_c^m corresponds to step (2b) in which $\tau_c^m > T$. In this case, since the option barrier cannot be hit from a time beyond τ_c^m, the algorithm will be applied to the interval $\left(\bar{\tau}^1, \tau_c^m\right)$. Third, τ_b^m corresponds to step (2d) where $\tau_b^m < T$ and $S(\tau_b^m) > b(\tau_b^m)$. In this case, the option barrier can be hit either on a left hand side or a right hand side of (τ_b^m). The algorithm will have to be applied in the interval $\left(\bar{\tau}^1, \tau_b^m\right)$ (step 2(d)i) or in the interval (τ_b^m, T) (step 2(d)ii).

3.2.4 The stopping conditions

The algorithm halts if one of two conditions is true. It stops if

1. $\left|\bar{\tau}^i - \underline{\tau}^i\right| < \varepsilon_\tau^1$, or
2. if $S\left(\bar{\tau}^i\right)$ and $S\left(\underline{\tau}^i\right)$ both lie above the barrier, then if $\left|\bar{\tau}^i - \underline{\tau}^i\right| < \varepsilon_\tau^2$ compute (or approximate) $p = \mathbb{P}\left[\bar{\tau}^i < \tau < \underline{\tau}^i \mid S\left(\bar{\tau}^i\right),\ S\left(\underline{\tau}^i\right)\right]$. Stop if $p < \varepsilon_p$.

ε_τ^1 is the binding condition when a hitting time $\tau \in (\bar{\tau}^i, \underline{\tau}^i)$ has been found. In the numerical example, ε_τ^1 is set to be 10^{-10}. This degree of refinement is possible in a Dirichlet method only if $T/\varepsilon_\tau^1 = T \times 10^{10}$ time steps are used. Usually, it might not be possible to work with such a very large number.

The second condition is introduced to enable the algorithm to exit early if the probability of hitting the barrier is sufficiently low. Without this condition, the algorithm will be inefficient. In particular, the algorithm will always take T/ε_τ^1 steps to establish that the barrier was not hit.

When working with exotic barriers, p is not usually computable and an approximation is used. Suppose that ε_τ^2 is small enough so that over an interval $\varsigma = [t, t+\varepsilon_\tau^2]$ the option barrier can be approximated reasonably well by a contour, $b(t) \sim \beta(t \mid \alpha; g)$ for $t \in \varsigma$. This is a legitimate assumption if, for instance, over ς both the option barrier and a contour are approximately linear. Write τ^m for the conditional hitting time to the contour, and set $p^m = \mathbb{P}\left[t < \tau^m < t+\varepsilon_\tau^2 \mid S(t),\ S\left(t+\varepsilon_\tau^2\right)\right]$. When p^m can be computed, one can approximate p with p^m.

p^m can be computed when S is a geometric Brownian motion and contours are exponential. Suppose one is given values s_0 and s_1 of a geometric Brownian motion S at times t_0 and t_1, $t_0 < t_1$, and values β_0 and β_1 of an exponential contour $\beta(t \mid \alpha)$ also for times t_0 and t_1. Suppose that $s_i > \beta_i$, $i = 0, 1$. The conditional hitting probability $p = \mathbb{P}[t_0 < \tau < t_1 \mid s_0, s_1, \beta_0, \beta_1]$ of S to β is known (see Baldi (1999) [19]). Set $u_i = \ln\left(\frac{s_i}{\beta_i}\right)$ and let

$$g = \frac{1}{t_1 - t_0} \ln\left(\frac{\beta_1}{\beta_0}\right) \tag{3.2.40}$$

be the growth rate of $\beta(t \mid \alpha)$. Then

$$p = \exp\left(-\frac{2}{\sigma^2} u_0 \left(\frac{u_1}{t_1 - t_0} - g\right)\right) \tag{3.2.41}$$

$$= \exp\left(-\frac{2}{\sigma^2 (t_1 - t_0)} u_0 v_{01}\right) \tag{3.2.42}$$

where $v_{01} = \ln\left(\frac{s_1}{\beta_0}\right)$.

If ε_p is too large the method would encounter simulation bias. However, numerical results show that. even with a small value of ε_p, the associated stopping condition very significantly reduces execution time. ε_τ^2 can be set to a value much larger than ε_τ^1. If the barrier is not hit, in the worse case the method now takes only T/ε_τ^2 steps before halting.

3.2.4.1 Intersection times

In steps 2(d)i and 2(d)ii it is necessary to know the first and last intersection times I^L and I^R between the method and option barriers in the interval $(\bar{\tau}^i, \underline{\tau}^i)$,

$$I^L = \min_{t\in(\bar{\tau}^i,\underline{\tau}^i)} \{\beta(t) = b(t)\}, \tag{3.2.43}$$

$$I^R = \max_{t\in(\bar{\tau}^i,\underline{\tau}^i)} \{\beta(t) = b(t)\}. \tag{3.2.44}$$

A knowledge of I^L, I^R, for each contour, and whether $\beta(\bar{\tau}^i)$ and $\beta(\underline{\tau}^i)$ are above or below $b(\bar{\tau}^i)$ and $b(\underline{\tau}^i)$, is sufficient to establish whether iterations are needed to the left or to the right of a hitting time $\tau \in (\bar{\tau}^i, \underline{\tau}^i)$. Tables 3.2.45 illustrate. 'L' means that iteration is required on $(\bar{\tau}^i, \tau)$. 'R' means that it is required on $(\tau, \underline{\tau}^i)$.

Left-hand side conditions		$\tau > I^L$	
		true	false
$\beta(\bar{\tau}^i) > b(\bar{\tau}^i)$	true	L, R	R
	false	L, R	L

Right-hand side conditions		$\tau > I^R$	
		true	false
$\beta(\underline{\tau}^i) > b(\underline{\tau}^i)$	true	L,R	L, R
	false	L	L, R

(3.2.45)

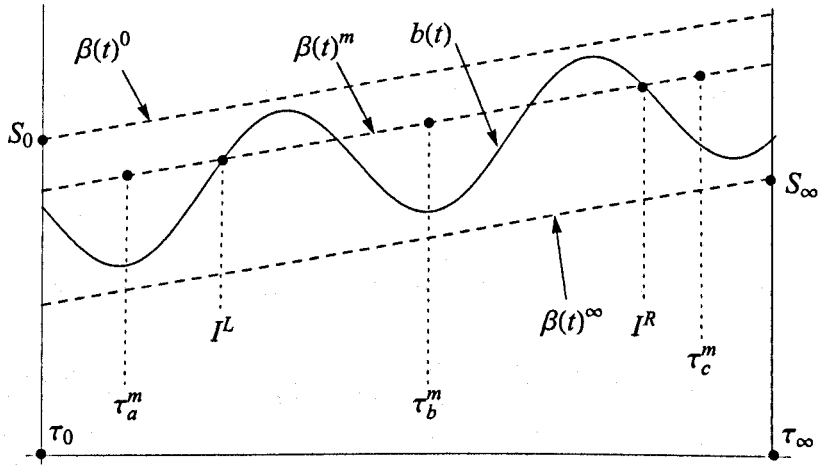


Figure 3.4. Intersection time illustration

Figure 3.4 illustrates intersection times I^L and I^R. Bounding contours are $\beta^0(t)$ and $\beta^\infty(t)$ and $\tau^0 < \tau^\infty < T$. The intermediate contour is $\beta^m(t)$. If a bridge hitting time to $\beta^m(t)$, $\tau^m = \tau_a^m$, one refines only on the right of τ^m. This decision can be made because one knows that $\tau^m < I^L$ and $\beta(\bar{\tau}^i) > b(\bar{\tau}^i)$ and hence the option cannot be hit before τ^m. If a bridge hitting time $\tau^m = \tau_b^m$, then because

of $I^L < \tau^m < I^R$ and $\beta\left(\bar{\tau}^i\right) > b\left(\bar{\tau}^i\right)$, one refines both on the left and the right hand sides of τ^m. If $\tau^m = \tau_c^m$, one has $\tau^m > I^R$ and $\beta\left(\underline{\tau}^i\right) > b\left(\underline{\tau}^i\right)$. In this case one also refines both on the left and the right hand sides of τ^m because the option barrier can be hit on either one of the sides. The knowledge of I^L and I^R enables one to determine if a bridge hitting time is τ_a^m, τ_b^m, or τ_c^m. Unfortunately it may be expensive to compute I^L and I^R. This issue will be addressed in section 3.2.6.

3.2.5 Computing the value of S_T

A bare-ι option does not require a value of the asset price at maturity to be generated. For a general ι-option, such as a knock option, this is not the case. Indeed the payoff to a general barrier option may depend on both values of τ and S_T, or on the value of S_t for some $t \in [0, T]$. It is easy to extend the contour bridge method to simulate values of S_t for some value of t consistent with the values generated for τ^i. Although only the case when $t = T$ is considered, it will be clear how the method applies for other times t. To compute the value of S_T, there are three cases to consider:

Case 1. $\tau^\infty < T$

In this case, $\hat{\tau} \in \left(\tau^0, \tau^\infty\right)$, and thus a value for S_T can be found directly from $S\left(\tau^\infty\right) = \beta\left(\tau^\infty \mid \alpha^\infty\right)$. If the asset process SDE has a solution, the value may be generated in a single step from τ^∞ to T, otherwise a short-step simulation can be used.

Case 2. $\tau^0 < T < \tau^\infty$

If the algorithm establishes that $\tau < T$, then S_T can be sampled from the conditional distribution $S_T \mid S_\tau$. Otherwise, when the algorithm exits it will have generated a set of hitting times

$$\tau^0 < \ldots < \tau^L < T < \tau^R < \ldots < \tau^\infty \tag{3.2.46}$$

where τ^L is the largest hitting time less than T and τ^R is the smallest hitting time greater than T. Suppose that τ^L is the hitting time to the contour $\beta\left(t \mid \alpha^L\right)$ and τ^R to $\beta\left(t \mid \alpha^R\right)$.

One needs to generate a value $S_T \mid \tau^L, \tau^R$. If this conditional distribution is known, then S_T can be sampled directly, and no further computation is required. If this is not possible then a following numerical approximation can be used:

Numerical approximation of S_T:

1. Let $\alpha^m = \frac{1}{2}\left(\alpha^L + \alpha^R\right)$ and compute a conditional hitting time τ^m to $\beta\left(t \mid \alpha^m\right)$.

2. If $\tau^m > T$, one sets $\tau^R = \tau^m$ and repeats step 1 on the new $\left(\tau^L, \tau^R\right)$ pair.

3. if $\tau^m < T$, one sets $\tau^L = \tau^m$ and goes to step 1.

4. The algorithm exits when $\left|\tau^L - \tau^R\right| < \varepsilon_\tau^1$, and one sets $S_T = \frac{1}{2}\left(S\left(\tau^L\right) + S\left(\tau^R\right)\right)$.

Case 3. $T < \tau^0$

In this case, assume that one may find a value α^0 such that $S_0 = \beta\left(0 \mid \alpha^0\right)$. Then the previous method case can be applied to the pair $\left(\tau^L, \tau^R\right) = \left(0, \tau^0\right)$.

When implementing, if contours are shallow, with small gradients, the iterative procedure previously described may stall. When two shallow contours are close together, it is possible that the asset value process hits an intermediate contour very close to one of the end points, $\bar{\tau}^i$ or $\underline{\tau}^i$. The hitting density becomes heavily bimodal with very little likelihood of the next hitting time occurring towards the centre of the interval. Figure 3.5 illustrates this.[1]

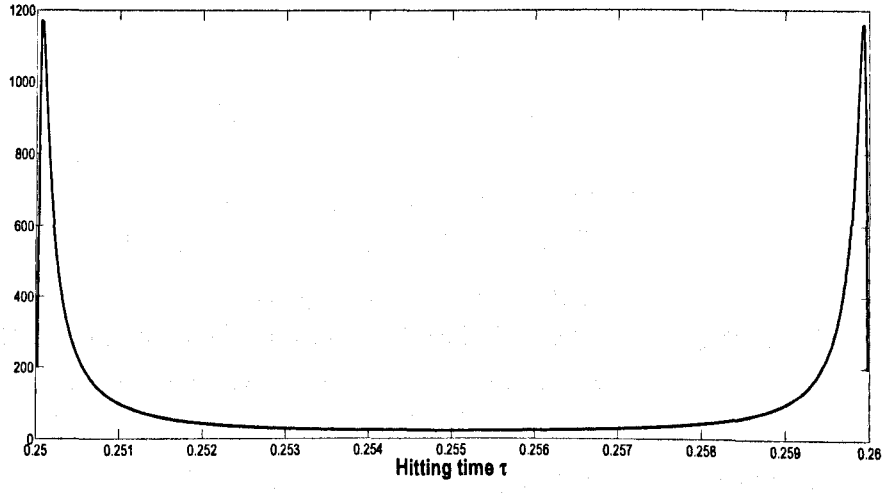


Figure 3.5. Bimodal bridge hitting time density; $f\left(\tau^\alpha \mid \tau^{\alpha_1}, \tau^{\alpha_2}\right)$. $\tau^{\alpha_1} = 0.25$, $\tau^{\alpha_2} = 0.26$, $\frac{1}{\alpha_1} = 70.2$, $\frac{1}{\alpha} = 70$, $\frac{1}{\alpha_2} = 69.8$

One can see that, from figure 3.5, there is a high probability that a bridge hitting time variate τ will be close to either $\bar{\tau}$ or $\underline{\tau}$ ($\bar{\tau} = \tau^{\alpha_1}$ and $\underline{\tau} = \tau^{\alpha_2}$). As a

[1]Numerical integration gives an area under the curve of $\sim$1.

result, as the method evolves, the contours get very close together but the values $\bar{\tau}^i$ and $\underline{\tau}^i$ may stay far apart. This is a numerical problem. Theoretically the method should still converge to a true option barrier hitting time τ. However, practically the method gets bogged down because of rounding errors.

The problem particularly has an impact on simulating for S_T. Option values may be heavily biased. Even though one can use steeper contours to avoid the problem, the computational times will increase and the efficiency gains will be reduced. To get rid of a bias, one can first compute the hitting time τ, even if only ι is required. That is, if one establishes that $\tau < T$, $S_T \mid S_\tau$ can be simulated directly by using (3.1.3). Although this method gives an unbiased option value, and can be used to value knock-in options effectively, it works only when S_T needs to be computed when $\tau < T$. Consequently, it cannot be applied directly to knock-out options.

However, even though one cannot value a knock-out option directly from the contour bridge method, one can compute a value of knock-out options from a value of knock-in options computed from the method by using in-out parity. In-out parity states that the combination of knock-in and knock-out option results in a corresponding European option. In-out parity holds for a European barrier option regardless of the shape of a barrier. Therefore, since the method can value knock-in options accurately (see numerical results in section 3.3), values of knock-out option can be obtained.

3.2.6 Single-hit method barriers

In the contour bridge algorithm, the most computationally intensive step is to calculate the first and last intersection times I^L and I^R. If no explicit solution to their values is available, one is likely to have to conduct an expensive search to find them. Although this search can be accelerated, by tracking previously found first and last intersection times, it is a potentially very expensive step.

The method can be simplified if the contours $\beta(t \mid \alpha)$ can be chosen such that for all $\alpha \in [\alpha^0, \alpha^\infty]$ each contour intersects the option barrier at most once. In this case the conditions in steps 2(d)i and 2(d)ii can be determined from a knowledge of $\beta(\bar{\tau}^i)$, $\beta(\tau)$ and $\beta(\underline{\tau}^i)$, and $b(\bar{\tau}^i)$, $b(\tau)$ and $b(\underline{\tau}^i)$, without having to compute I^L and I^R.

This is referred to as the single-hit version of the contour bridge method. Since the single-hit condition may be satisfied only if contours are steep enough, the efficiency gain may be reduced. Vertical contours are obviously always single-hit. However, numerical results in section 3.3 suggest that that they are inefficient.

To obtain a sufficient condition for single-hit contours, the minimum slope of each contour must be greater than the maximum slope of the option barrier. For an exponential contour,

$$\beta(t \mid \alpha; g) = \frac{1}{\alpha} e^{gt}, \tag{3.2.47}$$

fix $\beta_T \leq b(T)$ and suppose that β_T is a maximal value such that a lower bounding contour can be found that takes value β_T at time T, so that $\beta(T \mid \alpha^\infty; g) = \beta_T$ for some index α^∞. One has

$$\beta_T = \frac{1}{\alpha^\infty} e^{gT}, \tag{3.2.48}$$

$$g = \frac{1}{T} \ln(\alpha^\infty \beta_T). \tag{3.2.49}$$

Write $\alpha^\infty = \alpha$, differentiating (3.2.47) with respect to t yields $\frac{g}{\alpha^\infty} e^{gt}$. The minimum slope at $t = 0$ is

$$\frac{g}{\alpha^\infty}, \tag{3.2.50}$$

for $g > 0$. Substituting the value of g in (3.2.49) into (3.2.50), for all $\alpha \in [\alpha^0, \alpha^\infty]$, this family of contours has minimum slope

$$d_{\min} = \frac{1}{\alpha^\infty T} \ln(\alpha^\infty \beta_T). \tag{3.2.51}$$

If the maximum slope of the option barrier in $[0, T]$ is greater than $d_{\min}$ a single-hit contour bridge version is not possible.

3.2.7 The biggest-bite variant

This is an improved version of the single-hit method. Suppose that on an interval $(\bar{\tau}, \underline{\tau})$, with $\underline{\tau} < T$, one has $S(\bar{\tau}) > b(\bar{\tau})$ and $S(\underline{\tau}) < b(\underline{\tau})$ so that $\tau \in (\bar{\tau}, \underline{\tau})$.

An interval $(\bar{\tau}, \underline{\tau})$ with these properties is referred to as a bracketing interval. Suppose the current bounding contours are $\beta(t \mid \bar{\alpha})$ and $\beta(t \mid \underline{\alpha})$, where $\bar{\alpha} < \underline{\alpha}$. The biggest-bite variant algorithm proceeds as follows:

1. Set $\bar{d} = S(\bar{\tau}) - b(\bar{\tau})$ and $\underline{d} = b(\underline{\tau}) - S(\underline{\tau})$.
2. If $\bar{d} > \underline{d}$ choose α such that $\beta(\bar{\tau} \mid \alpha) = b(\bar{\tau})$. Otherwise choose α such that $\beta(\underline{\tau} \mid \alpha) = b(\underline{\tau})$.

Figure 3.6 illustrates the biggest-bite variate. A choice whether $\beta^{i+1}(t) = \beta_a^{i+1}(t)$ or $\beta^{i+1}(t) = \beta_b^{i+1}(t)$ depends on the quantities $\bar{d}$ and $\underline{d}$.

The biggest-bite method essentially introduces a new selection method for intermediate contours once a bracketing interval $(\bar{\tau}, \underline{\tau}) \subseteq [0, T]$ has been found. The

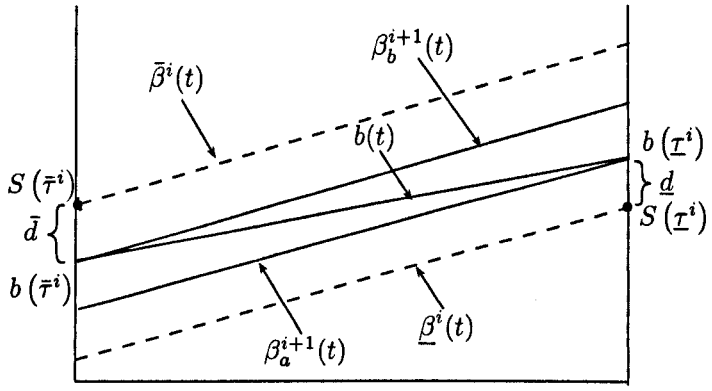


Figure 3.6. Biggest bite illustration

method constructs a sequence of hitting times to a series of contours so that at each step the algorithm reduces the range $(\bar{\alpha}, \underline{\alpha})$ by the greatest possible amount while preserving the bracketing property.

3.2.8 The vertical contours variant

In this case contours are vertical, and thus the method is equivalent to the Brownian bridge method discussed in section 3.2 and generating a hitting time to a vertical contour will be equivalent to generating an asset value itself by using a bridge method.

The main disadvantage of this method is that since the contours are vertical, the range $(\bar{\tau}, \underline{\tau})$ will not be reduced substantially as it is in the single-hit or the biggest-bite variant case. Therefore, it is expected not to perform very well compared with exponential contours.

However, in situations where the single-hit or the biggest-bite variate cannot be applied, for example those where hitting time densities cannot be sampled, the vertical contour variant may be applied. This is because one is required to sample $S_T \mid S_0$, and for $t \in (t_1, t_2)$ to sample $S_t \mid S_{t_1}, S_{t_2}$. Hence, one may work with a broader range of models when using the vertical contour method.

3.2.9 Valuation of a book of options

The contour bridge method can be extended to the valuation problem of a book of exotic barrier options simultaneously. Each option can has different barriers and different strikes. To value a book of options, the method must have core contours that are common for every option. The overview of the method is as follows.

1. Construct bounding contours by choosing α^0 and α^∞ such that they are single-hit to all option's barrier.

2. Construct a sequence of N intermediate single hit contours $\{\beta(t \mid \alpha^i)\}_{i=1,\ldots,N}$ and hitting times to these contours.

3. Then the method generates refinements in an interval only if there is an option in a book whose barrier might be hit in that interval.

To save computational times, sample paths must be reused. Although there are additional costs because refinements are needed for more intervals, these costs are cheaper than those from valuing each option separately. This is an advantage over PDE methods. In particular, with different strikes and barriers, PDE methods can only to be used to value one option at at time, and a computational cost can increase significantly if a book of option is required.

The application of the contour bridge method to value a book of options is not discussed in this thesis.

3.3 Numerical results

In this section, the method is implemented. First, a benchmark option is valued and then a set of exotic barrier options are valued. The benchmark option is a plain vanilla flat barrier option.

To assess the performance of the method relative to the standard method, comparisons are made with a Dirichlet method (as described on page 23) and with a Brownian bridge method. Most comparisons are made with a moderate level of volatility, but low and high volatility cases are also presented. Barrier options are also valued when S_0 is close to or far from the barrier.

All options are 'down'-style, so that $S_0 > b(0)$. Base-case parameter values for the asset process are $S_0 = 85$, $r = 0.05$, $\sigma = 0.2$. All options have time to maturity $T = 1$.

The tables report five quantities. They are (1) the option value $\widehat{v}$, (2) the Monte Carlo standard error (se), (3) the computational time in seconds [t], (4) the bias {b} and (5) the efficiency gain **e**.

The bias is computed as

$$\{\mathrm{b}\} = \frac{\widehat{v} - v}{se}, \tag{3.3.1}$$

where v is the true option value. It measures how many standard errors the Monte Carlo value is away from the true value. Values of {b} in the range ± 2 denote no bias (to some confidence level). Large absolute values of {b} indicate the presence of bias in a method.

The efficiency gain of method x over method y is $\mathbf{e}^{xy}$,

$$\mathbf{e}^{xy} = \frac{(se^y)^2 t^y}{(se^x)^2 t^x}, \tag{3.3.2}$$

where se^x and se^y are the standard errors of method x and method y, computed with computational times t^x and t^y. $\mathbf{e}^{xy}$ is the multiple of the time taken by method x for method y to achieve the same standard error. A gain of less then 1 indicates that method x is less efficient than method y. Efficiency gains in the tables are with respect to the standard Dirichlet method, reported with an efficiency gain of 1.

3.3.1 The benchmark option

First, a benchmark barrier option is valued to confirm the accuracy of the new method, and to measure its performance compared with existing alternative methods. The option, that is valued, is a vanilla first touch rebate option, $\mathcal{O}^R$. This pays a fixed rebate of R at the hitting time $\tau \equiv \tau^b$ to a flat barrier, $b(t) \equiv b$, conditional on $\tau \leq T$.

When the underlying asset follows a geometric Brownian motion, there is an explicit formula for $\mathcal{O}^R$(Reiner and Rubinstein (1991) [205]). Set

$$z = \frac{1}{\sigma\sqrt{T}} \ln\left(\frac{b}{S_0}\right) + \delta\sigma\sqrt{T}, \tag{3.3.3}$$

then

$$\mathcal{O}^R(S_0) = R\left[\left(\frac{b}{S_0}\right)^{\psi+\delta} \mathrm{N}(z) + \left(\frac{b}{S_0}\right)^{\psi-\delta} \mathrm{N}\left(z - 2\delta\sigma\sqrt{T}\right)\right],$$

where

$$\begin{aligned} \psi &= \frac{1}{\sigma^2}\left(r - \frac{1}{2}\sigma^2\right), \\ \delta &= \sqrt{\psi^2 + \frac{2r}{\sigma^2}}. \end{aligned}$$

Numerical results are reported in tables 3.1, 3.2 and 3.3. The option has barrier level $b = 80$, and rebate $R = 5$, maturing at $T = 1$. The asset process has $S_0 \in \{80.5, 85, 100, 115\}$ with $r = 0.05$, with $\sigma = 0.1$ (table 3.1), $\sigma = 0.2$ (table 3.2),

and $\sigma = 0.3$ (table 3.3). The values of the explicit solutions are displayed beneath the asset value.

Every simulated value uses $M = 10^6$ sample paths. The contour methods use tolerances $\varepsilon_\tau^1 = 10^{-10}$, $\varepsilon_\tau^2 = 10^{-10}$, and $\varepsilon_p = 10^{-10}$. The standard Dirichlet method uses $N = 1000$ time steps.

Tables 3.1, 3.2 and 3.3 report option values computed by four methods: the standard Dirichlet method; the contour bridge method with vertical contours; the single-hit contour bridge method; and the biggest-bite variant. For the single-hit method and the biggest-bite variant, four different values of the growth rate g are used. $\beta(T \mid \alpha^\infty; g) = 80$ in every case and $\beta(0 \mid \alpha^\infty; g) = 75$, 50, 20, 1, respectively. As the value of g increases, contours are steeper.

The entries in the tables show the option value produced by each method, the standard error in round brackets, the computation time in square brackets, the bias in curly brackets, and the efficiency gain in bold. Values other than the option value are rounded to two significant figures.

The results in the tables 3.1, 3.2 and 3.3 show no evidence of bias. All biases are within the range ± 2. Standard errors are constant across all methods. As expected, the vertical contour method's performance is worst in all cases. Even at its best, when S_0 is far from the barrier, it is only around two thirds the efficiency of the standard Dirichlet method. Except when S_0 is close to the barrier the contour methods always produce substantial efficiency gains.

When S_0 is close to the barrier the single-hit method does not perform well; specially when the contours are steeper. This is because, when S_0 is close to the barrier, the Diriclet method will use very small computational times because there is high probability that the barrier will be hit. It has an in-built advantage when only a single option is being valued. If a book of options were being valued, so that the Dirichlet method would usually need to generate an entire sample path, the contour bridge method would maintain substantial efficiency gains. Nonetheless, even with S_0 is close to the barrier, the biggest-bite method still produces gains when contours are shallow. The gains in this case range from $\sim$6 to 11.

In all cases the contour bridge method exhibits increasing efficiency gains as the contours become shallower. This is because shallow contours enable the hitting time to be found faster than steep contours. Efficiency gains increase as the initial asset moves further away from the barrier, to a maximum of 480 when $\sigma = 0.1$, 360 when $\sigma = 0.2$, and 240 when $\sigma = 0.3$. When comparing the single-hit method with the the biggest-bite variate, it is clear that gains from the biggest-bite variant are always significantly larger than those from the single-hit method. This emphasises

S_0, exact.	Standard Dirichlet	Vertical contour variant	Single-hit contour bridge method				Biggest-bite variant			
			g				g			
			0.0645	0.4700	1.3863	4.3820	0.0645	0.4700	1.3863	4.3820
80.5 4.5873	4.5899 (0.0014) [69] {1.9} **1**	4.5888 (0.0014) [320] {1.1} **0.21**	4.5879 (0.0014) [36] {0.44} **1.9**	4.5864 (0.0014) [46] {-0.7} **1.5**	4.5876 (0.0014) [67] {0.20} **1.0**	4.5891 (0.0014) [110] {1.28} **0.62**	4.5866 (0.0014) [6.3] {-0.57} **11**	4.5868 (0.0014) [16] {-0.4} **4.3**	4.5879 (0.0014) [28] {0.40} **2.5**	4.5872 (0.0014) [61] {-0.08} **1.1**
85 1.9600	1.9571 (0.0024) [360] {-1.2} **1**	1.9625 (0.0024) [680] {1.0} **0.53**	1.9618 (0.0024) [23] {0.78} **16**	1.9598 (0.0024) [48] {-0.10} **7.4**	1.9617 (0.0024) [98] {0.71} **3.6**	1.9594 (0.0024) [240] {-0.24} **1.5**	1.9611 (0.0024) [3.5] {0.48} **100**	1.9604 (0.0024) [13] {0.2} **29**	1.9577 (0.0024) [23] {-0.93} **15**	1.9609 (0.0024) [58] {0.39} **6.2**
100 0.0418	0.0415 (0.0004) [480] {-0.68} **1**	0.0410 (0.0004) [740] {-1.8} **0.65**	0.0416 (0.0004) [9.4] {-0.50} **51**	0.0423 (0.0004) [20] {1.1} **23**	0.0423 (0.0004) [47] {1.1} **10.0**	0.0416 (0.0004) [120] {-0.34} **4.0**	0.0420 (0.0005) [1.1] {0.57} **430**	0.0412 (0.0004) [3.5] {-1.3} **140**	0.0418 (0.0004) [6.5] {-0.04} **73**	0.0417 (0.0004) [21] {-0.31} **23**
115 0.00024	0.00025 (0.00003) [470] {0.25} **1**	0.00021 (0.00003) [730] {-0.93} **0.76**	0.00023 (0.00003) [7.8] {-0.30} **66**	0.00029 (0.00004) [17] {1.27} **25**	0.00025 (0.00003) [32] {0.26} **15**	0.00026 (0.00004) [86] {0.51} **5.3**	0.00020 (0.00003) [1.0] {-1.3} **480**	0.00026 (0.00004) [1.6] {0.64} **280**	0.00022 (0.00003) [4.1] {-0.46} **130**	0.00024 (0.00003) [11] {-0.02} **44**

Table 3.1. Benchmark valuation, rebate, flat barrier. $\sigma = 0.1$

S_0, exact.	Standard Dirichlet	Vertical contour variant	Single-hit contour bridge method				Biggest-bite variant			
			g				g			
			0.0645	0.4700	1.3863	4.3820	0.0645	0.4700	1.3863	4.3820
80.5 4.8458	4.8460 (0.0008) [33] {0.26} **1**	4.8462 (0.0008) [280] {0.41} **0.12**	4.8470 (0.0009) [35] {1.4} **0.96**	4.8465 (0.0008) [39] {0.8} **0.90**	4.8460 (0.0008) [48] {0.26} **0.70**	4.8463 (0.0008) [66] {0.60} **0.50**	4.8458 (0.0009) [4.8] {0.04} **6.9**	4.8459 (0.0008) [11] {0.1} **2.9**	4.8468 (0.0008) [20] {1.2} **1.7**	4.8460 (0.0008) [39] {0.25} **0.85**
85 3.5932	3.5934 (0.0022) [210] {0.11} **1**	3.5953 (0.0022) [510] {0.95} **0.41**	3.5912 (0.0022) [30] {-0.89} **6.9**	3.5949 (0.0022) [43] {0.79} **4.8**	3.5917 (0.0022) [69] {-0.69} **3.0**	3.5943 (0.0022) [140] {0.50} **1.54**	3.5943 (0.0022) [4.2] {0.51} **50**	3.5929 (0.0022) [12] {-0.14} **17**	3.5953 (0.0022) [19] {0.96} **11**	3.5929 (0.0022) [42] {-0.12} **5.0**
100 1.0812	1.0802 (0.0020) [430] {-0.48} **1**	1.0835 (0.0020) [730] {1.1} **0.59**	1.0837 (0.0020) [19] {1.2} **22**	1.0810 (0.0020) [30] {-0.09} **14**	1.0784 (0.0020) [56] {-1.4} **7.60**	1.0813 (0.0020) [130] {0.03} **3.3**	1.0809 (0.0020) [2.0] {-0.15} **218**	1.0836 (0.0020) [5.8] {1.2} **73**	1.0807 (0.0020) [10] {-0.24} **42**	1.0825 (0.0020) [26] {0.66} **16**
115 0.2537	0.2531 (0.0011) [460] {-0.54} **1**	0.2546 (0.0011) [750] {0.84} **0.62**	0.2543 (0.0011) [12] {0.55} **40**	0.2551 (0.0011) [20] {1.3} **24**	0.2546 (0.0011) [37] {0.87} **12**	0.2525 (0.0011) [92] {-1.1} **5.1**	0.2548 (0.0011) [1.3] {1.0} **360**	0.2541 (0.0011) [2.3] {0.35} **210**	0.2541 (0.0011) [5] {0.40} **85**	0.2532 (0.0011) [14] {-0.48} **33**

Table 3.2. Benchmark valuation, rebate, flat barrier. $\sigma = 0.2$

S_0, exact.	Standard Dirichlet	Vertical contour variant	Single-hit contour bridge method				Biggest-bite variant			
			g				g			
			0.0645	0.4700	1.3863	4.3820	0.0645	0.4700	1.3863	4.3820
80.5	4.9116	4.9101	4.91170	4.9120	4.9121	4.9120	4.9109	4.9128	4.9117	4.9113
4.9114	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)	(0.0006)
	[22]	[260]	[33]	[35]	[40]	[52]	[4.2]	[9.6]	[16]	[33]
	{0.31}	{-2.1}	{0.39}	{0.9}	{0.98}	{0.92}	{-0.82}	{2.1}	{0.38}	{-0.15}
	1	**0.08**	**0.67**	**0.60**	**0.56**	**0.43**	**5.4**	**2.4**	**1.4**	**0.68**
85	4.1553	4.1521	4.1572	4.1560	4.1548	4.1545	4.1537	4.1526	4.1551	4.1533
4.1546	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)	(0.0018)
	[140]	[430]	[30]	[43]	[56]	[100]	[4.0]	[11]	[16]	[33]
	{0.39}	{-1.3}	{1.4}	{0.77}	{0.11}	{-0.04}	{-0.45}	{-1.08}	{0.29}	{-0.69}
	1	**0.32**	**4.5**	**3.2**	**2.4**	**1.4**	**34**	**12**	**8.5**	**4.2**
100	2.2052	2.2036	2.2067	2.2076	2.2099	2.2085	2.2082	2.2100	2.2094	2.2121
2.2087	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)	(0.0024)
	[350]	[660]	[26]	[39]	[59]	[110]	[2.7]	[6]	[13]	[26]
	{-1.4}	{-2.1}	{-0.80}	{-0.44}	{0.52}	{-0.07}	{-0.19}	{0.56}	{0.30}	{1.4}
	1	**0.53**	**13**	**9.0**	**5.9**	**3.1**	**130**	**58**	**27**	**13**
115	1.0762	1.0792	1.0757	1.0797	1.0737	1.0767	1.0763	1.0767	1.0745	1.0774
1.0773	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)
	[440]	[720]	[19]	[26]	[44]	[96]	[1.8]	[3.9]	[7]	[17]
	{-0.54}	{0.96}	{-0.76}	{1.22}	{-1.8}	{-0.26}	{-0.48}	{-0.26}	{-1.4}	{0.05}
	1	**0.61**	**23**	**16**	**9.8**	**4.5**	**240**	**110**	**60**	**25**

Table 3.3. Benchmark valuation, rebate, flat barrier. $\sigma = 0.3$

the effectiveness of the fast-bracketing interval discussed in section 3.2.7.

As the volatility gets larger, efficiency gains tend to reduce. This is because a higher volatility implies a higher hitting probability for the Dirichlet method which will exit early. However, even with the high volatility case $\sigma = 0.3$, efficiency gains with the contour bridge method remain significant.

3.3.2 The exotic options

In this section, exotic barrier options are valued. There are three types of options and four types of barriers.

Option types

The three option types are:

1. A vanilla rebate option, $\mathcal{O}^R$. This pays a rebate of R at the hitting time $\tau \equiv \tau^b$ to the barrier $b(t)$, conditional on $\tau \leq T$.

2. Knock-in and knock-out options. The payoff h_T at the maturity time T is, for example,

$$H_T = \begin{cases} (X - S_T)^+ \mathbb{I}_{\tau \leq T}, & \text{for a knock-in put, } \mathcal{O}^{IP}, \\ (S_T - X)^+ \mathbb{I}_{T < \tau}, & \text{for a knock-out call, } \mathcal{O}^{OC}, \end{cases} \tag{3.3.4}$$

   for some strike X, where $\tau \equiv \tau^b$ is the hitting time to the barrier $b(t)$, and $\mathbb{I}$ is an indicator function.

3. A recovery option, $\mathcal{O}^{rec}$. This models the recovery part paid to a bond holder if a firm defaults when its asset value, S_t, hits a barrier before time T. Recovery values are paid at two times. At the time τ of default an amount $D_\tau = \max(p_\tau S_\tau, R_\tau)$ is paid, where p_τ and R_τ are a time dependent proportion and rebate respectively. At the final time T a second recovery value of $D_T = \max(p_T S_T, R_T)$ is paid. The present value of the cash received along the jth sample path is c_j,

$$c_j = \left(\mathrm{e}^{-r\tau_j} D_\tau + \mathrm{e}^{-rT} D_T\right) \mathbb{I}_{\tau \leq T}. \tag{3.3.5}$$

   In the numerical examples, $p_\tau \equiv 0.1$, for all $\tau < T$, $p_T = 0.05$ and $R_\tau = R_T = 5$.

Barrier types

There are four different types of barrier. They are linear, bull spread, concave quadratic, and sine barriers. The motivation for using these barrier is to test a numerical efficiency of the contour bridge method. Options that are traded in markets have a much simpler barrier. The idea is that if the contour bridge method can value options with difficult barriers, it can easily be used to price options with simpler barriers.

The four barrier types are described as follows.

1. A linear barrier, b_t^{lin},

$$b_t^{\text{lin}} = l + mt. \tag{3.3.6}$$

   Two linear barriers are considered: an increasing linear barrier, $m > 0$, and a decreasing linear barrier, $m < 0$.

2. A bull spread barrier, b_t^{bs},

$$b_t^{\text{bs}} = l + (u - l)\,\mathrm{N}\left(w\left(t - c\right)\right),\ u > l \tag{3.3.7}$$

   where N(.) is the standard normal distribution function.

3. A quadratic concave barrier, b_t^{c},

$$b_t^{\text{c}} = a + bt + ct^2. \tag{3.3.8}$$

4. An linear sine barrier, b_t^{ls},

$$b_t^{\text{ls}} = l + mt + a\sin\left(b\left(t + c\right)\right), \tag{3.3.9}$$

   Two cases are considered: an increasing linear sine barrier, $m > 0$, and a decreasing linear sine barrier, $m < 0$.

The four types of barrier types together with the parameters are summarised in table 3.4. Note that barrier parameters are chosen such that b_0 is less than $S_0(= 85)$ for a base case and that a barrier neither rises too sharply nor falls too sharply during a life of an option. In both cases, the contour bridge and the benchmark method will exit early and hence one might not be able to obtain a meaningful comparison.

Five of these six barriers are plotted in Table 3.5, together with bounding pairs of exponential contours.

Types	Functional form	Parameters
b_t^{lin}	$l + mt$	$\{l, m\} = \{80, 10\}$ and $\{80, -10\}$
b_t^{bs}	$l + (u - l)\,\text{N}(w(t - c))$	$\{l, u, w, c\} = \{80, 90, 8, 0.4\}$
b_t^{c}	$a + bt + ct^2$	$\{a, b, c\} = \{80, 30, -30\}$
b_t^{ls}	$l + mt + a\sin(b(t + c))$	$\{l, m, a, b, c\} = \{79, 10, 5, 15, 0.2\}$
		$\{l, m, a, b, c\} = \{79, -10, 5, 15, 0.2\}$

Table 3.4. Exotic barrier types and parameters

3.3.3 Numerical results

Since there is no explicit solution for the value of options with the barriers shown in table 3.4, benchmark values for options with these barriers were obtained using a standard Dirichlet Monte Carlo method with 10^8 sample paths and up to 1024 time steps. Computation times were up to 15 hours for each value. In the majority of cases it is clear, by comparing values computed with 256 and 512 time steps, that the quoted values are unbiased within standard error.[2]

Unless otherwise stated the contour bridge method uses $\varepsilon_\tau^1 = 10^{-10}$, $\varepsilon_\tau^2 = 10^{-4}$ and $\varepsilon_p = 10^{-10}$. All (non-benchmark) results are obtained with $M = 10^6$ sample paths; the Dirichlet method uses $N = 512$ times steps.

3.3.3.1 The base case

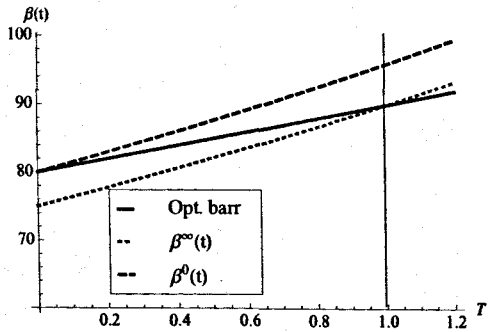
This section presents results for the base case, with $S_0 = 85$ and $\sigma = 0.2$, for the rebate $\mathcal{O}^R$, the knock-in call, $\mathcal{O}^{IC}$, the knock-in put, $\mathcal{O}^{IP}$, and the recovery option $\mathcal{O}^{rec}$, for the six option barriers discussed in section 3.3.2. Results are reported in tables 3.6, 3.7, 3.8 and 3.9. The tables show numerical results for the single-hit contour bridge method, the biggest bite variant and the standard Dirichlet method.

Values of the contour growth rate g are chosen such that they are the smallest values for which the single-hit contour bridge method is possible. α^0 is chosen so that $\beta(0 \mid \alpha^0; g) = S_0$; α^∞ is chosen so that $\beta(0 \mid \alpha^\infty; g) = b(T)$.

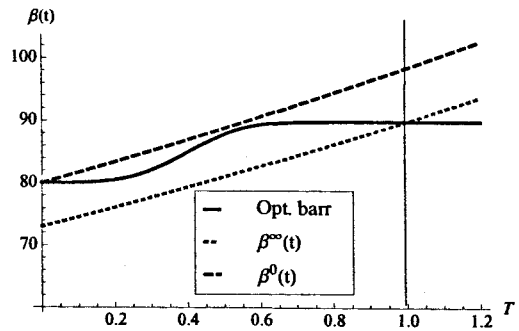
Tables 3.6, 3.7, 3.8 and 3.9 suggest that there is no bias in barrier option values since the biases are all within ± 2. The performance of the vertical contour variant is worst than that of the Dirichlet method, producing gains of only at best $\sim\frac{1}{4} < 1$. Results for the vertical variant are shown only for the rebate option.

The single-hit method achieves the highest efficiency gains when the barrier is linear, particularly with a negative slope. The gains are ~ 8. When a more complex

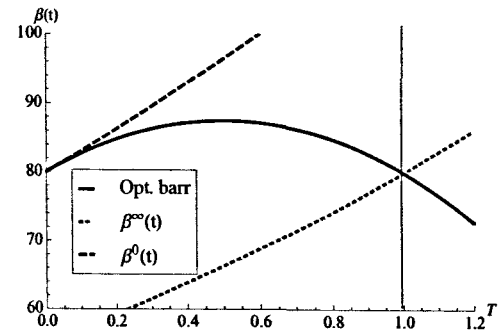
[2] In other cases, notably the decreasing linear sine case, the result may still be biased by an amount greater than the standard error.



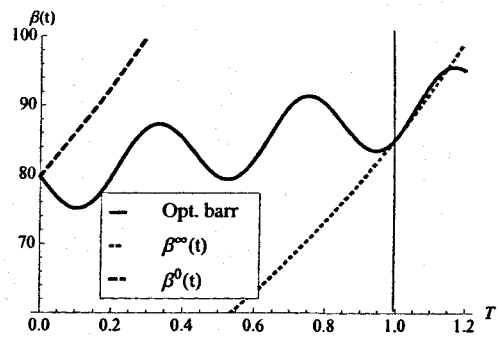
(a) Linear barrier, $\frac{1}{\alpha^\infty} = 75$, $\frac{1}{\alpha^0} = 80$



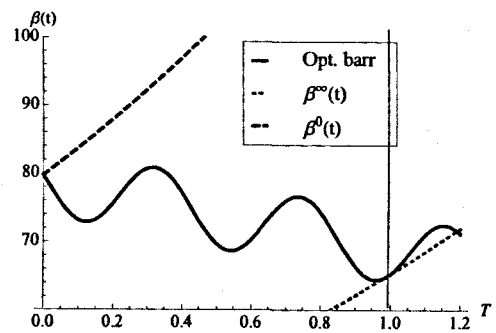
(b) Bull spread, $\frac{1}{\alpha^\infty} = 73$, $\frac{1}{\alpha^0} = 80$ barrier



(c) Concave quadratic barrier, $\frac{1}{\alpha^\infty} = 55$, $\frac{1}{\alpha^0} = 80$



(d) Increasing sine barrier, $\frac{1}{\alpha^\infty} = 40$, $\frac{1}{\alpha^0} = 79.7$



(e) Decreasing sine barrier, $\frac{1}{\alpha^\infty} = 30$, $\frac{1}{\alpha^0} = 80$

Table 3.5. Non-constant barriers

$S_0 = 85$, $\sigma = 0.2$		Barrier type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	4.23219	2.77827	4.26814	4.38986	4.13732	2.62100
	(se)	(0.00017)	(0.00025)	(0.00017)	(0.00016)	(0.00018)	(0.00025)
Standard	v^d	4.2328	2.7763	4.2658	4.3909	4.1347	2.6197
Dirichlet	(se^d)	(0.0017)	(0.0046)	(0.0017)	(0.0016)	(0.0018)	(0.0025)
	$[t^d]$	[90]	[150]	[95]	[67]	[109]	[170]
	$\{\mathrm{b}^d\}$	{-0.36}	{-0.42}	{-1.4}	{0.65}	{-1.5}	{-0.52}
Vertical	v^B	4.2309	2.7791	4.2691	4.3912	4.1402	2.6198
contour	(se^B)	(0.0017)	(0.0025)	(0.0017)	(0.0016)	(0.0018)	(0.0025)
variant	$[t^B]$	[390]	[526]	[530]	[432]	[591]	[652]
	$\{\mathrm{b}^B\}$	{-0.76}	{0.35}	{0.56}	{0.81}	{-0.72}	{-0.47}
	e^B	**0.23**	**0.29**	**0.18**	**0.16**	**0.08**	**0.26**
Contour	g	0.1304	0.0741	0.4055	0.3930	1.2149	1.1824
method	v^s	4.2311	2.7804	4.2666	4.3891	4.1374	2.6189
(single-hit, τ)	(se^s)	(0.0017)	(0.0025)	(0.0017)	(0.0016)	(0.0018)	(0.0025)
	$[t^s]$	[16]	[19]	[39]	[22]	[54]	[56]
	$\{\mathrm{b}^s\}$	{-0.7}	{0.9}	{-0.91}	{-0.47}	{0.04}	{-0.84}
	e^s	**5.7**	**7.9**	**2.4**	**3.0**	**2.0**	**3.0**
Contour	v^b	4.2314	2.7767	4.2675	4.3899	4.1347	2.6207
method	(se^b)	(0.0017)	(0.0025)	(0.0017)	(0.0016)	(0.0018)	(0.0025)
(biggest-bite)	$[t^b]$	[3.8]	[7]	[19]	[9.0]	[22]	[19]
	$\{\mathrm{b}^b\}$	{-0.46}	{-0.63}	{-0.38}	{0.03}	{-1.5}	{-0.12}
	e^b	**24**	**21**	**5.0**	**7.4**	**5.0**	**8.9**

Table 3.6. Rebate option $\mathcal{O}^R$, $\sigma = 0.2$

$S_0 = 85$, $\sigma = 0.2$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	1.33384	0.62779	1.29762	1.80679	1.17181	0.37943
	(se)	(0.00048)	(0.00033)	(0.00046)	(0.00057)	(0.00043)	(0.00024)
Standard	v^d	1.3312	0.6252	1.2923	1.7976	1.1710	0.3794
Dirichlet	(se^d)	(0.0047)	(0.0034)	(0.0046)	(0.0057)	(0.0043)	(0.0024)
	$[t^d]$	[89]	[151]	[95]	[67]	[109]	[167]
	$\{\mathrm{b}^d\}$	{-0.56}	{-0.76}	{-1.2}	{-1.6}	{-0.19}	{-0.03}
Contour	g	0.1304	0.0741	0.4055	0.393	1.1249	1.1824
method	v^s	1.3343	0.6258	1.2996	1.8051	1.1696	0.3764
(single-hit, τ)	(se^s)	(0.0048)	(0.0033)	(0.0046)	(0.0057)	(0.0043)	(0.0024)
	$[t^s]$	[16]	[18]	[39]	[22]	[55]	[56]
	$\{\mathrm{b}^s\}$	{0.10}	{-0.59}	{0.43}	{-0.30}	{-0.51}	{-1.2}
	e^s	**5.6**	**8.4**	**2.4**	**3.0**	**2.0**	**3.0**
Contour	v^b	1.3293	0.6299	1.2931	1.8042	1.1658	0.3814
method	(se^b)	(0.0048)	(0.0033)	(0.0046)	(0.0057)	(0.0043)	(0.0024)
(biggest-bite)	$[t^b]$	[4.0]	[7.5]	[20]	[9.1]	[22]	[19]
	$\{\mathrm{b}^b\}$	{-0.94}	{0.63}	{-0.97}	{-0.45}	{-1.4}	{0.83}
	e^b	**22**	**20**	**4.8**	**7.4**	**5.0**	**8.8**

Table 3.7. Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.2$

$S_0 = 85$, $\sigma = 0.2$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	13.2477	10.6085	13.2682	13.1423	13.1621	10.4940
	(se)	(0.0012)	(0.0013)	(0.0012)	(0.0012)	(0.0012)	(0.0013)
Standard	v^d	13.253	10.618	13.258	13.122	13.169	10.480
Dirichlet	(se^d)	(0.012)	(0.013)	(0.012)	(0.012)	(0.012)	(0.013)
	$[t^d]$	[90]	[151]	[95]	[67]	[109]	[170]
	$\{\mathrm{b}^d\}$	{0.41}	{0.75}	{-0.88}	{-1.7}	{0.57}	{-1.0}
Contour	g	0.1304	0.0741	0.4055	0.3930	1.1249	1.1824
method	v^s	13.253	10.597	13.279	13.140	13.155	10.503
(single-hit, τ)	(se^s)	(0.012)	(0.013)	(0.012)	(0.012)	(0.012)	(0.013)
	$[t^s]$	[16]	[19]	[39]	[23]	[56]	[57]
	$\{\mathrm{b}^s\}$	{0.44}	{-0.87}	{0.92}	{-0.16}	{-0.59}	{0.69}
	$\mathbf{e}^s$	**5.6**	**7.9**	**2.4**	**2.9**	**1.9**	**3.0**
Contour	v^b	13.260	10.613	13.272	13.153	13.171	10.509
method	(se^b)	(0.012)	(0.013)	(0.012)	(0.012)	(0.012)	(0.013)
(biggest-bite)	$[t^b]$	[4.1]	[7.5]	[20]	[9.1]	[22]	[19]
	$\{\mathrm{b}^b\}$	{1.0}	{0.35}	{0.32}	{0.89}	{0.74}	{1.2}
	$\mathbf{e}^b$	**22**	**20**	**4.8**	**7.4**	**5.0**	**8.9**

Table 3.8. Knock-in put option $\mathcal{O}^{IP}$, $\sigma = 0.2$

$S_0 = 85$, $\sigma = 0.2$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	11.07610	7.03916	11.18860	11.56830	10.82990	6.59548
	(se)	(0.00046)	(0.00062)	(0.00044)	(0.00042)	(0.00048)	(0.00062)
Standard	v^d	11.0722	7.0407	11.1862	11.5709	10.8334	6.5929
Dirichlet	(se^d)	(0.0046)	(0.0062)	(0.0045)	(0.0042)	(0.0048)	(0.0062)
	$[t^d]$	[73]	[119]	[75]	[53]	[86]	[135]
	$\{\mathrm{b}^d\}$	{-0.85}	{0.25}	{-0.54}	{0.62}	{0.74}	{-0.41}
	g	0.1304	0.0741	0.4055	0.3930	1.1249	1.1824
Contour	v^s	11.0800	7.0351	11.1894	11.5678	10.8323	6.6003
method	(se^s)	(0.0046)	(0.0062)	(0.0045)	(0.0042)	(0.0047)	(0.0062)
(single-hit)	$[t^s]$	[14]	[16]	[34]	[20]	[49]	[49]
	$\{\mathrm{b}^s\}$	{0.86}	{-0.66}	{0.18}	{-0.12}	{0.51}	{0.77}
	$\mathbf{e}^s$	**5.3**	**7.5**	**2.2**	**2.7**	**1.8**	**2.7**
Contour	v^b	11.0758	7.0356	11.1922	11.5645	10.8274	6.5868
method	(se^b)	(0.0046)	(0.0062)	(0.0045)	(0.0042)	(0.0047)	(0.006)2
(biggest-bite)	$[t^b]$	[3.2]	[5.8]	[17]	[7]	[19]	[20]
	$\{\mathrm{b}^b\}$	{-0.07}	{-0.57}	{0.81}	{-0.90}	{-0.53}	{-1.4}
	$\mathbf{e}^b$	**23**	**21**	**4.5**	**7.5**	**4.7**	**6.9**

Table 3.9. Recovery option $\mathcal{O}^{rec}$, $\sigma = 0.2$

barrier is valued, gains from the single-hit method is reduced to ~2-3. However, they are still worthwhile. The recovery option benefits from using the single-hit method to roughly the same extent as the other options. The highest gain when valuing the recovery option comes from the biggest-bite variant with a positive slope linear barrier. A gain is 23.

As expected, the biggest-bite variant out-performs the single-hit method in all cases. The largest gains are achieved when valuing linear barriers, both increasing and decreasing. In these cases, the biggest-bite variant produces gains of ~20. This level of gains is significant. Similar to the single-hit method, gains are decreasing when valuing options with more difficult barriers, namely those with sine function. Nevertheless, even with these barriers, the minimum gain is around ~5 which is at least twice as great as the single-hit method. Notice that gains for the contour bridge method, both the single-hit and the biggest-bite variants, are broadly comparable across different types of option. This is because the main computational burden is to compute a hitting time τ, which will be similar for the same barrier regardless of the option type.

3.3.3.2 Bias in the computation of S_T

The knock-in option values in tables 3.7 and 3.8 were produced by simulating directly values of S_T from those of S_τ, when one establishes that $\tau < T$, as described in section 3.2.5. As pointed out, when using the iterative method, there may be biases in option values because of the bimodal distribution. This is illustrated in table 3.10. Table 3.10 shows the extent of the bias seen in S_T when using the iterative method.

Results in table 3.10 suggest that biases are very large. They are many standard errors away from the range ± 2. Even though biases seem to decrease as contours become steeper, they are still ~5 standard errors away from the benchmark values. The consequences when increasing the growth rate g are that the method will become more computationally expensive and gains will be decreased. In this case, efficiency gains are less than 1. Therefore, the iterative method of computing S_T should not be used.

3.3.3.3 Efficiency gains close to and far from the barrier

In this section, efficiency gains when S_0 is either close to or far from the barrier's initial value $b(0)$ are investigated. First, the case where S_0 is close to $b(0)$ is discussed.

$S_0 = 85$, $\sigma = 0.2$		Benchmark	Standard Dirichlet	Contour bridge method (single-hit, ι)			
				g			
				2.1972	6.8024	9.1050	11.4076
Knock-out put	c	0.09029	0.09016	0.10955	0.09687	0.09532	0.09415
	(se)	(0.00007)	(0.00067)	(0.00068)	(0.00068)	(0.00068)	(0.00068)
	$[t]$		[89]	[79]	[195]	[249]	[305]
	{b}		{-0.20}	{28}	{10}	{7.4}	{5.7}
	e		**1**	**1.1**	**0.46**	**0.70**	**0.3**
Knock-in call	c	1.33384	1.3312	1.1912	1.2898	1.3098	1.3041
	(se)	(0.00048)	(0.0047)	(0.0045)	(0.0047)	(0.0047)	(0.0047)
	$[t]$		[89]	[42]	[101]	[128]	[157]
	{b}		{-0.56}	{-32}	{-9}	{-5.1}	{-6.3}
	e		**1**	**2.1**	**0.88**	**0.70**	**0.57**

Table 3.10. Knock-in call option $\mathcal{O}^{IC}$ and knock-out put option $\mathcal{O}^{OP}$, single-hit ι, $\sigma = 0.2$

Near to the barrier initial values

Tables 3.11 and 3.12 report results for the rebate, $\mathcal{O}^{R}$, and the knock-in call, $\mathcal{O}^{IC}$, when $S_0 = 80.5$.

Numerical results suggest that, when S_0 is close to $b(0)$, efficiency gains from the single-hit method are often less than one. The biggest-bite variant has gains of around 1 or slightly greater (except for the bull spread barrier), but they are not very substantial. The vertical contour variant performs very poorly, taking 10 to 20 times longer that the Dirichlet method to achieve the same standard error.

It is clear that no method performs well when initial asset values are close to the barrier's initial values. Although the biggest-bite method marginally out-performs the Dirichlet method, it cannot be argued that it is exceptionally superior.

This is due to the advantage of the Dirichlet method when the initial asset value is close to the barrier or the barrier rises sharply during the life of the option. In these cases, there is a high probability that the barrier will be hit. Since the Dirichlet method exits early when the barrier is hit, its computational time decreases significantly in this case.

Far from the barrier initial values

Tables 3.13 and 3.14 give results when $S_0 = 100$.

When S_0 moves further away from $b(0)$, gains from the contour bridge method, both the single-hit variant and the biggest-bite variant, increase significantly.

Even though the biggest-bite variant out-performs the single-hit method in

$S_0 = 80.5$, $\sigma = 0.2$		Barrier type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	4.92702	4.71688	4.92327	4.95458	4.77969	4.20378
	(se)	(0.00006)	(0.00011)	(0.00006)	(0.00005)	(0.00010)	(0.00018)
Standard Dirichlet	v^d	4.92664	4.7170	4.92430	4.95470	4.7794	4.2015
	(se^d)	(0.00058)	(0.0011)	(0.0006)	(0.00045)	(0.0010)	(0.0018)
	$[t^d]$	[11]	[21]	[12]	[7.2]	[35]	[64]
	$\{\mathrm{b}^d\}$	{0.67}	{0.14}	{1.72}	{0.27}	{-0.29}	{-1.3}
Vertical contour variant	v^B	4.92717	4.7152	4.92330	4.9541	4.7790	4.2011
	(se^B)	(0.00058)	(0.0012)	(0.00059)	(0.0045)	(0.0010)	(0.0018)
	$[t^B]$	[171]	[197]	[234]	[302]	[445]	[459]
	$\{\mathrm{b}^B\}$	{0.26}	{-1.5}	{0.05}	{0.27}	{-0.72}	{-1.5}
	$\mathbf{e}^B$	**0.06**	**0.11**	**0.05**	**0.36**	**0.08**	**0.14**
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.3930	1.2149	1.1824
	v^s	4.92720	4.7172	4.92324	4.9548	4.7794	4.2047
	(se^s)	(0.00058)	(0.0011)	(0.00058)	(0.00045)	(0.0010)	(0.0018)
	$[t^s]$	[15]	[20]	[31]	[20]	[44]	[50]
	$\{\mathrm{b}^s\}$	{0.31}	{0.28}	{-0.05}	{0.49}	{-0.29}	{0.51}
	$\mathbf{e}^s$	**0.73**	**1.1**	**0.40**	**0.36**	**0.8**	**1.3**
Contour method (biggest-bite)	v^b	4.92646	4.7176	4.92414	4.95397	4.7810	4.2040
	(se^b)	(0.00058)	(0.0011)	(0.00058)	(0.00046)	(0.0010)	(0.0018)
	$[t^b]$	[4]	[10]	[23]	[7.4]	[29]	[25]
	$\{\mathrm{b}^b\}$	{-0.97}	{0.64}	{1.5}	{-1.3}	{1.3}	{0.12}
	$\mathbf{e}^b$	**2.7**	**2.1**	**0.54**	**0.97**	**1.2**	**2.6**

Table 3.11. Rebate option $\mathcal{O}^R$, close to the barrier, $\sigma = 0.2$

$S_0 = 80.5$, $\sigma = 0.2$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	1.80283	1.66813	1.78235	1.87312	1.48693	1.13244
	(se)	(0.00058)	(0.00056)	(0.00057)	(0.00059)	(0.00051)	(0.00045)
Standard Dirichlet	v^d	1.8059	1.6667	1.7829	1.8627	1.4913	1.1284
	(se^d)	(0.0058)	(0.0056)	(0.0057)	(0.0059)	(0.0051)	(0.0045)
	$[t^d]$	[11]	[18]	[10]	[6]	[29]	[52]
	$\{\mathbf{b}^d\}$	{0.53}	{-0.25}	{0.09}	{-1.8}	{0.85}	{-0.89}
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.393	1.1249	1.1824
	v^s	1.7980	1.6632	1.7809	1.8684	1.4924	1.1294
	(se^s)	(0.0057)	(0.0055)	(0.0057)	(0.0059)	(0.0051)	(0.0045)
	$[t^s]$	[16]	[17]	[26]	[16]	[38]	[41]
	$\{\mathbf{b}^s\}$	{-0.85}	{-0.90}	{-0.26}	{-0.80}	{1.1}	{-0.7}
	$\mathbf{e}^s$	**0.69**	**1.1**	**0.38**	**0.38**	**0.76**	**1.3**
Contour method (biggest-bite)	v^b	1.8073	1.6610	1.7755	1.8641	1.4827	1.1343
	(se^b)	(0.0057)	(0.0055)	(0.0057)	(0.0059)	(0.0051)	(0.0045)
	$[t^b]$	[4.1]	[9.7]	[23]	[7.6]	[28]	[25]
	$\{\mathbf{b}^b\}$	{0.78}	{-1.3}	{-1.2}	{-1.5}	{-0.83}	{0.41}
	$\mathbf{e}^b$	**2.7**	**1.9**	**0.43**	**0.79**	**1.04**	**2.1**

Table 3.12. Knock-in call option $\mathcal{O}^{IC}$, close to the barrier, $\sigma = 0.2$

$S_0 = 100$, $\sigma = 0.2$		Barrier type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	2.00141	0.43819	2.21583	1.86621	2.01812	0.52961
	(se)	(0.00024)	(0.00014)	(0.00024)	(0.00024)	(0.00024)	(0.00015)
Standard Dirichlet	v^d	1.9969	0.4369	2.212	1.86239	2.0182	0.5288
	(se^d)	(0.0024)	(0.0014)	(0.0024)	(0.0024)	(0.0024)	(0.0015)
	$[t^d]$	[220]	[257]	[212]	[210]	[214]	[253]
	$\{\mathrm{b}^d\}$	{-1.9}	{-0.90}	{-1.6}	{-1.6}	{0.02}	{-0.54}
Vertical contour variant	v^B	1.9991	0.4417	2.2163	1.8647	2.0173	0.5277
	(se^B)	(0.0024)	(0.0014)	(0.0024)	(0.0024)	(0.0024)	(0.0015)
	$[t^B]$	[713]	[773]	[924]	[634]	[747]	[840]
	$\{\mathrm{b}^B\}$	{-0.96}	{2.5}	{0.20}	{-0.63}	{-0.34}	{-1.3}
	e^B	**0.31**	**0.33**	**0.23**	**0.33**	**0.29**	**0.34**
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.3930	1.2149	1.1824
	v^s	2.0031	0.4396	2.2172	1.864	2.0172	0.5274
	(se^s)	(0.0024)	(0.0014)	(0.0024)	(0.0024)	(0.0024)	(0.0015)
	$[t^s]$	[12]	[11]	[35]	[24]	[49]	[40]
	$\{\mathrm{b}^s\}$	{0.7}	{1.01}	{0.57}	{-0.92}	{-0.38}	{-1.5}
	e^s	**18**	**23**	**6.1**	**8.8**	**4.4**	**6.3**
Contour method (biggest-bite)	v^b	2.0034	0.4369	2.2157	1.8672	2.0159	0.5278
	(se^b)	(0.0024)	(0.0014)	(0.0024)	(0.0024)	(0.0024)	(0.0015)
	$[t^b]$	[2.6]	[2.0]	[12]	[6.7]	[15]	[10]
	$\{\mathrm{b}^b\}$	{0.80}	{-0.92}	{-0.05}	{0.41}	{-0.93}	{-1.2}
	e^b	**85**	**128**	**18**	**31**	**14**	**25**

Table 3.13. Rebate option $\mathcal{O}^R$, far from the barrier, $\sigma = 0.2$

$S_0 = 100$, $\sigma = 0.2$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	0.38195	0.02376	0.58910	0.67193	0.57186	0.05313
	(se)	(0.00023)	(0.00006)	(0.00028)	(0.00033)	(0.00029)	(0.00009)
Standard Dirichlet	v^d	0.3794	0.02418	0.5927	0.6716	0.5752	0.05257
	(se^d)	(0.0023)	(0.00058)	(0.0028)	(0.0033)	(0.0029)	(0.00088)
	$[t^d]$	[217]	[257]	[208]	[207]	[210]	[248]
	$\{\mathrm{b}^d\}$	{-1.1}	{0.73}	{1.3}	{-0.11}	{1.1}	{-0.64}
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.393	1.1249	1.1824
	v^s	0.3819	0.02385	0.5928	0.6776	0.5721	0.05271
	(se^s)	(0.0023)	(0.00058)	(0.0028)	(0.0033)	(0.0029)	(0.00088)
	$[t^s]$	[14]	[11]	[36]	[25]	[50]	[41]
	$\{\mathrm{b}^s\}$	{-0.01}	{0.16}	{1.3}	{1.7}	{0.09}	{-0.48}
	e^s	**16**	**23**	**5.8**	**8.3**	**4.2**	**6.0**
Contour method (biggest-bite)	v^b	0.3791	0.02293	0.5919	0.6669	0.5685	0.05327
	(se^b)	(0.0023)	(0.00058)	(0.0028)	(0.0033)	(0.0029)	(0.00088)
	$[t^b]$	[2.8]	[2.4]	[12]	[6.8]	[15]	[10]
	$\{\mathrm{b}^b\}$	{-1.2}	{-1.4}	{1.0}	{-1.5}	{-1.2}	{0.16}
	e^b	**78**	**107**	**17**	**30**	**14**	**25**

Table 3.14. Knock-in call option $\mathcal{O}^{IC}$, far from the barrier, $\sigma = 0.2$

all cases, gains from the single-hit method itself are respectable. The minimum gain from the single-hit method is 4.2 for the increasing sine barrier and the maximum gain is 23 for the decreasing linear barrier. These are substantial increases from the case where S_0 is close to $b(0)$.

Gains from the biggest-bite variant are much larger. Even with the worse cases, the increasing sine barrier, the gains of 14 are obtained. When S_0 is further away from $b(0)$, the biggest-bite achieves the highest gain of 128 when valuing rebate options with a decreasing barrier.

This major improvement in the contour bridge's performance compared to that of the Dirichlet method is clear and expected. When S_0 is further away from $b(0)$, and the barrier does not rise sharply during the life of the option, the hitting probability is low. Therefore, the Dirichlet method is often forced to compute an entire sample path to establish whether the barrier is hit. In contrast, the number of steps taken by the contour bridge method is capped at T/ε_τ^2. This is potentially a considerable reduction in computational effort.

3.3.3.4 Efficiency gains with high and low volatility

Now the effects of high and low volatility on efficiency gains are discussed. Tables 3.15 and 3.16 display results when the volatility is low, $\sigma = 0.1$, and tables 3.17 and 3.18 when the volatility is relatively high, $\sigma = 0.3$.

By comparing results from table 3.15 with table 3.6, one can see that gains, both from the single-hit variant and the biggest-bite variant, are approximately the same for options with non-linear barriers. However, gains are much larger when valuing barrier options with linear barriers, both positive and negative slope. The maximum gains for the single-hit variate and the biggest-bite variant, with a decreasing linear barrier, are 18 and 55. These gains are significant.

With linear barriers, gains are also increased for knock-in calls when volatility decreases. This is illustrated by comparing the results in tables 3.16 and 3.7. When $\sigma = 0.1$, the single-hit method and the biggest-bite method achieve maximum gains of 17 and 52 respectively. When volatility increases, efficiency gains decrease slightly. One can see that gains in table 3.17 are slightly less than those in table 3.6, and gains in table 3.18 are slightly less than those in table 3.7. Nonetheless, even though gains are decreased, they are still respectable. This is shown by a maximum gain of 18 for the biggest-bite method and of 4.3 for the single-hit method.

It seems that gains to the contour bridge method tend to decline as volatility increases. This may be because, as discussed, an asset with higher volatility is more likely to hit the barrier, favouring the Dirichlet method over the contour bridge

$S_0 = 85$, $\sigma = 0.1$		Barrier type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	3.79958	0.56852	4.08922	4.13750	4.14803	1.12493
	(se)	(0.00021)	(0.00016)	(0.00018)	(0.00018)	(0.00018)	(0.00021)
Standard Dirichlet	v^d	3.7990	0.5671	4.0876	4.1371	4.1462	1.1215
	(se^d)	(0.0021)	(0.0016)	(0.0018)	(0.0018)	(0.0018)	(0.0021)
	$[t^d]$	[139]	[249]	[134]	[88]	[115]	[222]
	$\{\mathrm{b}^d\}$	{-0.28}	{-0.89}	{-0.90}	{-0.22}	{-1.0}	{-1.6}
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.3930	1.2149	1.1824
	c^c	3.7988	0.5701	4.0876	4.1368	4.1467	1.1234
	(se^c)	(0.0021)	(0.0016)	(0.0018)	(0.0018)	(0.0018)	(0.0021)
	$[t^c]$	[16]	[14]	[50]	[28]	[68]	[69]
	$\{\mathrm{b}^c\}$	{-0.37}	{0.99}	{-0.90}	{-0.39}	{-0.74}	{-0.71}
	e	**8.7**	**18**	**2.7**	**3.1**	**1.7**	**3.2**
Contour method (biggest-bite)	v^c	3.8008	0.5694	4.0900	4.1384	4.1473	1.1261
	(se^c)	(0.0021)	(0.0016)	(0.0018)	(0.0018)	(0.0018)	(0.0021)
	$[t^c]$	[4.2]	[4.5]	[21]	[12]	[26]	[18]
	$\{\mathrm{b}^c\}$	{0.58}	{0.55}	{0.43}	{0.50}	{-0.41}	{0.56}
	e	**33**	**55**	**6.4**	**7.3**	**4.4**	**12**

Table 3.15. Rebate option $\mathcal{O}^R$, $\sigma = 0.1$

$S_0 = 85$, $\sigma = 0.1$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	0.07060	0.00348	0.09559	0.20837	0.14744	0.00631
	(se)	(0.00007)	(0.00001)	(0.00007)	(0.00012)	(0.00010)	(0.00002)
Standard Dirichlet	v^d	0.07009	0.00345	0.09441	0.2062	0.1477	0.00612
	(se^d)	(0.00066)	(0.00015)	(0.00073)	(0.0012)	(0.0010)	(0.00018)
	$[t^d]$	[137]	[250]	[131]	[89]	[117]	[224]
	$\{\mathrm{b}^d\}$	{-0.78}	{-0.17}	{-1.6}	{-1.8}	{0.31}	{-1.1}
Contour method (single-hit, τ)	g	0.1304	0.0741	0.4055	0.393	1.1249	1.1824
	v^s	0.07118	0.00347	0.09583	0.2082	0.1470	0.00642
	(se^s)	(0.00066)	(0.00015)	(0.00073)	(0.0012)	(0.0010)	(0.00019)
	$[t^s]$	[17]	[15]	[51]	[29]	[69]	[70]
	$\{\mathrm{b}^s\}$	{0.88}	{-0.03}	{0.33}	{-0.19}	{-0.47}	{0.59}
	e^s	**8.1**	**17**	**2.6**	**3.1**	**1.7**	**3.2**
Contour method (biggest-bite)	v^b	0.07079	0.00345	0.09529	0.2089	0.1484	0.00653
	(se^b)	(0.00066)	(0.00015)	(0.00073)	(0.0012)	(0.0010)	(0.00018)
	$[t^b]$	[4.3]	[4.5]	[22]	[12]	[26]	[19]
	$\{\mathrm{b}^b\}$	{0.29}	{-0.17}	{-0.41}	{0.44}	{1.0}	{1.2}
	e^b	**36**	**52**	**6.0**	**7.4**	**4.5**	**12**

Table 3.16. Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.1$

$S_0 = 85$, $\sigma = 0.3$		Barrier type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	4.45818	3.75405	4.45797	4.54520	4.27622	3.40713
	(se)	(0.00015)	(0.00021)	(0.00015)	(0.00014)	(0.00017)	(0.00023)
Standard	v^d	4.4589	3.7543	4.4561	4.5447	4.2752	3.4092
Dirichlet	(se^d)	(0.0015)	(0.0021)	(0.0015)	(0.0014)	(0.0017)	(0.0023)
	$[t^d]$	[65]	[98]	[70]	[51]	[87]	[124]
	$\{\mathrm{b}^d\}$	{0.51}	{0.10}	{-1.2}	{-0.35}	{-0.62}	{0.91}
Contour	g	0.1304	0.0741	0.4055	0.3930	1.2149	1.1824
method	c^c	4.4570	3.7521	4.4588	4.5442	4.2782	3.4080
(single-hit, τ)	(se^c)	(0.0014)	(0.0021)	(0.0015)	(0.0014)	(0.0017)	(0.0023)
	$[t^c]$	[15]	[18]	[33]	[20]	[45]	[46]
	$\{\mathrm{b}^c\}$	{-0.84}	{-0.93}	{0.55}	{0.0}	{1.2}	{0.4}
	$\mathbf{e}$	**4.3**	**5.4**	**2.1**	**2.6**	**1.9**	**2.7**
Contour	v^c	4.4554	3.7563	4.4576	4.5447	4.2776	3.4054
method	(se^c)	(0.0015)	(0.0021)	(0.0015)	(0.0014)	(0.0017)	(0.0023)
(biggest-bite)	$[t^c]$	[3.6]	[7.6]	[17]	[7.7]	[21]	[19]
	$\{\mathrm{b}^c\}$	{-1.9}	{1.1}	{-0.26}	{-0.36}	{0.81}	{-0.76}
	$\mathbf{e}$	**18**	**13**	**4.1**	**6.6**	**4.1**	**6.5**

Table 3.17. Rebate option $\mathcal{O}^R$, $\sigma = 0.3$

$S_0 = 85$, $\sigma = 0.3$		Barrier Type					
Method		Linear ↑	Linear ↓	Bull.	Quad.	Sine ↑	Sine ↓
Benchmark	$\widehat{v}$	3.6822	2.63255	3.5802	4.2511	2.95794	1.71070
	(se)	(0.0011)	(0.00091)	(0.0010)	(0.0011)	(0.00091)	(0.00070)
Standard	v^d	3.679	2.6347	3.589	4.266	2.9747	1.7094
Dirichlet	(se^d)	(0.011)	(0.0091)	(0.010)	(0.011)	(0.0091)	(0.0070)
	$[t^d]$	[65]	[96]	[68]	[51]	[88]	[126]
	$\{\mathrm{b}^d\}$	{-0.29}	{0.24}	{0.87}	{1.4}	{1.8}	{-0.18}
Contour	g	0.1304	0.0741	0.4055	0.393	1.1249	1.1824
method	v^s	3.673	2.6295	3.577	4.247	2.9612	1.7114
(single-hit, τ)	(se^s)	(0.011)	(0.0091)	(0.010)	(0.011)	(0.0091)	(0.0070)
	$[t^s]$	[16]	[19]	[34]	[21]	[46]	[48]
	$\{\mathrm{b}^s\}$	-0.84	-0.34	-0.31	-0.40	0.36	0.10
	$\mathbf{e}^s$	**4.1**	**5.1**	**2.0**	**2.4**	**1.9**	**2.6**
Contour	v^b	3.672	2.6436	3.581	4.249	2.9492	1.7072
method	(se^b)	(0.011)	(0.0091)	(0.010)	(0.011)	(0.0091)	(0.0070)
(biggest-bite)	$[t^b]$	[3.9]	[7.7]	[18]	[2.8]	[21]	[20]
	$\{\mathrm{b}^b\}$	{-0.93}	{1.2}	{0.08}	{-0.19}	{-0.96}	{-0.50}
	$\mathbf{e}^b$	**17**	**12**	**3.8**	**6.5**	**4.2**	**6.3**

Table 3.18. Knock-in call option $\mathcal{O}^{IC}$, $\sigma = 0.3$

method in the single option versions implemented and compared here.

3.4 Conclusion

This chapter described a novel simulation method, the contour bridge method, that has been applied to the valuation of barrier options with exotic barriers. The results in section 3.3 illustrate that the contour bridge method generates efficiency gains over the Dirichlet method. These gains are sometimes very substantial. Although not demonstrated in this chapter it can be argued that the contour bridge method would yield significantly greater efficiency gains when applied to a book of options. The method can also be extended to apply other types of barrier option such a partial barrier option.

The method shows no evidence of bias when applied to rebate and knock-in style options. However, it may be difficult to value knock-out options since S_T cannot be simulated directly. Even though one cannot value a knock-out directly from the contour bridge method, one can recover a knock-out value from a knock-in value computed from the method by using the in-out parity. This is possible since the results suggest that the method can value a knock-in option accurately.

The numerical results suggest that the biggest-bite version of the method can produce very substantial efficiency gains. The method achieves the greatest efficiency gains when volatility is low and an initial asset value is further away from the initial barrier value. Even though gains decrease when volatility is high, they are still good. The shape of the option barrier has a clear effect on the method's performance. The highest gains are achieved when the method is applied to value options with linear barriers. However, reasonable gains are also made with the more difficult linear-sine barriers.

Even though the single-hit version of the method does not produce efficiency gains as large as the biggest-bite version does, those gains are still respectable in a number of cases.

The method relies upon the existence of contours with certain mathematical properties. This chapter has investigated only single barrier options. However, if there are contours appropriate for double barrier options, the method would extend to these options too.

In conclusion, the contour bridge method is a valuable addition to the set of simulation methods that can be applied to value barrier options.

Part II

American Put Options and Control Variates

Chapter 4

Literature Review on American Option Valuation

American options and American style derivatives have become one of the commonest financial instruments that are traded both in exchanges and in the over-the-counter market. They are popular among financial institutions such as investment banks, commercial banks, and mutual funds. However, these instruments are generally hard to price because of their early exercise features. The pricing problem of American options and American-style derivatives is important and challenging. This chapter will focus on the valuation problem and methods of American put options in both geometric Brownian motion (GBM) and non-GBM models. The chapter begins by describing American put option valuation problems in section 4.1. Section 4.2 describes approximation and general numerical methods. Simulation methods are reviewed in section 4.3. Section 4.4 provides a review of literature of American option valuation methods in non-GBM models. American barrier and power options are described in section 4.5. Section 4.6 concludes.

4.1 American put option valuation problems

Consider the standard model of Black and Scholes (1973) [38] in which the price process S_t follows the geometric Brownian motion

$$\mathrm{d}S_t = rS_t + \sigma S_t \mathrm{d}z_t, \tag{4.1.1}$$

where the risk free rate r and volatility σ are constant and positive, and z_t is a one-dimensional Brownian motion. The value of an American put option is given in Definition 3.

Definition 3. *An American put option gives a holder the right to sell an underlying asset at any time up to and including the expiration date. The value of an American put option in the Black-Scholes model at time $t \in [0,T]$ with an underlying S_t, a strike price X, a risk free rate r and volatility σ, is a value process $v_t(S)$ defined by:*

$$v_t(S) = \max_{\tau^* \in [t,T]} \mathbb{E}^{\mathbb{Q}} \left[e^{-r\tau} (X - S_{\tau^*})^+ \right], \tag{4.1.2}$$

where $\mathbb{Q}$ is the risk neutral measure and $\tau^* \in [t,T]$ is an optimal stopping time.

The distinct feature of American options, which make them different from European options, is the early exercise feature. The holder of American options can exercise anytime before an expiry date. The values of the underlying asset, that make it optimal for an option holder to exercise, form an early exercise boundary S^E (EEB thereafter). The optimal stopping time τ^* in (4.1.2) is the first time that an asset value falls below S^E. That is,

$$\tau^* = \inf\{t \geq 0 : S_t \leq S_t^E\}. \tag{4.1.3}$$

The EEB determines exercise strategies for a holder of an American put option. Given a value S_t, a holder can choose whether to exercise an American put option. A set of (S_t, t) in which it is optimal to exercise is the immediate exercise region $\mathcal{I}$:

$$\mathcal{I} = \left\{ S_t \in \mathbb{R}^+ \times [0,T] : v(S_t) \leq S_t^E \right\}. \tag{4.1.4}$$

A complement of $\mathcal{I}$ is a set of (S_t, t) in which it is optimal to continue holding an American put. It is the continuation region $\mathcal{C}$:

$$\mathcal{C} = \left\{ S_t \in \mathbb{R}^+ \times [0,T] : v(S_t) > S_t^E \right\}. \tag{4.1.5}$$

The two regions $\mathcal{I}$ and $\mathcal{C}$ are separated by the EEB, S_t^E. This is illustrated in figure 4.1.

Since the option's EEB is not known in advance, an American option pricing problem can be regarded as the free boundary value problem in which one has to compute both $v_t(S)$ and S_t^E simultaneously. This type of problem was studied by McKean (1965) [181]. There exists no closed-form solution to the American pricing problem, and the option value needs to be computed by approximation and numerical techniques.

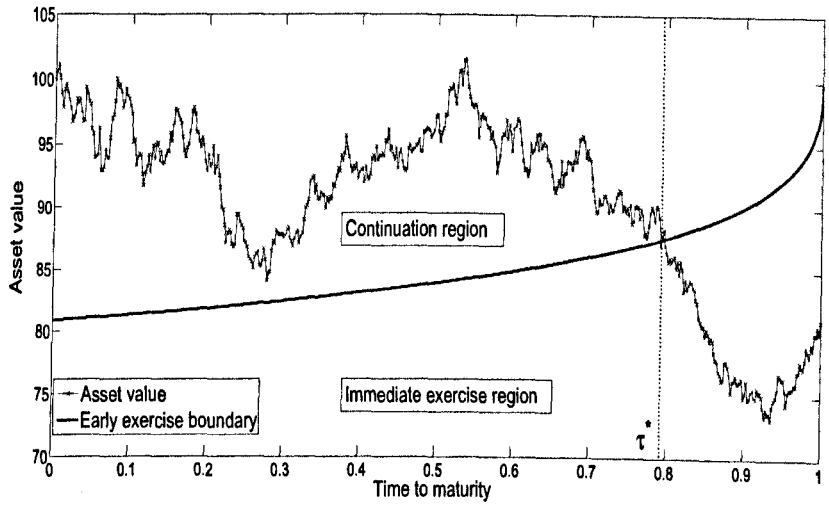


Figure 4.1. American put option's early exercise boundary (EEB)

4.2 Approximation and numerical methods

There are two main stains in the literature on the valuation methods of the American options: approximations and numerical methods. The methods discussed in sections 4.2.1 assume that the underlying asset follows a GBM process in (4.1.1).

4.2.1 Analytical approximation methods

Barone-Adesi and Whaley (1987) [27] (BAW method) proposed a quasi-analytical approximation method. The method was developed from the quadratic method of MacMillan (1986) [178]. The BAW method separates the American put option into two parts, which are the European component and the early exercise component. Barone-Adesi and Whaley (1987) [27] derived the option's EEB, and so found an approximate value of an American put option. However, despite the fast computational speed, the estimated EEB fails to completely capture time-varying shape of the exercise boundary (Ait-Sahlia and Lai (2001) [3]). Bjerksund and Stensland (1993) [37] proposed an approximation method in which the EEB is assumed to be flat. The method decomposes an American call option into a knock-out option with a rebate paid at the knock-out time. However, since the EEB is assumed to be flat, the exercise strategy is not optimal and then the technique provides only a lower bound of the option value. Bjerksund and Stensland [36] extended the method of Bjerksund and Stensland [37] by separating the option's time maturity into two

parts, each of them with a flat EEB. The performance of the method depends on accuracy in approximating bivariate normal distribution.

Another approximation approach makes use of a lower and an upper bound for an American option value. This method was proposed by Johnson (1983) [143], and Broadie and Detemple (1996) [48]. Johnson (1983) [143] used regression coefficients, which are estimated from a large set of option contracts, to obtain lower and upper bounds for the American option value. Johnson (1983) [143] then approximated the functional form of the EEB and American put prices. Broadie and Detemple (1996) [48] derived lower and upper bounds of American calls by using a capped call option written on the same underlying asset.

Geske and Johnson (1984) [104] proposed a method, the GJ method, that approximated the value of American options through an infinite series of multivariate normal distribution functions, and then used Richardson extrapolation to estimate the option value. Bunch and Johnson (1992) [53] applied the GJ method with an option with only two-exercise times but these times are chosen such that the option value is maximised.

PréKopa and Szántai (2010) [200] applied the idea of Geske and Johnson (1984) [104] and proposed a recursive method to value Bermudan and American options. Chang et al. (2007) [62] modified the GJ method by applying the idea proposed by Omberg (1987) [195]. The modified method assumes that exercise opportunities are separated by time steps in a geometric series rather than in an arithmetic series and then repeated Richardson extrapolation is used.[1] Their results suggest that the modified method performs better than the original GJ method.

Carr (1998) [56] developed a semi-analytical method to price American puts. By dividing the time to expiry into sub-intervals and assuming each of them is random and exponentially distributed, the pricing problem at each time step is equivalent to that of a perpetual option, and a PDE is reduced to an ODE. This enables Carr (1998) [56] to derive an expression for American put options.

The other important analytical approximation method for American options is the Integral Representation method, which was proposed by Kim (1990) [157], Jacka (1991) [136], Carr et al. (1992) [59] and Jamshidian (1992) [138]. This method is based on an associated integral equation for the exercise boundary of the option. The efficiency of the method depends on the particular iterative method chosen to compute the integral term. Kallast and Kivinukk (2003) [150] have recently used the Newton-Raphson method to solve an integral based on the equation

[1]See Dahlquist and Björck (1974) [78] for a more detailed discussion of Richardson extrapolation technique.

of Kim (1990) [157]. Ju (1998) [147] proposed a multi-piecewise exponential function, and in combination with Richardson Extrapolation, he showed, as pointed out by Detemple (2006) [85], that a three-point extrapolation scheme performs as well as the lower-upper bound approximation method in terms of both root mean square error and computational speed.

4.2.2 PDE and lattice methods

In pricing American options there are two main categories of numerical methods. The first category is the method that solves directly the partial differential equation (PDE) governing an American option value in which both time and space are discretised (PDE method). The second category is the method that is based on risk-neutral valuation at each time step (lattice and simulation methods). This section will review PDE and lattice methods and section 4.3 will review simulation methods.

One of the first PDE methods is the finite difference method introduced by Schwartz (1977) [223] and Brennan and Schwartz (1978) [46]. Let S be the asset value that follows a geometric Brownian motion in (4.1.1) and X be the strike price. Denote a value of an American put option as v_t, the idea of this method is to solve the following free boundary value problem:

$$\frac{\partial v}{\partial t} + rS\frac{\partial v}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 v}{\partial S^2} - rv = 0, \qquad \forall S > S_t^E, t > 0 \tag{4.2.1}$$

$$v_t > (X - S)^+, \qquad \forall S > S_t^E, t > 0 \tag{4.2.2}$$

$$v_t = (X - S)^+, \qquad \forall S \leq S_t^E, t \geq 0 \tag{4.2.3}$$

$$v_T = (X - S_T)^+, \tag{4.2.4}$$

$$\left.\frac{\partial v}{\partial S}\right|_{S=S_t^E} = -1, \qquad t \geq 0 \tag{4.2.5}$$

$$\lim_{S \uparrow \infty} v_t = 0, \qquad t \geq 0, \tag{4.2.6}$$

where S_t^E is the early exercise boundary, by converting a PDE into a set of difference equations, each of which is solved iteratively. This is a free boundary problem because S_t^E is unknown and one has to solve for v and S_t^E simultaneously. Regarding some finite difference methods such as the implicit and the Crank-Nicolson method, as the stock price step size and time step size approach zero, the solution converges. In some cases, such as an explicit finite difference, the method may not converge if a certain condition is not satisfied (Tavella and Randall (2000) [234]).

When only the time derivative is discretised, the method is called the (horizontal) method of lines. In particular, the method is based on semi-discretisation where the domain (s,t) is replaced by a set of parallel lines. Each of the lines is defined by a constant value of t. Meyer and van der Hoek (1997) [185] proposed this method to value American options. The advantage of this method, as mentioned by Chiarella et al. (2006) [66], is that the option price, delta, gamma, and exercise boundary are all computed as part of a solution process.

Another way to solve American option values is to formulate the problem as a Linear Complementary problem. One of the most popular algorithms is the project successive over-relaxation (PSOR) iterative method of Cryer (1971) [77]. PSOR is an iterative method. It is a standard successive over-relaxation (SOR) method in a sense that it has a relaxation parameter that is chosen to improve the convergence of the iteration. It is modified to update only non-negative SOR solutions. However, as mentioned by Dempster et al. (1998) [83], the number of iterations required to converge may vary and sometimes can grow rapidly with the number of spatial grid points.

One of the earliest methods in the second category are lattice methods, first proposed by Cox et al. (1979) [76]. These methods divide the continuous movement of stock prices into equally spaced discrete time steps, and the value of an option is obtained by discounting back the sum of a possible option value of each future state (up or down) multiplied by a predetermined risk neutral probability. Even though this method converges as the number of time steps increases, it may be difficult to value an option with more than one or two underlying assets. Amin and Khanna (1994) [12] used a probabilistic approach to prove the convergence of a binomial tree for American options. The analysis of the convergence of a binomial tree was also investigated in Jiang and Dai (1999) [141].

Another method that is related to a binomial tree method is the Bernoulli walk method proposed by Ait-Sahlia and Lai (199) [4]. In this method, an underlying Brownian motion is approximated by a symmetric random Bernoulli walk and a backwards algorithm is applied to obtain the option value and its early exercise boundary.

4.3 Simulation methods

The remaining category of numerical methods for valuing American options is Monte Carlo simulation. Let $v_t(S)$ be the value of time t of an American option maturing at time $T > t$. The option value is required at time t_0. Divide the interval $[0,T]$

into N equal steps $0 = t_0 < t_1 < \ldots < t_N = T$ where $t_i - t_{i-1} = \Delta t$ is constant. Write $S^i = S_{t_i}$ for the value of the underling asset at time t_i, $v^i = v\left(S^i\right)$ for the value of the option at time t_i. Suppose a risk free rate r is constant. Define

$h^i \equiv h\left(S^i\right)$: The immediate exercise payoff at time t_i.
$C^i \equiv C\left(S^i\right)$: The continuation value of the American option; the option value if not exercise at time t_i but may optimally exercise at times $t_{i+1}, \ldots, t_N$.

The Monte Carlo value of an American option at time $t = 0$, $\widehat{v}_0$ is determined recursively:

$$\widehat{v}^N = h^N \tag{4.3.1}$$

$$\widehat{v}^{i-1} = \max\left(h^{i-1}, \mathbb{E}\left[\mathrm{e}^{-r\Delta t} v^i \mid S^{i-1}\right]\right), \tag{4.3.2}$$

for $i = 1, \ldots, N$. From (4.3.2), the option value at time to maturity is its payoff h^N. Equation (4.3.2) means that the value at time t_{i-1} of American option is the maximum of the immediate exercise value and the time t_{i-1} continuation value. One can write $C^i = \mathbb{E}\left[\mathrm{e}^{-r\Delta t} v^i | S^{i-1}\right]$.

Simulation techniques can be divided into four main types; which are the mesh method, the state-space partitioning method, the duality method, and the functional form method.

4.3.1 Mesh methods

The mesh methods can be sub-categorised into two types; which are the random lattice and the stochastic mesh method. These were proposed by Broadie and Glasserman (1997, 2004) [50, 49].

4.3.1.1 Random lattice method

The first method is based on randomly sampled states. It generates two option value estimators. One is biased upward and the other one is biased downward. The method is based on the simulation of a lattice of paths of asset values $S^0, S^1, \ldots, S^N$ using (3.1.3). First, a branching parameter $k \geq 2$ is chosen. Then at S^0, one simulates k independent successor asset values $S_1^1, S_2^1, \ldots, S_k^1$ all conditioning on S^0. Next, from each S_j^1, where $j = 1, \ldots, k$, simulate another k independent successors $S_{j1}^2, \ldots, S_{jk}^2$. Next, from each $S_{j_1 j_2}^2$, one generates k successors $S_{j_1 j_2 1}^3, \ldots, S_{j_1 j_2 k}^3$, *et cetera.*

$S^i_{j_1,\ldots,j_i}$ is a generic node in the tree at time step i. This means the node is reached by the j_1th branch from S^0 and the j_2th branch from the next node, and so on. It is important to note the difference between the random lattice method and a standard non-recombining lattice method. In a lattice method, successor nodes are generated by deterministic up and down probability and hence nodes at the ith step appear according to their values; that is the highest node corresponds to the greatest asset value. In the random lattice method, successor nodes are simulated (randomly sampled) and nodes at the ith step appear according to the order in which they are simulated.

Denote $\widehat{v}_{j_1,\ldots,j_i}$ as the high estimator at node $S^i_{j_1,\ldots,j_i}$. It is defined recursively as:

$$\widehat{v}^N_{j_1,\ldots,j_N} = h^N_{j_1\ldots j_N} \tag{4.3.3}$$

$$\widehat{v}^i_{j_1,\ldots,j_i} = \max\left(h^i_{j_1,\ldots,j_i}, \frac{1}{k}\sum_{j=1}^{k} \mathrm{e}^{-r\Delta t}\widehat{v}^{i+1}_{j_1,\ldots,j_i}\right). \tag{4.3.4}$$

$\frac{1}{k}\sum_{j=1}^{k}\Delta_i\widehat{v}^{i+1}_{j_1,\ldots,j_i}$ is computed simply by taking an average of discounted option values from k successors, each of them is assigned equal weight. (4.3.4) is a high bias estimator $\left(\mathbb{E}[\widehat{v}^i_{j_1,\ldots,j_i} \mid S^i_{j_1,\ldots,j_i}] > v^i_{j_1,\ldots,j_i} \mid S^i_{j_1,\ldots,j_i}\right)$ because it embodies perfect foresight.

To remove the high bias, Broadie and Glasserman (1997) [50] introduced the low estimator. The idea is to divide branches at each node into two sets, separating the exercise decision from the continuation value. At each node, the method uses the first branch to estimate the continuation value and other $k-1$ branches to approximate the exercise value. The procedure is repeated $k-1$ times by using the second branch to estimate continuation value, *et cetera*. Denote $\underline{\widehat{v}}_{j_1,\ldots,j_i}$ as the low estimator at node $S^i_{j_1,\ldots,j_i}$. It is defined recursively as:

$$\underline{\widehat{v}}^N_{j_1,\ldots,j_N} = h^N_{j_1,\ldots,j_N}. \tag{4.3.5}$$

At node $j_1,\ldots,j_i$ at time step i and for each $l = 1,\ldots,k$, set

$$\underline{\widehat{v}}^{il}_{j_1,\ldots,j_i} = \begin{cases} h^i_{j_1,\ldots,j_i} & \text{if } \frac{1}{k-1}\sum_{j=1,j\neq l}^{k}\mathrm{e}^{-r\Delta t}\underline{\widehat{v}}^{i+1}_{j_1,\ldots,j_i} \leq h^i_{j_1,\ldots,j_i} \\ \mathrm{e}^{-r\Delta t}\widehat{v}^{i+1}_{j_1,\ldots,j_il} & \text{otherwise.} \end{cases} \tag{4.3.6}$$

Then, set

$$\widehat{\underline{v}}^{i}_{j_1 \ldots j_i} = \frac{1}{k} \sum_{l=1}^{k} \widehat{\underline{v}}^{il}_{j_1 \ldots j_i}. \tag{4.3.7}$$

As k increases, both estimators are asymptotically unbiased and the method provides a confidence interval for the American option value. As suggested by Broadie and Glasserman (2004) [49], the main drawback of this method is that the computational burden increases exponentially in the number of time steps (exercise dates).

4.3.1.2 Stochastic mesh method

The idea of the stochastic mesh method is similar to the random lattice method but now the weight assigned to each node is assumed to be random. Another distinction is that when computing the option at a node at time step t_i, the mesh uses values from all nodes at time t_{i+1}, unlike the first method that uses values from successors of the current node. Since this method keeps the number of nodes at each time step fixed, the computational requirement is linear, rather than exponential as in the first method, in the number of time step. In this case, for $i = 1, ..., N$ and $j = 1, ..., k$ let S^i_j denotes the jth node at the ith exercise date. The estimated option value, $\widehat{v}^i_j$ is defined as:

$$\widehat{v}^N_j = h^N_j \tag{4.3.8}$$

$$\widehat{v}^i_j = \max\left(h^i_j, \frac{1}{k} \sum_{k=1}^{k} \mathrm{e}^{-r\Delta t} w^i_{jl} \widehat{v}^{i+1}_l\right), \tag{4.3.9}$$

where w^i_{jl} is a weight attached to the arc joining S^i_j with S^{i+1}_k. At node 0, one sets

$$\widehat{v}^0 = \frac{1}{k} \sum_{l=1}^{k} \mathrm{e}^{-r\Delta t} \widehat{v}^1_l. \tag{4.3.10}$$

Note that if one allows exercise at time t_0, set $\widehat{v}^0 = \max\left(\frac{1}{k}\sum_{l=1}^{k} \mathrm{e}^{-r\Delta t}\widehat{v}^1_l, h^0\right)$. Although this method is convergent, it relies on the explicit knowledge of the transition density of the underlying state variables, which may not be known in many cases (Broadie and Glasserman (2004) [49]).

4.3.2 State-space partitioning methods

The second method is the state-space partitioning methods proposed by by Barraquand and Martineau (1995) [29], Tilley (1993) [236] and extended by Raymar

and Zwecher (1997) [204] and Jin et al. (2007) [142]. The main idea of this method is based on the partitioning, that is defined in advance, of the state space of the underlying asset. Unlike the random tree method that uses random sample states, the states in this method are defined in advance. For the time step $t_i, i = 1, ..., N$, one has a finite partition $Q_1^i, ..., Q_{k_i}^i$ of S^i into k_i subsets. At time t_0, set $k_0 = 1$ and $Q_1^0 = S_0$. Next, define transition probabilities

$$q_{jl}^i = \mathbb{P}\left[S^{i+1} \in Q_l^{i+1} \mid S^i \in Q_j^i\right] \tag{4.3.11}$$

where $j = 1, \ldots, k_i$, $l = 1, \ldots, k+1$, and $i = 0, \ldots, N-1$. Then for each $j = 1, \ldots, k_i$ and $i = 1, \ldots, N$, define averaged payoffs

$$h_j^i = \mathbb{E}\left[h^i \mid S^i \in Q_j^i\right]. \tag{4.3.12}$$

This method uses the simulation to approximate $\widehat{q}_{jl}^i \approx q_{jl}^i$ and $\widehat{h}_j^i \approx h_j^i$. To do this, simulate a large number of paths of $S^0, S^1, \ldots, S^N$ and then set p_{jl}^i as the number of paths moving from Q_j^i to Q_j^{i+1} for all $j = 1, \ldots, k_i$, $l = 1, \ldots, k+1$, and $i = 0, \ldots, N-1$. Then one computes

$$\widehat{q}_{jl}^i = \frac{p_{jl}^i}{p_{j1}^i + \ldots + p_{jk_i}^i}, \tag{4.3.13}$$

and $\widehat{h}_j^i$ as an average of h^i for those paths where $S^i \in Q_j^i$.
The estimated option value is recursively defined as:

$$\widehat{v}_j^N = \widehat{h}_j^N \; \forall j = 1, \ldots, k_N \tag{4.3.14}$$

$$\widehat{v}_j^i = \max\left(\widehat{h}_j^i, \sum_{l=1}^{k_{i+1}} q_{jl}^i \mathrm{e}^{-r\Delta t} \widehat{v}_l^{i+1}\right), \tag{4.3.15}$$

Even though the method can be used to price multi-dimensional American options, it relies on the selection of the state-space partitioning. Also, when the dimension is large, the computational time is quadratic in a number of dimensions. Raymar and Zwecher (1997) [204] extended this method by including the second statistic, such as the second highest moment or the median of the stock price vector. They find that their method produces values that are closer to the benchmarks. This method was also investigated by Baldi and Pagés (2000, 2003) [22, 23] and Baldi et al. [24].

The method that is similar to the partitioning method is the bundling method

of Tilley (1993) [236]. The method simulates the asset value paths at once, reorders them from the lowest to the highest, and bundles them into group. These groups are used to approximate the option's continuation value. At each time, the method compares the immediate exercise value and the continuation value. The American option value is obtained by backward iteration. The bundling method was extended by Jin et al. (2007) [142] to value high-dimensional American options. They employed quasi-Monte Carlo to construct bundles. Their results show that the method can be used to value options up to 15 assets.

4.3.3 Duality methods

Unlike other simulation methods that view the American option pricing problem as a maximum over stopping times, this method is based on a minimisation. Roger (2002) [216], Haugh and Kogan (2004) [120], and Anderson and Broadie (2004) [13] proposed a method based on a dual formation. Like the mesh method, this approach can generate both lower and upper bounds of option prices. The upper bound is approximated by the dual minimising over a class of supermartingales or martingales. In this method, the primal problem is an American option pricing problem. Write $h_t = h(S_t)$. The primal problem is

$$v_0 = \max_{\tau^* \in [0,T]} \mathbb{E}\left[e^{-r\tau^*} h_{\tau^*}\right], \tag{4.3.16}$$

where $\tau^* \in [0,T]$ is a stopping time of the American option. Define a martingale $\pi = \{\pi_i, i = 0, ..., N\}$ with $\pi_0 = 0$. By the optional sampling theorem, one has the upper bound

$$\mathbb{E}\left[e^{-r\tau^*} h_{\tau^*}\right] = \mathbb{E}\left[e^{-r\tau^*} h_{\tau^*} - \pi_{\tau^*}\right] \leq \mathbb{E}\left[\max_{i=1,...,N}\left(e^{-rt_i} h^i - \pi_i\right)\right], \tag{4.3.17}$$

and hence

$$\mathbb{E}\left[e^{-r\tau^*} h_{\tau^*}\right] \leq \inf_{\pi} \mathbb{E}\left[\max_{i=1,...,N}\left(e^{-rt_i} h^i - \pi_i\right)\right]. \tag{4.3.18}$$

Since the inequality in (4.3.18) holds for every τ^*, it must also hold for the supremum over τ^*. Hence, write $U_0 = \inf_\pi \mathbb{E}\left[\max_{i=1,...,N}\left(e^{-rt_i} h^i - \pi_i\right)\right]$, one has

$$v_0(S_0) = \sup_{\tau^*} \mathbb{E}\left[e^{-r\tau^*} h_\tau\right] = U_0. \tag{4.3.19}$$

U_0 is the dual problem.

To find a martingale that gives a tight upper bound, Roger (2002) [216] used this duality method and generated a martingale process by using a separate

simulation. An extrapolation technique was also used.

The representation in (4.3.19) was also used by Haugh and Kogan (2004) [120] (they worked with π, a supermartingale) and Anderson and Broadie (2004) [13]. Haugh and Kogan (2004) [120] employ a low discrepancy sequence and neural network algorithm to estimate the continuation value of the option, and then a lower and an upper bound of the American option.

Anderson and Broadie (2004) [13] pursued a similar methodology. However, there are two distinct differences between their method and Haugh and Kogan's method. First, Anderson and Broadie (2004) [13] approximated the optimal exercise policy (rather than the initial option price) to estimate the bounds. Second, they employed only a simulation to construct an upper bound. By computing their upper bound, together with using Longstaff and Schwartz's method (which will be discussed next) to compute a lower bound, they found that their resulting confidential intervals are consistent, and some are tighter than the stochastic mesh results of Broadie and Glasserman (2004) [49].

4.3.4 Functional form methods

This class of method approximates an American option's continuation value with a functional form. It is based on the use of regression together with Monte Carlo simulation. The method suggests that the time t_i continuation value of the option, C^i can be approximated by a countable linear combination of known functions of asset values, S^i, and uses regression to estimate coefficients for this approximation. Carriere (1996) [61] suggested the use of non-parametric regression to approximate C^i.

Tsitsiklis and Roy (1999, 2000) [239, 238] and Longstaff and Schwartz (2001) [176] proposed the use of least-square regression to find a function to approximate continuation values. Tsitsiklis and Roy (1999, 2000) [239, 238] presented their work in a theoretical way while Longstaff and Schwartz (2001) [176] presented it in a practical way. The method of Longstaff and Schwartz (2001) [176] (LSLS) is described as follows.

4.3.4.1 The LSLS method

Suppose $\widehat{v}_j^i$ is the time t_i Monte Carlo value of an American option on the jth path. Let $\widehat{C}_j^i = \mathrm{e}^{-r\Delta t}\widehat{v}_j^{i+1}$ be the Monte Carlo continuation value of an American option. Let $C^i = \left\{C_j^i\right\}_{j=1,\ldots,M}$ be the true continuation value. At time t_i, the LSLS method

approximates C^i by $\widetilde{C}^i$ using a known functional form f so that

$$C^i \approx \widetilde{C}^i = f\left(S \mid a^i\right), \tag{4.3.20}$$

where a^i are parameters to be found. The parameters a^i are chosen such that a norm $G(a) = \left\|\widehat{C}^i - f\left(S \mid a^i\right)\right\|$ is minimised. Longstaff and Schwartz (2001) [176] suggested that f can be a finite linear combination of a set of basis functions,

$$f\left(S \mid a^i\right) = \sum_{k=1}^{K} a_k^i \psi_k\left(S\right), \tag{4.3.21}$$

for a set of basis functions $\{\psi_k\}_{k=1,\ldots,K}$. A set of coefficients $a^i = \left\{a_1^i, \ldots, a_K^i\right\}$ can be estimated by a least square regression at each time t_i.

For a fixed time t_i, let $\Psi = \left\{\psi_k\left(S_j^i\right)\right\}_{j=1,\ldots,M}^{k=1,\ldots,K} \in \mathbb{R}^{M\times K}$ and let $\widehat{C} = \left\{\widehat{C}_j^i\right\}_{j=1,\ldots,M} \in \mathbb{R}^M$. One solves for $\mathrm{a} = \left\{a_k^i\right\}_{k=1,\ldots,K} \in \mathbb{R}^K$. That is

$$\min \left\|\Psi \mathrm{a} - \widehat{C}\right\|. \tag{4.3.22}$$

An optimal set of parameters, a, can be found by

$$\mathrm{a} = \left(\Psi^\top \Psi\right)^{-1} \Psi^\top \widehat{C}. \tag{4.3.23}$$

Unfortunately, it is well known that, when working with more than five or six basis functions the ordinary least square (OLS) may fail. This is because the basis functions may be highly correlated with each other, and hence the method cannot distinguish values of the criteria function, $G(a)$, when different combinations of basis functions are used. The method then attempts to invert a singular matrix (Longstaff and Schwartz (2001) [176]). To cope with this problem, a singular value decomposition (SVD) technique can be used instead. This method is useful when working with matrices that are close to singular (Press et al. (2007) [201]). The use of the SVD and other techniques in the LSLS method was investigated by Areal et al. (2008) [14].
Write $h_j^i = h\left(S_j^i\right)$ for the option's payoff. The LSLS algorithm to value American option is given as follows.

The LSLS algorithm.

1. Simulate a set of M independent paths $S_j = \left\{S_j^0, \ldots, S_j^N\right\}$. Then compute

an option value for time $t = t_N$, $\widehat{v}_j^N = h_j^N$ where, for a put option, $h_j^N = \left(X - S_j^N\right)^+$.

2. At each time t_i, where $1 \leq i \leq N-1$, write $\widehat{C}_j^i$ for the continuation value on the jth sample path. Set $\widehat{C}_j^i = \mathrm{e}^{-r\Delta t}\widehat{v}_j^{i+1}$.

3. Write $\widetilde{C}_j^i$ for an approximated continuation value. Find parameters $a^i = \{a_k^i\}_{k=1,\ldots,K}$ so that $\widehat{C}_j^i \sim \widetilde{C}_j^i = f\left(S_j^i \mid a^i\right)$, where

$$f\left(S_j^i \mid a^i\right) = \sum_{k=1}^{K} a_k^i \psi_k\left(S_j^i\right), \tag{4.3.24}$$

for the chosen set of basis functions $\{\psi_k\}_{k=1,\ldots,K}$. Parameters a^i can be found by

$$a^i = \arg\min_a \left\| \widehat{C}_j^i - f\left(S_j^i \mid a\right) \right\|. \tag{4.3.25}$$

4. Exercise is made if $h_j^i > \widetilde{C}_j^i$. To obtain the EEB, one sets $\mathbb{S}_i^{\mathrm{E}} = \left\{S_j^i \mid h_j^i > \widetilde{C}_j^i\right\}$, which is the set of S^i at which a holder exercises the option. Then one sets

$$S_i^{\mathrm{E}} = \max_S \left\{S \in \mathbb{S}^{\mathrm{E}}\right\}. \tag{4.3.26}$$

S_i^{E} is the greatest asset value at time t_i that the option is exercised. The set $S^E = \left\{S_i^{\mathrm{E}}\right\}$, $i = 1, \ldots, N$ represents the option's EEB.

5. The option value at time t_i, $\widehat{v}_j^i$, is set to be

$$\widehat{v}_j^i = \begin{cases} h_j^i, & \text{if } S_j^i \leq S_i^{\mathrm{E}}, \\ \mathrm{e}^{-r\Delta t}\widehat{v}_j^{i+1}, & \text{otherwise.} \end{cases} \tag{4.3.27}$$

6. At time t^0, the American option value is

$$\widehat{v}^0 = \max\left(\frac{1}{M}\sum_{j=1}^{M} \mathrm{e}^{-r\Delta t}\widehat{v}_j^1, h_j^0\right). \tag{4.3.28}$$

The properties of the LSLS method were studied and analysed by Stentoft (2004, 2004) [230, 231]. Clement et al. (2002) [72] and Glasserman and Yu (2004) [113]

analysed the convergence of the method. Robustness of the algorithm was studied by Moreno and Navas (2003) [189]. Glasserman and Yu (2004) [112] studied the relation between the LSLS method and the stochastic mesh method of Broadie and Glasserman (2004) [49]. The choice of basis functions was studied in Areal et al. (2008) [14] and Webber (2011) [247]. Implementation issues of the method was also discussed in Webber (2011) [247]. Stentoft (2004) [230] showed how the LSM method can be extended to handle multiple stochastic factors. A comparison of the LSLS method with the method proposed by Carriere (1996) [61] was done by Stentoft (2008) [232].

There are a number of papers in the literature that use and extend the LSLS method. Chaudhary (2005) [63] implemented the LSLS method in conjunction with a quasi-Monte Carlo and a Brownian bridge method. The method can reduce memory requirements and can value high-dimensional American options. A similar method was investigated by Jonen (2009) [144]. Wang and Caflisch (2010) [246] proposed a method that obtained a regression equation for time t_0 by generating initial asset values from a chosen distribution. This equation can be differentiated to obtain hedging parameters for American options. Ibánez and Velasco (2010) [130] proposed a cost function method in which the cost of suboptimal exercise is minimised. Belomestny et al. (2009) [32] introduced a regression-based method with a consumption process. A consumption process is defined by the difference between an immediate exercise payoff and the option's continuation value. An upper bound and a lower bound for the option value are obtained using this method.

The LSLS method is also used in other finance applications. Examples include Beveridge and Joshi (2009) [35] and Trolle and Schwartz (2009) [237] who applied the method to Bermudan swaption, Rodriques and Armada (2006) [215] and Gamma (2003) [99] who considered a real option, Gamma (2003) [99] who incorporated stochastic stopping time constraints, which can be used to design executive stock option plans, *et cetera.*

4.3.5 Overview of the control variate method

The control variates technique is one of the various methods for improving the efficiency of Monte Carlo simulation. The method uses the error in estimating known quantities to reduce the error in estimating unknown quantities. This section provides the overview of the control variate method. Then, the use of control variates for American option valuation problem is reviewed in section 4.3.6.

4.3.5.1 Single control variate

Write $\widehat{v}_j$ as the simulated value of an American option along the jth path. Suppose that for each $j = 1, \ldots, M$, one simulates another quantity x_j along with v_j. Suppose that $(\widehat{v}_j, \widehat{x}_j)_{j=1,\ldots,M}$ are independent and identically distributed (i.i.d) and that $\mathbb{E}[x]$ is known. Then for a fixed ω, one computes a control variate estimate $\widehat{v}_j^{cv}$

$$\widehat{v}_j^{cv} = \widehat{v}_j - \omega\left(\widehat{x}_j - \mathbb{E}[x]\right) \tag{4.3.29}$$

The error of $\widehat{x}_j - \mathbb{E}[x]$ is used as a control of $\mathbb{E}[v]$ estimation. The variance of $\widehat{v}_j^{cv}$ is given as

$$\begin{aligned}
\mathrm{Var}[\widehat{v}_j^{cv}] &= \mathrm{Var}[\widehat{v}_j - \omega\left(\widehat{x}_j - \mathbb{E}[x]\right)] \\
&= \mathrm{Var}[\widehat{v}_j] + (-\omega)^2\mathrm{Var}[\widehat{x}_j] - 2\omega\mathrm{Cov}(\widehat{x}_j, \widehat{v}_j) \\
&= \sigma_{\widehat{v}}^2 - 2\omega\sigma_{\widehat{x}}\sigma_{\widehat{v}}\rho_{\widehat{x}\widehat{v}} + \omega^2\sigma_{\widehat{x}}^2 \equiv \sigma_\omega^2,
\end{aligned} \tag{4.3.30}$$

where $\sigma_{\widehat{x}}^2 = \mathrm{Var}[x]$, $\sigma_{\widehat{v}}^2 = \mathrm{Var}[\widehat{v}]$, and $\rho_{\widehat{x}\widehat{v}}$ is the correlation between $\widehat{x}$ and $\widehat{v}$.

The coefficient, ω^*, is chosen such that σ_ω^2 is minimised:

$$\frac{\mathrm{d}\sigma_\omega^2}{\mathrm{d}\omega} = -2\omega\sigma_{\widehat{x}}\sigma_{\widehat{v}}\rho_{\widehat{x}\widehat{v}} + 2\omega, \tag{4.3.31}$$

$$\omega^* = \frac{\sigma_{\widehat{v}}}{\sigma_{\widehat{x}}}\rho_{\widehat{x}\widehat{v}}, \tag{4.3.32}$$

$$\omega^* = \frac{\mathrm{Cov}[\widehat{x}, \widehat{v}]}{\mathrm{Var}[x]}. \tag{4.3.33}$$

Write $\bar{x} = \frac{1}{M}\sum_{j=1}^M \widehat{x}_j$ and $\bar{v} = \frac{1}{M}\sum_{j=1}^M \widehat{v}_j$. Substitute ω^* into (4.3.30), a ratio of the variance of the controlled estimator to that of the uncontrolled estimator is given by:

$$\frac{\mathrm{Var}[\bar{v} - \omega^*(\bar{x} - \mathbb{E}[x])]}{\mathrm{Var}[\bar{v}]} = 1 - \rho_{xv}^2, \tag{4.3.34}$$

or

$$\frac{\mathrm{Var}[\bar{v}]}{\mathrm{Var}[\bar{v}^{cv}]} = \frac{1}{1-\rho^2}. \tag{4.3.35}$$

It can be seen that the effectiveness of a control variate in terms of the variance reduction ratio in (4.3.34) depends on the correlation of the quantity being estimated, v, and the control variate, x. When using a control variate on the option value, it is referred to as the pricing control variate.

4.3.5.2 Multiple control variates

The method of control variates can be generalised to the case of multiple controls. In this case, suppose that on the jth path, one simulates a vector $\widehat{x}_j = \left(\widehat{x}_j^1, \ldots, \widehat{x}_j^L\right)^\top$ where L is the total number of controls and $A^\top$ denotes the transpose of vector A. Also suppose that the vector of expected values $\mu = \left(\mathbb{E}[x^1, \ldots, \mathbb{E}[x^L]\right)^\top$ is known. Assume that the pairs (x_j, v_j), $j = 1, \ldots, M$ are i.i.d. with the covariance matrix

$$\begin{pmatrix} \Sigma_{\widehat{x}\widehat{x}} & \Sigma_{\widehat{x}\widehat{v}} \\ \Sigma_{\widehat{x}\widehat{v}}^\top & \sigma_{\widehat{v}}^2 \end{pmatrix}, \tag{4.3.36}$$

where $\Sigma_{\widehat{x}\widehat{x}}$ is the $L \times L$ covariance matrix of $\widehat{x}$, $\Sigma_{\widehat{x}\widehat{v}}$ is the $L \times 1$ covariance vector, and $\sigma_{\widehat{v}}^2$ is the variance of $\widehat{v}_j$. For any fixed $\omega \in \mathbb{R}^L$, the control variate estimate is given as

$$\widehat{v}_j^{\text{cv}} = \widehat{v}_j - \omega^\top \left(\widehat{x}_j - \mu\right), \tag{4.3.37}$$

with a variance

$$\text{Var}\left[\widehat{v}_j - \omega^\top \left(\widehat{x}_j - \mu\right)\right] = \sigma_{\widehat{v}}^2 + \omega^\top \Sigma_{\widehat{x}\widehat{x}} \omega - 2\omega^\top \Sigma_{\widehat{x}\widehat{v}}, \tag{4.3.38}$$

which is minimised at

$$\omega^* = \Sigma_{\widehat{x}\widehat{x}}^{-1} \Sigma_{\widehat{x}\widehat{v}}. \tag{4.3.39}$$

Now define

$$R^2 = \frac{\Sigma_{\widehat{x}\widehat{v}}^\top \Sigma_{\widehat{x}\widehat{x}}^{-1} \Sigma_{\widehat{x}\widehat{v}}}{\sigma_{\widehat{v}}^2}. \tag{4.3.40}$$

The quantity in (4.3.40) is used to measure a linear relationship between $\widehat{x}_j$ and $\widehat{v}_j$. By substituting (4.3.39) into (4.3.38) and using (4.3.40), one has the variance of the control variate estimator

$$\text{Var}\left[\widehat{v}^{\text{cv}}\right] = \sigma_{\widehat{v}}^2 - \Sigma_{\widehat{x}\widehat{v}}^\top \Sigma_{\widehat{x}\widehat{x}}^{-1} \Sigma_{\widehat{x}\widehat{v}} = \left(1 - R^2\right) \sigma_{\widehat{v}}^2. \tag{4.3.41}$$

Therefore, from (4.3.41) one can see that R^2 assesses a proportion of the variance of $\widehat{v}$ that is removed by using x as control variates.

4.3.6 Control variate method for American option valuation

There are two main control variate methods that have been proposed in literature to value American options. They are reviewed as follows.

4.3.6.1 European option control variates

The first method is to choose the corresponding European options as control variates. This idea was first implemented by Broadie and Glasserman (1997) [50] with their random tree method. Write $\mathrm{c}^x = \widehat{x} - \mathbb{E}[x]$ for the second term on the right hand side of (4.3.29). c^x is the control part for an American option and is referred to as a control variate. Suppose one values an American put option and uses a corresponding European put option as a control variate. In this case, c^x becomes

$$\begin{aligned} \mathrm{c}_T^p &= \widehat{x} - \mathbb{E}[x] && (4.3.42) \\ &= \mathrm{e}^{-rT}(X - S_T)^+ - p_0 && (4.3.43) \end{aligned}$$

where $p_0 = p_0(S_0)$ is a time t_0 Black-Scholes value of a corresponding European put. The use of European options as a control variate for American options in the LSLS method was investigated and extended by Rasmussen (2005) [203]. The method is described as follows.

Exercise time control variate

Instead of sampling the control variates at option's expiry time, Rasmussen (2005) [203] made use of the optional sampling theorem and proposed sampling the control at the exercise time of the American option. Let τ be the exercise time of an American option. The control, c^x, becomes

$$\mathrm{c}_\tau^p = \begin{cases} \mathrm{e}^{-r\tau} p_\tau - p_0, & \text{if } \tau < T, \\ \mathrm{e}^{-r\tau}(X - S_\tau)^+ - p_0, & \text{Otherwise.} \end{cases} \qquad (4.3.44)$$

The optional sampling theorem states that a martingale stopped at a stopping time is a martingale. More specifically, suppose $\varsigma \leq \tau$ are two stopping times and let M be a martingale, then the theorem says that $M_\varsigma = \mathbb{E}[M_\tau | \mathcal{F}_\varsigma]$ (where $\mathcal{F}_\varsigma$ is the filtration). Since the European put $e^{-rt} p_t(S_t)$ is a martingale, it follows that $e^{-r\tau} p_\tau(S_t)$ is a martingale and $\mathbb{E}[p_\tau(S_t)]$ is equal $p_0(S(0))$. Throughout this thesis, c_τ^p is referred to as the put-tau control variate.

Rasmussen (2005) [203] showed that the method is proven to be effective. The correlation between the put-tau control variate and the American option value can be as high as 99%.

The control in (4.3.44) was applied by Broadie and Cao (2008) [47] with a primal-dual method to improve a lower bound. They also implement a sub-simulation to improve an estimate of the early exercise boundary.

Continuation value control variate

Rasmussen (2005) [203] proposed the control variate that is applied to option's continuation values. The idea of this control variate is to improve an estimated option's EEB.

Write $\widehat{C}_j^i$ for the American option's continuation value at the ith time step along the jth path. The controlled continuation value $\bar{C}_j^i$ is given as

$$\bar{C}_j^i = \widehat{C}_j^i - \omega c_j^i, \tag{4.3.45}$$

where c_j^i is the control variate and ω is given (4.3.33). This method tightens the option's continuation values which results in a more accurate early exercise boundary (EEB). A similar idea was also used by Broadie and Glasserman (2004) [49] with their stochastic mesh method. They employed mesh weights as a control for option's continuation values. The control variate in (4.3.45) is referred to as the rollback control variate and will be discussed in more detail in chapter 5.

4.3.6.2 Martingale control variates

The second type of control variate techniques for valuing American options is more complicated. The technique constructs a martingale to approximate a control variate (Henderson and Glynn (2002) [121]). This method uses a so called two-phase simulation technique (applied in American option pricing problems in Broadie and Glasserman (2004) [49], Rasmussen (2005) [203], and Bolia and Juneja (2005) [39], Ehrlichman and Henderson (2007) [89], and Juneja and Kalra (2009) [148]), which is first to estimate the option's EEB, and then simulate another independent sample path (usually a lot less paths than the first phase), exercised at the EEB found in the first phase.

The martingale-based method was proposed by Henderson and Glynn (2002) [121], and its application to American option pricing was undertaken by Bolia and Juneja (2005) [39], Ehrlichman and Henderson (2007) [89], Broadie and Cao (2008) [47] and recently by Juneja and Kalra (2009) [148]. Its application to multi-factor stochastic volatility was investigated in Fouque and Han (2004, 2004, 2006) [94, 95, 96].

The method of Bolia and Juneja (2005) [39] and Juneja and Kalra (2009) [148] relies on a particular choice of basis function such that the expected values of the control variates can be computed analytically. Ehrlichman and Henderson (2007) [89] employed a regression technique called Multivariate Adaptive Regression Splines (MARS) discussed by Friedman (1991) [97]. This is a method where an additional

basis function is automatically added if the improvement from adding the function is greater than a given amount, and the number of a basis function does not exceed a specified level.

4.4 American option valuation in stochastic volatility, Lévy and jump processes, and stochastic interest rate models

The numerical computation of values of American derivatives, when the underlying asset has stochastic volatility or is a Lévy process, has received significant attention in recent years. Various numerical methods have been proposed to price American options in these non-GBM models. These processes considered are stochastic volatility, Lévy and jump diffusion and stochastic interest rate processes.

4.4.1 Stochastic volatility

One of the most popular stochastic volatility models is the Heston model of Heston (1993) [122]:

$$\begin{aligned} \mathrm{d}S_t &= \mu S_t \mathrm{d}t + \sqrt{V_t} S_t \mathrm{d}z_t \\ \mathrm{d}V_t &= \kappa(\theta - V_t)\mathrm{d}t + \varepsilon\sqrt{V_t}\mathrm{d}w_t, \end{aligned} \tag{4.4.1}$$

where κ, θ, and ε are strictly positive mean reversion, long run volatility, and volatility and volatility parameters.

4.4.1.1 PDE based methods

Let u_t be an American option value at time t. When the volatility is stochastic, the standard Black-Scholes PDE in (2.2.1) becomes

$$\begin{aligned} -\frac{\partial u}{\partial t} &= \frac{VS^2}{2}\frac{\partial^2 u}{\partial S^2} + \rho\varepsilon VS\frac{\partial^2 u}{\partial S \partial V} + \frac{\varepsilon^2 V}{2}\frac{\partial^2 u}{\partial V^2} \\ &\quad + rS\frac{\partial u}{\partial S} + (\kappa[\theta - V] - \lambda V)\frac{\partial u}{\partial V} - ru, \end{aligned} \tag{4.4.2}$$

where λ is the market price of volatility risk. The main idea behind this method is to convert the equation (4.4.2) into a set of differential equations and solve them iteratively. However, in this case the problem is more computationally expensive than that of the GBM case because one has to take into account the volatility derivative term.

Zvan et al. (1998) [252] applied a penalty term to handle the American early exercise feature. They transformed equation (4.4.2) into a set of differential algebraic equations, which are solved by approximate Newton iteration; the method is convergent.

Another convergent method is the multi-grid approach. Its application to American option pricing with stochastic volatility was proposed by Clarke and Parrott (1999) [71]. They employed a finite difference method combined with a coordinate-stretching procedure based on the strike price to solve the linear complementary problem. Even though convergence is shown, the accuracy of the method depends on how the solution domain is truncated. Ikonen and Toivanen (2004) [131] proposed an operator splitting method. In this method, each time step is divided into two fractional time steps. Linear equations are solved in the first step and the early exercise constraint is enforced in the second step. The time convergence of the time discretisation scheme is shown. Another method called component-wise splitting was proposed by Ikonen and Toivanen (2006) [132]. The method decomposes the problem into three linear complementary problems. Each of these problems is solved by using a finite difference method (in their paper by an implicit Euler scheme and a Crank-Nicolson scheme). Numerical examples show that the method is convergent.

Even though the methods just described are convergent, they may be computationally expensive because one now has to solve a two-factor PDE in a two-dimension grid. Also, it may be difficult to add other state variables into an already higher dimension pricing problem.

4.4.1.2 Lattice methods

Guan and Xiaoqiang (2000) [118] and Vellekoop and Kieuwenhuis (2009) [241] proposed a lattice-based method to value American options in the stochastic volatility model of Heston (1993) [122]. Since volatility is now stochastic, the lattice becomes three-dimensional. Guan and Xiaoqiang (2000) [118] employed an interpolation-based approach (based on Finuance and Tomas [92] with the model of Hull and White (1987) [128]) that interpolated option values on neighbouring nodes whenever they are needed and not only on the nodes. Vellekoop and Kieuwenhuis (2009) [241] employed an idea similar to Guan and Xiaoqiang (2000) [118] but used bicubic spline interpolation to ensure the continuity of the first derivative of the option value with respect to the asset value and volatility in order to improve a rate of convergence.

4.4.2 Lévy and jump processes

Write $u = u(t, S)$ as the value of an option at time t. When introducing jumps into the option pricing problem the Black-Scholes PDE in (2.2.1) becomes the PIDE. The value of an option at time t, u_t, is satisfied by the following PIDE (Cont and Tankov (2004) [73]):

$$\begin{aligned} -\frac{\partial u}{\partial t} &= rS\frac{\partial u}{\partial S} + \frac{\sigma^2 S^2}{2}\frac{\partial^2 u}{\partial S^2} - ru \\ &\quad + \int \nu(dy)\left[\hat{u} - u - S(e^y - 1)\frac{\partial u}{\partial S}\right], \end{aligned} \tag{4.4.3}$$

where $\hat{u} = u(t, Se^y)$, and ν is a Lévy density.

4.4.2.1 PIDE methods

There exists a number of PIDE methods for the pricing of American options under diffusions with Poisson jumps or more general Lévy processes. They are discussed as follows.

Finite difference methods have been used in several papers, including Carr and Hirsa (2003) [57], Hirsa and Madan (2004) [123], Almendral (2004) [10], and Levendorskiĭ et al. (2005) [165]. Carr and Hirsa (2003) [57] and Hirsa and Madan (2004) [123] developed a numerical procedure for the Variance Gamma process. Carr and Hirsa (2003) [57] worked with forward equations. Hirsa and Madan (2004) [123] employed a different method. They used two methods to discretise the integral term in (4.4.3). They worked with a transformed state variable, $y = \ln(S)$. They first expanded the integrand near its singularity ($y = 0$) and implemented an implicit method to this part. The rest of the integral is treated explicitly. The method is convergent. Almendral (2004) [10] formulated a pricing problem as a Linear Complementary problem and applies standard finite differences. He also implemented the Fast Fourier Transform (FFT) to discretise the jump term. Levendorskiĭ et al. (2005) [165] developed a new finite difference scheme based on the Randomisation procedure of Carr (1998) [56]. The main idea is to not approximate small jumps by a diffusion but by an additional drift. The method is convergent. They found that their method produced a more stable EEB than that of Hirsa and Madan (2004) [123], and consumed less computational time.

A stretch grid transformation was employed by Almendral and Oosterlee (2006) [11] for the Variance Gamma process to obtain higher accuracy in the numerical integration process of the integral term of a PIDE. The method is similar to that of Clarke and Parrott (1999) [71]. The method is convergent, and by comparing with

results from Hirsa and Madan (2004) [123], they found that they can achieve the same order accuracy with a fewer grid point (400 × 20 grids instead of 2,000 × 1,000 grids). Nevertheless, as pointed out by the authors, their grid transformation makes it impossible to apply fast Fourier transform (FFT) and hence the method cannot take advantage of FFT.

A variation inequality method is used in Matache et al. (2005) [180] and Zhang (1997) [251] for the CGMY process of Carr et al. (2002) [60] and jump diffusion model of Merton (1976) [183] respectively. Matache et al. (2005) [180] used the wavelet Galerkin discretisation scheme, and also suggested removing the drift in order to satisfy the necessary condition. However, as pointed out by Levendorskiĭ et al. (2005) [165], by removing drift, the payoff functions will become time-dependent and the dependence on t needs to be frozen at each time step and this may introduce additional errors.

Another PIDE method is a penalty method. This method is used to price American options by Wang et al. (2007) [243] and d'Halluin et al. (2003) [86]. Wang et al. (2007) [243] applied this method with the particular iterative method to price American options under the CGMY model. d'Halluin et al. (2003) [86] worked in a jump diffusion model of Merton (1976) [183] and used an iterative method with a FFT method to compute a jump diffusion integral term. Both of the methods show convergence.

4.4.2.2 Semi-analytical methods

The use of Fourier analysis in pricing American options in non-GBM frameworks was discussed in Chiarella and Ziogas (2005, 2006) [64, 65] and Chiarella et al. (2006) [67]. Chiarella and Ziogas (2005) [64] employed a Fourier Hermite series expansion to value an American call in a jump diffusion process. At each time step, Fourier Hermite expansion is applied and the exercise boundary and call values are solved iteratively. However, as pointed out by the authors, the main shortcoming for this method is its incapability to provide an accurate early exercise boundary near the time of expiry. Chiarella and Ziogas (2006) [65] and Chiarella et al. (2006) [67] extended the incomplete Fourier transform approach of McKean (1965) [181] and derived an integral representation of the EEB in jump-diffusion and stochastic volatility models respectively. As pointed out by the authors in the stochastic volatility case the convergence is slow and the iteration scheme needs to be improved.

4.4.2.3 Lattice methods

Kellezi and Webber (2004) [153] applied a lattice method to value Bermudan and American options when an underlying process follows a Lévy process. They constructed a lattice from the transition density function of the Lévy process, and also proposed several constructions for branching probabilities of a lattice. Their method gives prices of Bermudan and bounds of American options in VG and NIG models.

4.4.3 Stochastic interest rate

Boyarchenko and Levendorskiĭ (2007) [40] employed a regime switching method based on the Randomisation method of Carr (1998) [56] to value perpetual American options. Ho et al. (1997) [124] and Chung (1999, 2002) [70, 69] employed the GJ extrapolation method of Geske and Johnson (1984) [104]. Ho et al. (1997) [124] applied an extrapolation method on option values from multi-variate binomial approximations. Chung (1999, 2002) [70, 69] extended the method of Ho et al. (1997) [124] to value American options using only the value of European options and an option with two exercise dates, which involved a distribution of forward price. They proposed both a numerical integration method and a binomial approximation to estimate this forward distribution. Lindset and Lund (2007) [167] proposed a simulation method combined with the two-point GJ-method. Like Chung's method, they extrapolated from options with one and two exercise dates. The method also uses an option with two exercise dates as the control variate. Nevertheless, they did not sample the control variate at exercise times.

4.5 American barrier and power options

American barrier options are exotic options with the early exercise feature. Let τ be the exercise time of an American option and let τ_b be the hitting time to the option barrier b_t. Suppose the barrier option matures at time T, and the underlying American fractional power option matures at time $T_1 \geq T$. American barrier options can be divided into two types depending on whether they are in or out types:

1. American knock-in: American option comes in to existence when the option barrier b_t is hit at τ_b and a holder can exercise early at time τ where $\tau \in [\tau_b, T_1]$.

2. American knock-out: a holder can exercise an American option at time τ where $\tau \in [0, \tau_b)$. An American option ceases to exist after time τ_b.

Power option is an option in which its payoff is raised to some power. This type of option generalises a piecewise linear payoff of the standard option (where a power is 1).

This section provides a brief review of valuation methods for pricing American barrier options and power options.

4.5.1 American barrier

Gao et al. (2000) [100] employed a quasi-analytical approach to value a flat barrier American option. They decomposed the value of the American barrier option into a standard European barrier option and an early exercise premium (similar to the method of Barone-Adesi and Whaley (1987) [27]) and then presented an approximation of the option value. Haug (2001) [119] employed the reflection principle and obtained an approximation for American barrier options. Numerical results from Dai and Kwok (2004) [79] suggest that Haug's approximation may work only when the barrier is less than or equal to the American option's exercise price. Dai and Kwok (2004) [79] then proposed an approximation that decomposes the American barrier option into standard American options, European option and a plain vanilla knock-in option. The method of Dai and Kwok (2004) [79] can value options with a wide range of barrier levels.

Ait-Sahlia et al. (2003) [5] investigated a hybrid method which combines the decomposition technique of Gao et al. (2000) [100] with the Bernoulli walk method of Ait-Sahlia and Lai [4]. The method approximates the early exercise boundary of an American knock-out option by continuous piecewise exponential functions. Ait-Sahlia et al. (2003) [6] modified a lattice method by using a Bernoulli random walk to value American knock-in options. Even though these methods can value American barrier options with a constant barrier, it is not clear how they can value an option with a non-constant barrier.

4.5.2 Power option

A power call option pays

$$\max\left(S(T) - X, 0\right)^{\kappa}, \tag{4.5.1}$$

where $\kappa \in \mathbb{R}$, $S(T)$ is an asset value at time T and X is an exercise price. When $\kappa \in \mathbb{Z}$, Esser (2003) [90] derived the analytical formula in terms of sum series. However, when $\kappa \notin \mathbb{Z}$, numerical methods must be used.

4.6 Conclusion

This chapter has reviewed a number of valuation methods for American options, both in GBM and non-GBM models. It has particularly focused on the use of

Monte Carlo simulation methods to value American options. Simulation methods are flexible and can handle a problem where the number of state variables is large. However, Monte Carlo simulation is generally computationally expensive. Therefore, variance reduction techniques must be used in order to implement Monte Carlo simulation methods effectively.

This chapter has also provided a review of several control variate methods that have been used to reduce the variances of American option values found by simulation. There are two main methods; the martingale approach and the use of a suitable option as a control variate. The latter is the primary focus of the next chapter, chapter 5. It is concerned with the development of a new control variate method for valuing American options.

Chapters 6 and 7 propose a new simulation method that generalises the standard LSLS method. The new method will be used to value standard American put options and exotic American options.

Chapter 5

Valuing Bermudan and American Put Options with Bermudan Put Option Control Variates

This chapter is concerned with the valuation of American and Bermudan options using Monte Carlo simulation. It presents a simple yet effective control variate method that is based on the use of Bermudan put options as control variates. The control variate method is used with the Least Square Monte Carlo simulation method of Longstaff and Schwartz (2001) [176] (LSLS) to value Bermudan and American put options. By choosing a Bermudan put with two exercise times, one is able to write it in terms of a compound call option whose analytical solution is known.

The chapter illustrates the use of two-phase simulation methods in which an option's early exercise boundary is estimated in the first phase using M_1 sample paths and then an option value is computed with pricing control variates, using M_2 sample paths.

To approximate the value of American put options, Richardson extrapolation is implemented on three different values of Bermudan options. Numerical results suggest that the approximated American values are accurate and the Bermudan control variate can gain significant speed ups.

The chapter is organised as follows. Section 5.1 describes the difference between the Bermudan and the American option and also the convergence of Bermudan option values to the American option value. Section 5.2 presents the Bermudan put control variates. The proposed methodology is presented in section 5.3, and sec-

tion 5.4 illustrates the use of Richardson extrapolation to estimate American put options. Section 5.5 shows numerical results. Section 5.6 concludes.

5.1 Exercise at fixed times: Bermudan options

It is important to note the difference between American options and Bermudan options. American option can be exercised at any points in time $t \in [0, T]$. In contrast, Bermudan options can be only exercised at a fixed set of exercise dates, $0 = t_0 < t_1 < t_2 < \ldots < t_E = T$.[1] Hence, when using numerical methods to value an option, if time is discretised into E exercise dates, one is effectively valuing a Bermudan option. As $E \to \infty$, a Bermudan option value converges to a corresponding American option value. Figure 5.1 plots the convergence of Bermudan option values to an American option value using a trinomial lattice method.

Parameter values in figure 5.1 are $S = 100$, $X = 100$, $r = 0.05$, $\sigma = 0.1$, $T = 1$, and number of time steps $N = 204,800$. The American put value is 2.43679.

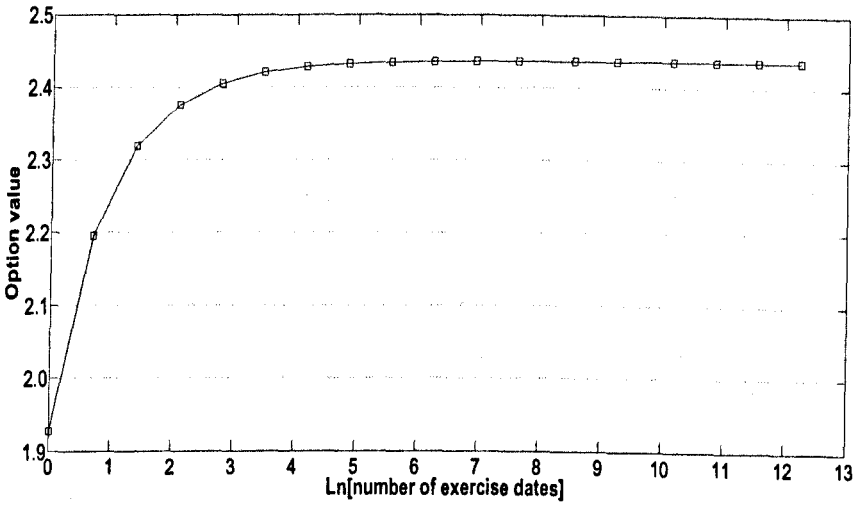
Panel (a) in figure 5.1 shows Bermudan put values with a different number of exercise dates E on log scale, $E = 1, 2, \ldots, 204800$, and a number of time step N is 204,800. One can see that as the number of exercise dates increases, the Bermudan put value convergences to its limiting value, an American put.

Panel (b) in figure 5.1 shows the error convergence. It plots $\ln(v_E^b - v^\infty)$, where v_E^b and v^∞ are the Bermudan put value and the American put value respectively, against $\ln(E)$. It shows that the Bermudan put option converges uniformly to the American put option approximately at the rate ~ 1. With this convergence rate, American put value can be computed using Richardson extrapolation. This value corresponds to a Bermudan value with $E = 204,800$. This is a very large number of E; it would be computationally prohibitively expensive to implement accurately with a Monte Carlo simulation method. However, section 5.4 will illustrate the use of extrapolation techniques with Monte Carlo values to obtain an American put value from Bermudan values at small values of E.

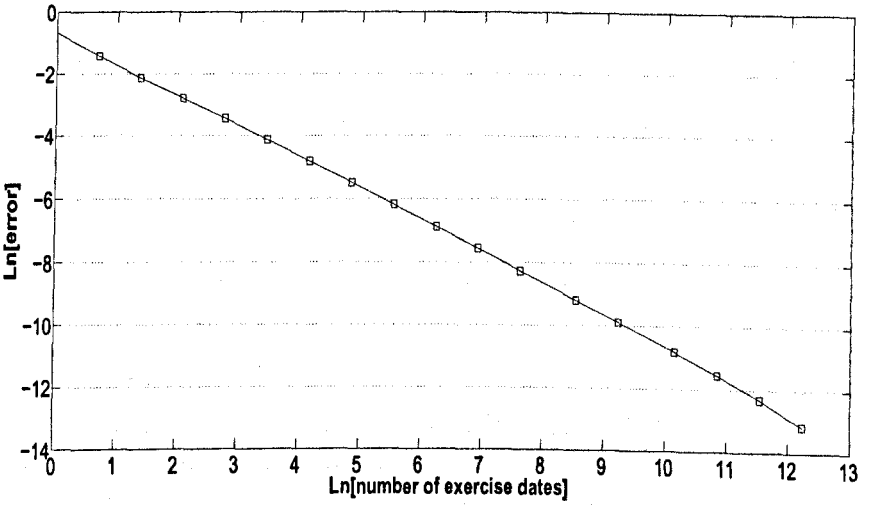
5.2 Bermudan option control variate

As discussed in section 4.3.6, a good control variate should be a quantity that is highly correlated to the American put option. A simple choice would be the corresponding European put option because it is straightforward to compute the

[1] Even though time intervals $\{t_i - t_{i-1}\}_{i=1,\ldots,E}$ do not have to be equal for Bermudan options, they are assumed to be equal through this chapter.



(a) Convergence of Bermudan option values to the American option value



(b) Convergence: ln(error) against ln(number of exercise dates)

Figure 5.1. Convergence of a Bermudan option value to an American option value from a trinomial lattice method

value of European put along simulated paths.

Although the European put might be a suitable choice of control variate, one would expect the Bermudan put option to be more effective (as its value might be more highly correlated with that of the American put). Typically, there exists no closed form solution for the Bermudan put option because of its early exercise possibilities. However, for Bermudan put options with only two exercise times ($T_1 < T_2 = T$), an analytical solution can be obtained in terms of the compound call option.[2] This is shown in Theorem 4.

Theorem 4. *Consider the Bermudan put option maturing at time T, strike X, with two exercise times $T_1 < T_2$. Assume that the interest rate r and volatility σ are constant. The value of this Bermudan option, b_t, at time $t < T_1$ is*

$$b_t = \gamma_t + e^{-r(T_1-t)}X - S_t, \tag{5.2.1}$$

where γ_t is the time t compound call option. The compound call option exercises at time T_1 with strike X_1, exercising into a call option with strike X_2 exercising at time T_2 where $X_1 = X - e^{-r(T_2-T_1)}X$ and $X_2 = X$.

Proof. Denote the European put option maturing at time T_2 with strike X as p_t. The value of the Bermudan option at time T_1, b_{T_1} is

$$\begin{aligned} b_{T_1} &= \max\left(X - S_{T_1},\ p_{T_1}\left(X, T_2\right)\right), &(5.2.2)\\ &= \max\left(0,\ p_{T_1}\left(X, T_2\right) + S_{T_1} - X\right) + X - S_{T_1}. &(5.2.3)\end{aligned}$$

Let c_{T_1} be a corresponding call option. By put-call parity, one has

$$b_{T_1} = \max\left(0,\ c_{T_1}\left(X, T_2\right) + e^{-r(T_2-T_1)}X - X\right) + X - S_{T_1}, \tag{5.2.4}$$

where $e^{-r(T_2-T_1)}X$ is the value at time T_1 of X received at time T_2. Next, one observes that

$$\max\left(0,\ c_{T_1}\left(X, T_2\right) + e^{-r(T_2-T_1)}X - X\right) \tag{5.2.5}$$

is the payoff at time T_1 of a compound call option with strike $X_1 = X - e^{-r(T_2-T_1)}X$ exercising at time T_1 call maturing at time T_2 with strike $X_2 = X$. Let γ_t be the value of this compound call option, then one can write the Bermudan option value

[2] A similar control was proposed in Lindset and Lund (2007) [167] in the stochastic interest rate model. Nevertheless, they used the non-exercise time control variate. The two-point Richardson extrapolation was also used.

at time t as

$$b_t = \gamma_t + e^{-r(T_1-t)}X - S_t, \tag{5.2.6}$$

as required. □

The analytical value of γ_t is (Geske (1979) [103])

$$\gamma_t = S_t \mathrm{N}_\rho(d_1, e_1) - X_2 e^{-r(T_2-T_1)} \mathrm{N}_\rho(d_2, e_2) - X_1 e^{-rT_1} N(e_2), \tag{5.2.7}$$

where $\mathrm{N}_\rho(d, e)$ is the bivariate normal distribution function with correlation ρ,

$$d_1 = \frac{1}{\sigma\sqrt{T_2}} \ln\left(\frac{S_t}{X_2 e^{-rT_2}}\right) + \frac{1}{2}\sigma\sqrt{T_2}, \tag{5.2.8}$$

$$e_1 = \frac{1}{\sigma\sqrt{T_1}} \ln\left(\frac{S_t}{S^*_{T_1} e^{-rT_1}}\right) + \frac{1}{2}\sigma\sqrt{T_1}, \tag{5.2.9}$$

with $d_2 = d_1 - \sigma\sqrt{T_2}$, $e_2 = e_1 - \sigma\sqrt{T_1}$, $\rho = \sqrt{\frac{T_1}{T_2}}$, and $S^*_{T_1}$ is a critical asset price such that

$$c_t\left(S^*_{T_1}, X_2, T_2 - T_1\right) = X_1. \tag{5.2.10}$$

$S^*_{T_1}$ is the value that determines if a compound call option is exercised at time T_1. If $S_{T_1} > S^*_{T_1}$, then $c_{T_1}(S_{T_1}, X_2, T_2 - T_1) > X_1$ and a call is exercised. For a Bermudan put option with two exercise times, $S^*_{T_1}$ satisfies

$$X - S^*_{T_1} = p_{T_1}\left(S^*_{T_1}, X\right), \tag{5.2.11}$$

and an option is exercised at time T_1 if $S_{T_1} \leq S^*_{T_1}$. The value of $S^*_{T_1}$ can be obtained by root-finding algorithms.

5.2.1 Bermudan put-terminal control variate

Denote the time t value of a Bermudan put option maturing at time T with two exercise times, $T_1 < T_2 = T$, by $b2_t$ and its value at time t_0 by $b2_0$. The option $b2_t$ can be exercised at time T_1 if $S_{T_1} \leq S^*_{T_1}$ for some critical stock price $S^*_{T_1} < X$. Write $T = T_2$. At time T_1, if $S_{T_1} \leq S^*_{T_1}$, the option is exercised and then ceases to exist. If $S_{T_1} > S^*_{T_1}$, then the option continues to exist and can be exercised at time T.

With this exercise structure, the simplest version of the $b2_t$ control variate, c^{b2}_T, is

$$\mathrm{c}^{b2}_T = \begin{cases} \mathrm{e}^{-rT_1}(X - S_{T_1}) - b2_0, & \text{if } S_{T_1} \leq S^*_{T_1}, \\ \mathrm{e}^{-rT}(X - S_T)^+ - b2_0, & \text{if } S_{T_1} > S^*_{T_1}. \end{cases} \tag{5.2.12}$$

The control variate c_T^{b2} is referred to as the Bermudan put-terminal control variate. This is because the value of $b2_t$ is used only at times T_1 and T. $S_{T_1} \leq S_{T_1}^*$ means $b2_t$ is exercised at time T_1 and its holder receives $X - S_{T_1}$. If $S_{T_1} > S_{T_1}^*$, $b2_t$ is effectively a standard put option and its value at time T is $(X - S_T)^+$.

The Bermudan put control variate, $b2_t$, can be used as the exercise time version discussed in section 4.3.6. This will be described next.

5.2.2 Bermudan put-tau control variate

Rasmussen (2005) [203] showed that by sampling a European put at the exercise time of an American put, the control variate becomes more effective than sampling a European put at maturity time T.

Let τ be the exercise time of the Bermudan put option being valued. The value of $b2_t$ can be used at τ instead of using only at T_1 and T. The value of control variate, c^{b2}, will be different depending on values of τ and $S_{T_1}^*$. In particular, there are four different cases to consider. They are summarised in table 5.1.

Values of τ and $S_{T_1}^*$	Situations
$\tau < T_1$	The EEB is hit before time T_1.
$T_1 \leq \tau < T$ and $S_{T_1} \leq S_{T_1}^*$	The EEB is hit after time T_1 but the option was exercised at time T_1.
$T_1 \leq \tau < T$ and $S_{T_1} > S_{T_1}^*$	The EEB is hit after time T_1 and the option was not exercised at time T_1.
$\tau \geq T$ and $S_{T_1} > S_{T_1}^*$	The EEB is not hit and the option was not exercised at time T_1.
$\tau \geq T$ and $S_{T_1} \leq S_{T_1}^*$	The EEB is not hit but the option was exercised at time T_1.

Table 5.1. Four cases of the Bermudan put control variate, c^{b2}

Write p_t for the time t value of a European put option. With these cases, the exercise time Bermudan put can be used as a control variate:

$$c_\tau^{b2} = \begin{cases} e^{-r\tau} b2_\tau - b2_0, & \text{if } \tau < T_1, \\ e^{-rT_1}(X - S_{T_1}) - b2_0, & \text{if } S_{T_1} \leq S_{T_1}^*, \\ e^{-r\tau} p_\tau - b2_0, & \text{if } T_1 \leq \tau < T \text{ and } S_{T_1} > S_{T_1}^*, \\ e^{-rT}(X - S_T)^+ - b2_0, & \text{if } \tau = T \text{ and } S_{T_1} > S_{T_1}^*. \end{cases} \tag{5.2.13}$$

Note that when $S_{T_1} \leq S_{T_1}^*$, the option is exercised regardless whether the EEB is hit or not. This is shown by the second case in (5.2.13). When $\tau < T_1$, $b2_t$ still exists and its value at τ, $b2_\tau$, is used as a control. When $S_{T_1} \leq S_{T_1}^*$, $b2_t$ is exercised at T_1

and the payoff $X - S_{T_1}$ is used. If $T_1 < \tau < T$ and $S_{T_1} > S^*_{T_1}$, $b2_t$ becomes a put and its value at the exercise time, p_τ, is used. Lastly, when $\tau = T$ and $S_{T_1} > S^*_{T_1}$, $b2_t$ is not exercised at T_1. It becomes a European put and its value at maturity time T, $(X - S_T)^+$, is used. c^{b2}_τ is referred to as the Bermudan put-tau control variate.

The twice-exercise Bermudan put option, $b2_t$, can also used in several other ways which yield three versions of the c^{b2} control variate. These are shown next.[3]

5.2.2.1 The plain Bermudan T_1 control variate, $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1}$

The plain version of the Bermudan control variate is given by

$$\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1} = \begin{cases} \mathrm{e}^{-r\tau} b2_\tau - b2_0, & \text{if } \tau < T_1, \\ \mathrm{e}^{-rT_1}(X - S_{T_1}) - b2_0, & \text{if } S_{T_1} \le S^*_{T_1}, \\ \mathrm{e}^{-rT}(X - S_T)^+ - b2_0, & \text{if } T_1 \le \tau < T \text{ and } S_{T_1} > S^*_{T_1}. \end{cases} \tag{5.2.14}$$

When $\tau < T_1$ (first row in (5.2.14)) and $S_{T_1} \le S^*_{T_1}$ (second row in (5.2.14)), $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1} = \mathrm{c}^{b2}_\tau$. Nevertheless, $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1}$ is different from c^{b2}_τ in two ways. First, the value of $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1}$ is not defined at T. Second, the exercise time τ used in $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1}$ is not allowed to be greater than T_1. This is why the value of a payoff at time T_1, $(X - S_T)^+$ is used when $T_1 \le \tau < T$ and $S_{T_1} > S^*_{T_1}$.

5.2.2.2 The Bermudan T_1 control variate, $\mathrm{c}^{b2}_{\tau,T_1}$

$$\mathrm{c}^{b}_{T_1} = \begin{cases} \mathrm{e}^{-r\tau} b2_\tau - b2_0, & \text{if } \tau < T_1, \\ \mathrm{e}^{-rT_1}(X - S_{T_1}) - b2_0, & \text{if } S_{T_1} \le S^*_{T_1}, \\ \mathrm{e}^{-rT} p_{T_1} - b2_0, & \text{if } T_1 \le \tau < T \text{ and } S_{T_1} > S^*_{T_1}. \end{cases} \tag{5.2.15}$$

$\mathrm{c}^{b}_{T_1}$ is a slightly improved version to $\mathrm{c}^{b,\mathrm{pl}}_{\tau,T_1}$. When $T_1 \le \tau < T$ and $S_{T_1} > S^*_{T_1}$, it uses a value of a European option at time T_1, p_{T_1}, instead of the payoff $(X - S_T)^+$.

5.2.2.3 The early Bermudan control variate, $\mathrm{c}^{b2,\mathrm{e}}_\tau$

First define the portfolio

$$B_t = \{b2_t - p'_t\left(S_t | S^*_{T_1}, T_1\right) - \left(X - S^*_{T_1}\right) d_t\left(S_t | S^*_{T_1}, T_1\right)\}, \tag{5.2.16}$$

[3]These control variates appeared in Chirayukool and Webber (2010) [68] and are also reported in Webber (2011) [247].

where $p'_t\left(S_t|S^*_{T_1}, T_1\right)$ is the European put maturing at time T_1 with strike $S^*_{T_1}$, and $d_t\left(S_t|S^*_{T_1}, T_1\right)$ is a digital option with payoff h_{T_1}:

$$h_{T_1} = \begin{cases} 1, & \text{if } S_{T_1} \leq S^*_{T_1}, \\ 0, & \text{if } S_{T_1} > S^*_{T_1}. \end{cases} \tag{5.2.17}$$

The portfolio B_t can be viewed as a knock-out put option with a payoff $X - S_T$ maturing at time T. It knocks out at time T_1 if $S_{T_1} \leq S^*_{T_1}$. The put and the digital payoff cancel out the Bermudan put payoff. This control variate removes the cash flow at time T_1 and instead takes the cash flow at the earlier time $t < T_1$.

The early Bermudan control variate, $\mathrm{c}_\tau^{b2,\mathrm{e}}$, is defined as follows.

$$\mathrm{c}_\tau^{b2,\mathrm{e}} = \begin{cases} \mathrm{e}^{-r\tau}B_\tau - B_0, & \text{if } \tau < T_1, \\ -B_0, & \text{if } S_{T_1} \leq S^*_{T_1}, \\ \mathrm{e}^{-r\tau}p_\tau - B_0, & \text{if } T_1 \leq \tau < T \text{ and } S_{T_1} > S^*_{T_1}, \\ \mathrm{e}^{-rT}\left(X - S_T\right)^+ - B_0, & \text{if } \tau = T \text{ and } S_{T_1} > S^*_{T_1} \end{cases} \tag{5.2.18}$$

Choice of T_1

Throughout this chapter, a value of T_1 is to be $T_1 = \frac{T}{2}$. This is clearly not the only possible choice. The only restriction on T_1 is that $T_1 \in [0, T]$. In fact, T_1 can be chosen such that a value of $b2_t$ is maximised, which may increase a correlation between $b2_t$ and Bermudan option being valued. The method of choosing T_1 to maximise $b2_t$ was first purposed by Bunch and Johnson (1992) [53] to approximate a value of American option by using two-point Richardson extrapolation. The method was extended to value American options in stochastic interest rate model by Lindset and Lund (2007) [167]. Numerical results from Lindset and Lund (2007) [167] showed that, for out-of-the-money options, an option value is maximised when T_1 is closed to T. For at-the-money options, an option value is maximised when T_1 is approximately closed to $\frac{3}{4}T$ and for in-the-money options, an option value is maximised when T_1 is slightly less than $\frac{1}{2}T$.

Even though a value of the Bermudan option with two exercise dates can be maximised by carefully choosing T_1, this method is not pursued in this thesis and is left for future research.

5.2.3 Obtaining values of τ_j from simulation

Recall from section 4.3.4 on page 77 that h_j^i and $\widetilde{C}_j^i$ are the option's payoff and its approximated continuation value at the ith time step on the jth path respectively. When implementing the exercise time control variate, to obtain the exercise time $\tau_j, j = 1, \ldots, M$ of a Bermudan option, write τ_j^i for the earliest exercise time on the jth path for the Bermudan put being valued after time t_i, conditioned on not being exercised before time t_i,

$$\tau_j^i = \min_t \left\{ t_k \geq t_i \mid h_j^k > \widetilde{C}_j^k \right\}, \tag{5.2.19}$$

where h_j^k is an immediate exercise payoff and $\widetilde{C}_j^k$ is the approximated continuation value at time step k on the jth path. Then set $\tau_j = \tau_j^i$. The exercise time control variate on the jth path is

$$\mathrm{c}_{\tau_j}^x = \mathrm{e}^{-r\tau_j} \widehat{x}_{\tau_j} - \mathbb{E}[x], \tag{5.2.20}$$

for any control variate $\widehat{x}$. For instance, $\widehat{x}$ can be p_τ or $b2_\tau$.

5.3 Two-phase simulation method

A two-phase simulation method separates the simulation for valuing Bermudan option into two phases. The first phase is to use a set of M_1 sample paths to estimate the option's EEB. This is done by applying steps 1 to 4 of section 4.3.4.1, page 79.

Once the option's early exercise boundary is estimated, the second phase generates a new set of M_2 sample path to use it to value an option based on the exercise boundary estimated in the first phase.

The advantage of this method is that the second set of simulated paths do not involve an expensive regression algorithm and is then less computationally intensive. This means that standard errors can be reduced at a slight additional cost.

5.3.1 First phase: Estimating the early exercise boundary

It is very important to estimate an option's early exercise boundary accurately in order to obtain an accurate option value. To ensure there is less bias in the approximated option's continuation values, one can apply a control variate to the option's continuation value itself. This method is referred to as the rollback control variate and was briefly described in section 4.3.6. This method is discussed in more detail now.

5.3.1.1 The rollback control variate

Recall from section 4.3.6 that the controlled continuation value at the ith time step on the jth path, $\bar{C}_j^i$, is given by

$$\bar{C}_j^i = \widehat{C}_j^i - \omega \mathrm{c}_j^i, \tag{5.3.1}$$

where $\widehat{C}_j^i$ is the American option's continuation value at the ith time step along the jth path, c_j^i is the control variate and ω is given (4.3.33). Then, with $\bar{C}_j^i$, find parameters $\bar{a}_i$ so that $\widetilde{C}_j^i = f\left(S^i \mid \bar{a}^i\right)$ where $f\left(S^i \mid \bar{a}^i\right)$ is given by (4.3.24). $\widetilde{C}_j^i$ optimally fits $\bar{C}_j^i$ and is used to determine the EEB of an option in step 4 on page 79.

The EEB that is obtained from $\bar{C}_j^i$ is an improvement over that obtained from those without the rollback control variate (see Rasmussen (2005) [203], Broadie and Glasserman (2004) [49] and Ehrlichman and Henderson (2007) [89]).

Since the rollback control variate is used to improve option's continuation values, it is implemented at each time step. The use of the put-tau rollback control variate was discussed by Rasmussen (2005) [203]. Let τ be the exercise time of a Bermudan option being valued conditional on the option is not being exercised before time t. The $\mathrm{c}_{\tau,t}^p$ rollback is[4]

$$\mathrm{c}_{\tau,t}^p = \begin{cases} \mathrm{e}^{-r(\tau-t)}p_\tau - p_t, & \text{if } \tau < T, \\ \mathrm{e}^{-r\tau}\left(X - S_\tau\right)^+ - p_t, & \text{Otherwise.} \end{cases} \tag{5.3.2}$$

The Bermudan put-tau control variate, c_τ^{b2}, discussed in section 5.2.2 can be used as the rollback control variate. Since the value of $b2_\tau$ can be obtained only at $t < T_1$, it is desirable to combine it with the put-tau control variate. The combined control variate, c_τ^{p+b2}, is

$$\mathrm{c}_{\tau,t}^{p+b2} = \begin{cases} \mathrm{e}^{-r(\tau-t)}p_\tau - p_t + \mathrm{e}^{-r(\tau-t)}b2_\tau - b2_t, & \text{if } \tau < T_1 < T, \\ \mathrm{e}^{-r(\tau-t)}p_\tau - p_t, & \text{if } T_1 < \tau < T, \\ \mathrm{e}^{-r\tau}\left(X - S_\tau\right)^+ - p_t, & \text{if } \tau = T. \end{cases} \tag{5.3.3}$$

5.3.1.2 S_0-dispersion technique

Another technique that can be used to improve an estimation of the option's EEB is the S_0-dispersion technique proposed by Rasmussen (2005) [203]. The aim of the method is to increase the number of in-the-money paths in the early stage of an

[4]The subscript t means that this is the value of a control variate at time t.

option's life, resulting in a smoother option's EEB. The method samples S_0 from an initial distribution rather than a single staring point. In particular, it samples from S_0 at the fictitious time $-t_d^* < 0$. Instead of generating a sample path from time $t = 0$, this method generates a sample path from time $-t_d^* < 0$.

In other words, an initial asset value S_0 will be shifted away from the origin (toward $-t_d^*$) and this will result in more sample paths closer to the EEB in the early part of an option's life. This method can be done because the option's EEB is independent of an initial asset value S_0.

Let τ_d^* be the fictitious initial time, the new initial asset value S_0^d is given by

$$S_0^d = S_0 \exp\left(-\frac{1}{2}\sigma^2\tau_d^* + \sigma\sqrt{\tau_d^*}\epsilon\right), \tag{5.3.4}$$

where $\epsilon \sim N(0,1)$. Throughout this chapter τ_d^* is set to 0.5.[5]

5.3.1.3 Choices for basis functions

It is important to describe basis functions that will be used to approximate continuation values in (4.3.24). Two types of basis functions are considered. They are as follows.

Orthogonal polynomials

A system of polynomials $\{\psi_n(x)\}$ with degree n is orthogonal on $[a, b]$ if it satisfies an orthogonality relation

$$\int_a^b w(x)\psi_m(x)\psi_n(x)\mathrm{d}x = \delta_{mn}c_n, \tag{5.3.5}$$

where $w(x)$ is a weighting function, δ_{mn} is the Kronecker delta and c_n is a normalisation constant. There are several orthogonal polynomials to choose from such as Hermite, Legendre, Laguerre, *et cetera.* In this thesis, Leguerre basis function will be used. These have the following characteristics (Abramowitz and Stegun (1965) [1]).

a	b	$w(x)$	c_n	Recurrence relationship
0	∞	e^x	1	$(n+1)\psi_{n+1}(x) = (2n+1-x)\psi_n(x) - n\psi_{n-1}(x)$

Table 5.2. Laguerre polynomial characteristics

Since $\psi_n(x) \sim x^n$ to leading order, when an asset value is large, says $x = 100$,

[5]This value of τ_d^* was suggested by Rasmussen (2005) [203].

$\psi_n(x) \sim 100^n$. Suppose one uses $n = 12$, $\psi_n(x)$ will be very large. Suppose that the continuation value is $\sim$10. It is clear that using a basis function with $\sim 100^{12}$ to fit a curve with a much smaller value of $\sim$10 is likely to fail. To work around this problem, one can scale down the values of the basis functions. Instead of using x, one uses $z = \frac{x-X}{S_0}$ where X and S_0 are the strike price and initial asset value respectively. Then a variable z is much smaller and the fit will be more stable.

B-splines basis functions

B-splines are normally used for curve fitting purposes; for example, in approximating the term structure of interest rates (see James and Webber (2009) [137]).

A spline of order K is defined as a piecewise polynomial function that is $K-1$ times differentiable everywhere. A K-order spline is defined on an interval $[\delta_0, \delta_n]$, with knot points (where adjacent polynomials meet) $\delta_0 < \delta_1 < \ldots < \delta_n$.

In this chapter, the cubic B-spline is used. There are only four values of cubic B-splines that are non-zero for every value of x. For cubic splines, set $K = 3$ spline and define additional six knot points, $\delta_{-3} < \delta_{-2} < \delta_{-1} < \delta_0$ and $\delta_n < \delta_{n+1} < \delta_{n+2} < \delta_{n+3}$. Then write

$$\psi_i^B(x) = \sum_{k=i}^{i+4} \left(\prod_{\substack{j=i \\ j \neq k}}^{i+4} \frac{1}{\delta_j - \delta_k} \right) \left((x - \delta_k)^+\right)^3, \tag{5.3.6}$$

for $i = -3, \ldots, n-1$. $\psi_i^B(x)$ is non-zero on the interval $[\delta_i, \delta_{i+4}]$. For instance, if $x \in (\delta_k, \delta_{k+1})$, $\psi_i^B(x)$ is non-zero for $i = k-3, k-2, k-1$ and k. Note that since $\psi_i^B(s)$ is non-zero on the interval $[\delta_i, \delta_{i+4}]$, one needs at least four B-spline basis functions. Figure 5.2 shows a plot of a B-spline defined by knot points $\{0, 2, 4, 8, 16\}$. One can see that the B-spline is non-zero only in a range $[\delta_i, \delta_{i+4}] = [0, 16]$.

ψ_i^B estimates coefficients, $a_i, i = -3, \ldots, n-1$. For $x \in [\delta_0, \delta_n]$, one has

$$f(x) = \sum_{i=-3}^{n-1} a_i \psi_i^B(x). \tag{5.3.7}$$

5.3.2 Second phase: Computing the option value

The first phase algorithm gives a set $\left\{\widehat{S}^E\right\}_{i=1,\ldots,N-1}$, which is the estimated option's EEB. The second phase uses the estimated early exercise boundary to value the option with the Bermudan (or a combination of Bermudan) pricing control variate as described in section 5.2.2.

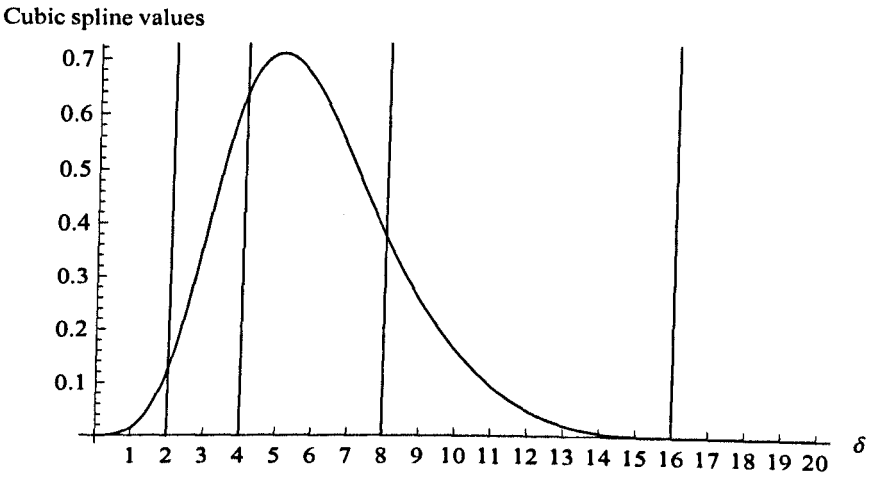


Figure 5.2. A B-spline defined by a set of knot points $\{0, 2, 4, 8, 16\}$

The second phase simulates another set of independent sample paths $\{S_j\}_{j=1,\ldots,M_2}$, where M_2 is not restricted to be equal to M_1. On the jth path, the option value with the pricing control variate is

$$\widehat{v}_j^{cv} = \widehat{v}_j - \omega c_j, \tag{5.3.8}$$

where ω is given by (4.3.33) and c_j is the value the pricing control variate on the jth path. and the option value is computed as the average

$$\frac{1}{M_2}\sum_{j=1}^{M_2}\widehat{v}_j^{cv} = \frac{1}{M_2}\sum_{j=1}^{M_2}\widehat{v}_j - \omega\left(\frac{1}{M_2}\sum_{j=1}^{M_2} c_j\right). \tag{5.3.9}$$

As discussed in section 4.3.6, one can use more than one control variate. Write $c^p_{\tau,j}$ for the value of c^p_τ on the jth path. Suppose one combines the put-tau, $c^p_{\tau,j}$, with the Bermudan put-tau control variate, $c^{b2}_{\tau,j}$, (5.3.8) becomes

$$\widehat{v}_j^{cv} = \widehat{v}_j - \omega^p c^p_{\tau,j} - \omega^{b2} c^{b2}_{\tau,j}, \tag{5.3.10}$$

where ω^p and ω^{b2} are elements of $\omega \in \mathbb{R}^2$ which is given by (4.3.39). The option value is found by

$$\frac{1}{M_2}\sum_{j=1}^{M_2}\widehat{v}_j^{cv} = \frac{1}{M_2}\sum_{j=1}^{M_2}\widehat{v}_j - \omega^p\left(\frac{1}{M_2}\sum_{j=1}^{M_2} c^p_{\tau,j}\right) - \omega^{b2}\left(\frac{1}{M_2}\sum_{j=1}^{M_2} c^{b2}_{\tau,j}\right). \tag{5.3.11}$$

5.4 Approximate American put options using Richardson extrapolation

This section illustrates the use of Monte Carlo values for Bermudan put options to estimate values of an American put option by using Richardson extrapolation, as suggested by Geske and Johnson (1984) [104]. The use of Richardson extrapolation in American option pricing is done in the context of lattice methods (Broadie and Detemple (1996) [48]). The use of extrapolation to estimate American option values from simulation techniques was suggested by Roger (2002) [216].

A method that can be viewed as a more sophisticated version of an extrapolation method is the multilevel Monte Carlo method of Giles (2008) [106]. The method generalises the idea of Kebaier (1990) [152], which used two levels of time step, to multiple levels using time steps in a geometric series. The main idea of a method is to achieve a desired accuracy with a smaller number of time steps, reducing the simulation variance by using a calculation with larger time steps. Results in Giles (2008) [106] suggest that the method can significantly reduce computational costs.

Next, the standard two-point Richardson extrapolation is discussed first and then the three-point scheme suggested by Geske and Johnson (1984) [104] is explored.

5.4.1 Two-point scheme

Write y for a time interval. Suppose a function $v(y)$ is known and satisfies

$$v(y) = v(0) + \alpha y^{k_1} + O\left(y^{k_2}\right), \; k_2 > k_1, \tag{5.4.1}$$

where k_1 is assumed to be known, $v(0)$ is the quantity one wants to compute, and $O\left(y^{k_2}\right)$ denotes a sum of terms of order k_2 and higher. This is one equation with two unknowns; $v(0)$ and α.

To solve for these unknowns, write another equation with a different step size qy:

$$v(qy) = v(0) + \alpha (qy)^{k_1} + O\left(y^{k_2}\right), \tag{5.4.2}$$

where $q > 1$. Substitute α and solving for $v(0)$ yields

$$v(0) = v(y) + \frac{v(y) - v(qy)}{q^{k_1} - 1} + O\left(y^{k_2}\right). \tag{5.4.3}$$

Using values of $v(y)$ and $v(qy)$, one can construct an estimate for $v(0)$.

5.4.2 Three-point scheme

Geske and Johnson (1984) [104] expressed the value of American options as an infinite series of multivariate normal distributions and used Richardson extrapolation to estimate the value of American options. This method can also be used to approximate American put values by extrapolating Bermudan put values from Monte Carlo simulation.

Write v^N for the value of Bermudan put with N exercise dates. The objective is to compute the value of the American put, v^∞. Let $y < 1$ be the time interval between each exercisable date, so that $y = \frac{T}{N}$ where T is the option's time to maturity. Write $v(y)$ for v^N so that $v^\infty \equiv v(0)$. Suppose

$$v(y) = v(0) + \alpha_1 y^{k_1} + \alpha_2 y^{k_2} + O\left(y^{k_3}\right) \tag{5.4.4}$$

for some constants α_1 and α_2, where $k_1 < k_2 < k_3$. Then for $q_1 < q_2$, one has

$$v(q_1 y) = v(0) + \alpha_1 (q_1 y)^{k_1} + \alpha_2 (q_1 y)^{k_2} + O\left(y^{k_3}\right), \tag{5.4.5}$$

$$v(q_2 y) = v(0) + \alpha_1 (q_2 y)^{k_1} + \alpha_2 (q_2 y)^{k_2} + O\left(y^{k_3}\right). \tag{5.4.6}$$

By substituting α_1 and α_2 and solving for $v(0)$, one obtains

$$v(0) = v(y) + \frac{C_1}{C_3}[v(y) - v(q_1 y)] - \frac{C_2}{C_3}[v(q_1 y) - v(q_2 y)], \tag{5.4.7}$$

where

$$C_1 = q_2^{k_2} - q_2^{k_1} + q_1^{k_1} - q_1^{k_2}, \tag{5.4.8}$$

$$C_2 = q_1^{k_2} - q_1^{k_1}, \tag{5.4.9}$$

$$C_3 = q_2^{k_2}\left(q_1^{k_1} - 1\right) - q_2^{k_1}\left(q_1^{k_2} - 1\right) + q_1^{k_2} - q_1^{k_1}. \tag{5.4.10}$$

To illustrate this, assume $N = 64$ and $T = 1$. Write $v^{1/y} = v(y)$ for a Bermudan put option with $1/y$ exercise dates. In this case, $v^{64} = v(y)$. The next step is to determine step sizes $q_1 y$ and $q_2 y$. One can set these step sizes to be $\frac{N}{2}$ and $\frac{N}{4}$ respectively. That is, $q_1 = 2$ and $q_2 = 4$. This corresponds to using a Bermudan put with v^{32}, and v^{16}, exercise dates respectively.

To obtain k_1 and k_2, Geske and Johnson (1984) [104] used the Taylor series expansion on $v(0)$ and considering only terms up to second order to obtain $k_1 = 1$ and $k_2 = 2$.[6]

[6]The numerical experiment also shows that the rate of convergence to American option values

Hence, one has the path-wise three-point approximation of American put value

$$v(0) \approx v^{64} + \frac{5}{3}\left(v^{64} - v^{32}\right) - \frac{1}{3}\left(v^{32} - v^{16}\right). \tag{5.4.11}$$

Write $\widehat{v}^N$ for a Monte Carlo value of the Bermudan put v^N. The three Bermudan values $\widehat{v}^{64}, \widehat{v}^{32}$ and $\widehat{v}^{16}$ can be computed simultaneously. This can be done with the slice-based simulation where option values are stored in an array at each time step t_i. Write $\widehat{v}_j^{N,i}$ for the simulated Bermudan option value with N exercise dates at the ith time step on the jth path. The slice-based simulation can be viewed as

$$\begin{pmatrix} \widehat{v}_1^{N,0} \\ \widehat{v}_2^{N,0} \\ \vdots \\ \widehat{v}_M^{N,0} \end{pmatrix} \leftarrow \begin{pmatrix} \widehat{v}_1^{N,1} \\ \widehat{v}_2^{N,1} \\ \vdots \\ \widehat{v}_M^{N,1} \end{pmatrix} \leftarrow \ldots \leftarrow \begin{pmatrix} \widehat{v}_1^{N,N} \\ \widehat{v}_2^{N,N} \\ \vdots \\ \widehat{v}_M^{N,N} \end{pmatrix} \tag{5.4.12}$$

At each step i, option values $\widehat{v}_j^{N,i}, j = 1\ldots, M$ is stored in an array. To compute values of $\widehat{v}_j^{N/2}$ and $\widehat{v}_j^{N/4}$ on the jth path, one only needs to create and store another two arrays and iterate back according to their exercise dates. In particular, write $x \mod y$ as a remainder from $\frac{x}{y}$. For $i = N, \ldots, 1$, if $i \mod q_1 \neq 0$, only $\widehat{v}_j^N$ is rolled back. If $i \mod q_1 = 0$ but $i \mod q_2 \neq 0$, both $\widehat{v}_j^N$ and $\widehat{v}_j^{N/2}$ are rolled back. If $i \mod q_1 = 0$ and $i \mod q_2 = 0$, all $\widehat{v}_j^N$, $\widehat{v}_j^{N/2}$, and $\widehat{v}_j^{N/4}$ are rolled back, *et cetera*. Since it is feasible to compute three Bermudan put values along each path, the estimation in (5.4.11) can be done easily.

5.4.3 Implementing Richardson extrapolation with Monte Carlo and control variates

Write $\widehat{v}_j^{64}, \widehat{v}_j^{32}$ and $\widehat{v}_j^{16}$ as the jth path Bermudan put option values with $64, 32$, and 16 exercise dates. One can implement (5.4.11) with control variates. Three possibilities are considered. They are described now.

5.4.3.1 Pricing only control variate

First write

$$\widehat{v}_j^{\mathrm{cv},64} = \widehat{v}_j^{64} - \omega^x \mathrm{c}_j^x \tag{5.4.13}$$

$$\widehat{v}_j^{\mathrm{cv},32} = \widehat{v}_j^{32} - \omega^y \mathrm{c}_j^y \tag{5.4.14}$$

$$\widehat{v}_j^{\mathrm{cv},16} = \widehat{v}_j^{16} - \omega^z \mathrm{c}_j^z, \tag{5.4.15}$$

from lattice method is uniform at rate ~ 1.

where c^x, c^y, and c^z are control variates. The value of the American put is

$$\widehat{v}_j(0) = \widehat{v}_j^{\text{cv},64} + \frac{5}{3}\left(\widehat{v}_j^{\text{cv},64} - \widehat{v}_j^{\text{cv},32}\right) - \frac{1}{3}\left(\widehat{v}_j^{\text{cv},32} - \widehat{v}_j^{\text{cv},16}\right). \tag{5.4.16}$$

This is a standard pricing control variate in which control variates are applied to each Bermudan put option value, $\widehat{v}^{64}$, $\widehat{v}^{32}$, and $\widehat{v}^{16}$. This is not computationally expensive since Bermudan option values with different exercise dates can be computed using the same sample paths.

5.4.3.2 $\widehat{v}(0)$ only control variate

This type of control variate does not apply to individual Bermudan put option values ($\widehat{v}^{64}$, $\widehat{v}^{32}$, and $\widehat{v}^{16}$). Instead it uses $\widehat{v}^l, l = 64, 32$ and 16 as control variates for the estimated American put option value $\widehat{v}(0)$. Define $\widehat{v}(0)$ only control variate, $c_j^{v0,l}$, on the jth path to be

$$c_j^{v0,l} = \widehat{v}_j^l - \frac{1}{M}\sum_{j=1}^{M} \widehat{v}_j^l, \tag{5.4.17}$$

for $l = 16, 32$ and 64. The value of extrapolated American put is

$$\widehat{v}_j(0) = \widehat{v}_j^{64} + \frac{5}{3}\left(\widehat{v}_j^{64} - \widehat{v}_j^{32}\right) - \frac{1}{3}\left(\widehat{v}_j^{32} - \widehat{v}_j^{16}\right) - \sum_{l=16,32,64} \omega^l c_j^{v0,l} \tag{5.4.18}$$

Note that the sum does not have to contain all three terms. In fact, one should not include too many terms since this will introduce multicollinearity among the controls.

5.4.3.3 Both pricing and $\widehat{v}(0)$ control

Combining (5.4.16) and (5.4.18) one has

$$\widehat{v}_j(0) = \widehat{v}_j^{\text{cv},64} + \frac{5}{3}\left(\widehat{v}_j^{\text{cv},64} - \widehat{v}_j^{\text{cv},32}\right) - \frac{1}{3}\left(\widehat{v}_j^{\text{cv},32} - \widehat{v}_j^{\text{cv},16}\right) - \sum_{l=16,32,64} \omega^l c_j^{v0,l} \tag{5.4.19}$$

This is called both pricing and $\widehat{v}(0)$ control, where one controls both individual Bermudan put options and an extrapolated value of American put options.

5.5 Numerical results

This section applies the Bermudan put-tau control variate method to value single-asset Bermudan put options with 64 exercise opportunities. The option pays $(X - S_\tau)^+$,

where X is a strike price and S_τ is a stock price at time τ, if exercised at time τ. The option's volatility is σ and the risk-free interest rate is r. Option expires at time T. Parameter values are $S_0 = 100$, $r = 0.05$, $\sigma = 0.2$ and $T = 1$.

5.5.1 First phase results

In this section, the proposed Bermudan put-tau option control variate is implemented to Bermudan put option's continuation values (rollback control variate). The Bermudan put option has $N = 64$ exercise dates. As discussed in section 5.3.1, it is desirable to combine the Bermudan put-tau control variate with the put-tau control variate. A combination of the two control variates, c_τ^{p+b2}, is implemented and results are compared with the put-tau control variate, c_τ^p.

To measure the accuracy of the estimated EEB, RMSE between it and a benchmark EEB, found by a lattice method, is computed. Write $\widehat{S}_i^E$ for the Monte Carlo EEB at time t_i and S_i^E for the EEB computed from a lattice with $N = 1,228,800$ time steps. The RMSE is computed as

$$\text{RMSE} = \sqrt{\sum_{i=1}^{N} \frac{\left(\widehat{S}_i^E - S_i^E\right)^2}{N}}, \tag{5.5.1}$$

where N is the number of exercise dates.

M		12,500		50,000		250,000		1,000,000	
CVs		c_τ^p	c_τ^{p+b2}	c_τ^p	c_τ^{p+b2}	c_τ^p	c_τ^{p+b2}	c_τ^p	c_τ^{p+b2}
RMSEs	1-64	0.203	**0.187**	0.110	**0.097**	0.078	**0.077**	0.084	**0.080**
$X = 90$	1-32	0.262	**0.244**	0.134	**0.095**	0.072	0.072	0.072	0.077
	1-16	0.339	**0.292**	0.174	**0.090**	0.065	0.069	0.049	0.055
	1-8	0.351	0.368	0.195	**0.101**	0.076	0.080	0.062	0.056
RMSEs	1-64	0.164	**0.127**	0.107	**0.127**	0.110	0.112	0.120	0.124
$X = 100$	1-32	0.186	**0.124**	0.116	**0.112**	0.105	**0.085**	0.086	0.101
	1-16	0.201	**0.087**	0.108	**0.075**	0.083	**0.069**	0.081	0.096
	1-8	0.220	**0.085**	0.091	**0.059**	0.064	**0.056**	0.054	**0.027**
RMSEs	1-64	0.208	**0.182**	0.147	0.149	0.165	**0.153**	0.159	**0.155**
$X = 110$	1-32	0.237	**0.133**	0.110	0.116	0.153	**0.125**	0.143	**0.127**
	1-16	0.173	**0.161**	0.139	**0.136**	0.203	**0.123**	0.179	**0.178**
	1-8	0.141	**0.055**	0.082	**0.060**	0.173	**0.054**	0.112	0.200

Table 5.3. RMSEs comparison between the put-tau, c_τ^p, and a combination of the put-tau and the Bermudan put-tau, c_τ^{p+b2}, rollback method

Table 5.3 reports RSMEs of the Bermudan option's early exercise boundary with different strike prices, X, and sample paths, M. The method uses 12 B-spline basis functions. The control variate methods are the put-tau, c_τ^p, rollback and the Bermudan put-tau rollback combined with the put-tau method, c_τ^{p+b2}. The S_0-dispersion method is also implemented to reduce noise of an early segment of the EEB (Rasmussen (2005) [203]). Table 5.3 shows four sets of RMSEs. Each set measures RMSEs from different time steps. The first set is computed from all 64 exercise dates. The second set is computed from $1-32$ exercise dates, the third is from $1-16$ exercise dates and the fourth is from $1-8$ exercise dates only.

In table 5.3, cases where c_τ^{p+b2} rollback control variate yields smaller RMSEs than those computed from c_τ^p rollback control variate are highlighted. One can see that RMSEs from c_τ^{p+b2} control variate are mostly less than those from c_τ^p control variate. This seems to be consistent across all sets of the EEB. The smaller RMSEs implies that the c_τ^{p+b2} approximates the option's EEB more accurately than the c_τ^p does. Also, RMSEs decrease as sample paths M increases. Hence, M must not be too small, otherwise the EEB will not be estimated accurately.

5.5.2 Second phase results

In this section, Bermudan put-tau control variates are applied to the Bermudan put option value. To measure efficiency gains over the plain LSLS method, a variance reduction factor, **e**, in equation (3.3.2) is computed.

Section 5.5.2.1 presents Bermudan put values computed from Monte Carlo with the EEB computed from a lattice method. Then Bermudan put values from the two-phase method will be presented in section 5.5.2.2.

5.5.2.1 Control variate effectiveness with the lattice EEB

In this section, Bermudan put option values are computed based on the EEB estimated from a lattice method. The purpose is to assess the effectiveness of control variate methods so that one is certain that option values are not affected by noises from the estimated EEB. In this case, one can view Bermudan puts as barrier options whose barrier is the lattice EEB.

The control variates are Bermudan put-tau and its variants that are proposed and described in section 5.2.1 and 5.2.2. In addition, the put-tau control variate, c_τ^p, and the stock-tau control variate, c_τ^s, are used. Let τ be the exercise time of the

Bermudan put being valued. The stock-tau control variate is defined as

$$c_\tau^s = \begin{cases} e^{-r\tau} S_\tau - S_0, & \text{if } \tau < T, \\ e^{-rT} S_T - S_0, & \text{Otherwise.} \end{cases} \tag{5.5.2}$$

All control variates used are summarised in table 5.4. The benchmark EEB for a

Control variates		Equations
c_τ^p	put-tau	(4.3.44)
c_τ^s	stock-tau	(5.5.2)
c_T^{b2}	Bermudan-terminal	(5.2.12)
c_τ^{b2}	Bermudan put-tau	(5.2.13)
$c_{\tau,T_1}^{b2,\mathrm{pl}}$	plain Bermudan T_1	(5.2.14)
c_{τ,T_1}^{b2}	Bermudan T_1	(5.2.15)
$c_\tau^{b2,\mathrm{e}}$	early Bermudan	(5.2.18)

Table 5.4. Summary of control variates used in table 5.5

Bermudan put with 64 exercise dates, S^E, is computed by a lattice method with $1,228,800$ time steps. A number of sample paths $M = 10^6$. Results are reported in table 5.5. It reports Bermudan put option values v^{64}, standard errors, computational time in seconds, and efficiency gains. Standard errors are reported in round brackets. Computational times are reported in square brackets and gains are shown in bold.

Panel (a) of table 5.5 reports results from using variants of Bermudan put-tau control variates described in section 5.2.1 and 5.2.2. It shows the correlation ρ between the single Bermudan control variate and the Bermudan put value.

Panel (b) of table 5.5 reports results from using these control variates combined with the put-tau control variate, except the second column where the single put-tau control variate is shown. It also shows the correlation between these control variates with the put-tau control variate. Panel (b) also reports values of R^2 from equation (4.3.40) computed from 64-exercise dates Bermudan put values and a combination of control variates.

Amongst single control variate, the European put-tau control variate achieves the highest variance reductions with the highest correlation of 0.999 (comparing 0.999 with values ρ with v^{64} from panel (a)) and an efficiency gain of 505. The second best is the Bermudan put-tau control variate with a slightly smaller correlation of 0.98 and much smaller variance reductions of 21. This is because the Bermudan option control variate can only be exercised at time $T_2 = T$ and $T_1 = T/2$. Hence, when $\tau = T_1$, the option holder will receive only a payoff $(X - S_{T_1})^+$ instead of

CV	Plain	c_τ^s	c_τ^{b2}	c_T^{b2}	c_{τ,T_1}^{b2}	$c_{\tau,T_1}^{b2,pl}$	$c_\tau^{b2,e}$
ρ with v^{64}		-0.77	0.98	0.84	0.82	0.89	0.45
v^{64}	6.0835	6.0872	6.0819	6.0822	6.0795	6.0801	6.0783
(se)	(0.0071)	(0.0046)	(0.0014)	(0.0039)	(0.0041)	(0.0032)	(0.0063)
$[t]$	[67]	[71]	[87]	[69]	[88]	[86]	[88]
e	**1.0**	**2.3**	**21**	**3.2**	**2.3**	**3.8**	**0.96**

Panel (a): Single control variate

CV	c_τ^p	$c_\tau^p + c_\tau^s$	$c_\tau^p + c_\tau^{b2}$	$c_\tau^p + c_T^{b2}$	$c_\tau^p + c_{\tau,T_1}^{b2}$	$c_\tau^p + c_{\tau,T_1}^{b2,pl}$	$c_\tau^p + c_\tau^{b2,e}$
R^2	-	0.5977	0.9983	0.6883	0.9990	0.7954	0.9992
ρ with c_τ^p	0.999	-0.77	0.98	0.83	0.80	0.89	0.49
v^{64}	6.08114	6.08132	6.08107	6.08136	6.08137	6.08165	6.08121
(se)	(0.00030)	(0.00029)	(0.00029)	(0.00030)	(0.00023	(0.00030)	(0.00020)
$[t]$	[74]	[78]	[94]	[76]	[95]	[93]	[95]
e	**505**	**506**	**421**	**494**	**694**	**406**	**872**

Panel (b): Combinations control variates with c_τ^p (except the second column)

Table 5.5. Bermudan put values and variance reduction gains from various kinds of Bermudan control variates by using Lattice boundary

an European put option (as in the put-tau case) which is more correlated to the option being valued. Except for the early Bermudan control variate that produces an efficient gain of less than 1, other Bermudan-based control variates achieve marginal efficient gains of ~2-3.

When combining the put-tau control variate with the Bermudan-style control variate, additional efficient gains are substantially improved. In particular, amongst all combinations one can see that a combination of the early Bermudan control variate with the put-tau control variate yields the highest efficiency gain of 872 and the highest R^2 of 0.999. By comparing with the single put-tau, even though the computational time is increased by ~30% (from 74 to 95 seconds), the standard error is decreased by ~33% (from 0.0003 to 0.0002) and the efficiency gain is increased by ~72% (from 505 to 872). The use of the Bermudan T_1 control variate combined with the put-tau control also achieves greater efficiency gains of 694.

Even though other combinations may achieve a high R^2, their efficiency gains reduce (relative to efficiency gains from the single put-tau of 505). This is because the correlation between the two controls are high. In other words, adding one more control will not contribute much in these cases. Hence, one would expect a combination with the highest R^2 and the lowest correlation between the two controls

to be most effective. This is confirmed by the highest gain of 872 achieved by using the combination of the early Bermudan and the put-tau control variates. One can see that this combination yields the highest R^2 of 0.9992 with the lowest correlation of 0.49. By comparing this combination with the combination between the Bermudan T_1 and the put-tau control variate, $c_{T_1}^{b2} + c_\tau^p$, one can see that even though the $c_{T_1}^{b2} + c_\tau^p$ combination yields a high R^2 of 0.9990, a gain is 694 which is ~20% less that a gain from $c_\tau^{b2,e} + c_\tau^p$. This is because a correlation between the Bermudan T_1 and the put-tau control variates of 0.80 is much higher than that between the early Bermudan and the put-tau control variates of 0.49. The correlation between the early Bermudan and the put-tau control variates is the lowest among other combinations because the cash flow at time T_1 has been removed unlike other types of Bermudan put control variates.

5.5.2.2 Full two-phase results

A full two-phase method is now implemented to value Bermudan put option with 64 exercise dates. Numbers of sample paths for the first and the second phase, M_1 and M_2, are varied. The Bermudan put-tau is used with the put-tau as the rollback control variate. For the pricing control variate, the put-tau control variate is combined with the early Bermudan control variate. Results are compared with the single put-tau control variate. Results include option values, standard errors, computational time in seconds, biases computed by equation (3.3.1) and efficiency gains computed by equation (3.3.2). Standard errors are reported in round brackets. Computational times are reported in square brackets. Biases are reported in curly brackets and efficiency gains are reported in bold. The efficiency gains are computed relatively to the plain LSLS method with a total number of sample paths $M = M_1$.

Tables 5.6 to 5.8 report the results of the two-phase method with different strikes, X. All results are obtained by using 12 B-spline basis functions. The number of sample paths used to estimate the EEB, M_1, and the number of sample paths used to value options, M_2, are varied from 2500 to 10^6. The S_0-dispersion method is also implemented in the first phase. The single put-tau and the combination of the put-tau and the early Bermudan control variates are implemented at the pricing stage. The total computational times are the sum of the time taken from the first phase, t_{M_1}, and from the second phase, t_{M_2}.

Tables 5.6 to 5.8 suggest that M_2 has an impact on the standard errors while M_1 has an impact on bias. When the value of M_1 is small, $M_1 = 2,500$ say, there are large biases in option values when $M_2 \geq 50,000$. This is because the option's EEB is computed with a small number of sample paths and thus is effected by simulation

$X = 95$										
M_1 [time]	M_2 / [time]									
	2500		12500		50000		250000		1000000	
	c_τ^p [0.26]	$c_\tau^p + c_\tau^{b2,e}$ [0.30]	c_τ^p [0.99]	$c_\tau^p + c_\tau^{b2,e}$ [1.2]	c_τ^p [3.7]	$c_\tau^p + c_\tau^{b2,e}$ [4.4]	c_τ^p [18]	$c_\tau^p + c_\tau^{b2,e}$ [21]	c_τ^p [72]	$c_\tau^p + c_\tau^{b2,e}$ [86]
2500	4.0019	4.0084	4.0054	4.0077	4.0055	4.00645	4.00559	4.00564	4.00584	4.00568
	(0.0045)	(0.0031)	(0.0027)	(0.0014)	(0.0010)	(0.00069)	(0.00046)	(0.00031)	(0.00023)	(0.00016)
	{-1.0}	{0.62}	{-0.39}	{0.88}	{-0.96}	{-0.02}	{-1.9}	{-2.6}	{-2.7}	{-5.0}
[13]	**147**	**322**	**664**	**1358**	**2140**	**4441**	**5429**	**10496**	**7959**	**14833**
12500	4.0024	4.0072	4.0085	4.0050	4.0070	4.00615	4.00645	4.00608	4.00619	4.00639
	(0.0045)	(0.0030)	(0.0021)	(0.0014)	(0.0010)	(0.00068)	(0.00046)	(0.00031)	(0.00023)	(0.00015)
	{-0.90}	{0.24}	{1.0}	{-1.1}	{0.54}	{-0.45}	{-0.03}	{-1.2}	{-1.2}	{-0.47}
[65]	**31**	**68**	**142**	**318**	**516**	**1150**	**2061**	**4333**	**5041**	**10005**
50000	4.0045	4.0035	4.0085	4.0070	4.0075	4.00624	4.00588	4.00686	4.00610	4.00617
	(0.0047)	(0.0032)	(0.0020)	(0.0014)	(0.0010)	(0.00069)	(0.00046)	(0.00031)	(0.00023)	(0.00015)
	{-0.4}	{-0.92}	{1.0}	{0.38}	{1.0}	{-0.32}	{-1.3}	{1.3}	{-1.6}	{-1.9}
[261]	**7**	**16**	**37**	**83**	**134**	**294**	**610**	**1323**	**2062**	**4352**
250000	4.0045	4.0059	4.0087	4.0069	4.0067	4.00611	4.00646	4.00672	4.00605	4.00669
	(0.0047)	(0.0031)	(0.0020)	(0.0014)	(0.0010)	(0.00069)	(0.00046)	(0.00031)	(0.00023)	(0.00015)
	{-0.42}	{-0.18}	{1.1}	{0.31}	{0.24}	{-0.51}	{0.01}	{0.83}	{-1.8}	{1.5}
[1301]	**1**	**3**	**8**	**16**	**28**	**60**	**130**	**286**	**498**	**1089**
1000000	4.0081	4.0060	4.0063	4.0074	4.0062	4.00731	4.00655	4.00702	4.00609	4.00647
	(0.0046)	(0.0031)	(0.0020)	(0.0014)	(0.0010)	(0.00069)	(0.00046)	(0.00031)	(0.00023)	(0.00016)
	{0.35}	{-0.14}	{-0.08}	{0.67}	{-0.30}	{1.2}	{0.19}	{1.8}	{-1.6}	{0.05}
[5210]	**2**	**4**	**9**	**20**	**35**	**77**	**162**	**360**	**629**	**1356**

Table 5.6. Two-phase method to value Bermudan put options. Put+Bermudan tau roll back with S_0-dispersion. Pricing control variates are put-tau and put-tau + Bermudan early. Benchmark price is 4.00646.

$X = 100$										
M_1 [time]	M_2 / [time]									
	2500		12500		50000		250000		1000000	
	c_τ^p [0.27]	$c_\tau^p + c_\tau^{b2,e}$ [0.33]	c_τ^p [1.02]	$c_\tau^p + c_\tau^{b2,e}$ [1.3]	c_τ^p [3.8]	$c_\tau^p + c_\tau^{b2,e}$ [4.9]	c_τ^p [19]	$c_\tau^p + c_\tau^{b2,e}$ [24]	c_τ^p [74]	$c_\tau^p + c_\tau^{b2,e}$ [96]
2500	6.0886	6.0787	6.0830	6.08114	6.0808	6.08115	6.07900	6.08048	6.07967	6.07999
	(0.0061)	(0.0040)	(0.0028)	(0.0019)	(0.0014)	(0.00092)	(0.00061)	(0.00041)	(0.00031)	(0.00021)
	{1.2}	{-0.62}	{0.65}	{-0.03}	{-0.27}	{-0.03}	{-3.6}	{-1.7}	{-4.9}	{-5.8}
[14]	**157**	**363**	**687**	**1494**	**2249**	**4682**	**6417**	**12201**	**9600**	**17066**
12500	6.0843	6.0806	6.0768	6.0844	6.0801	6.08264	6.08192	6.08095	6.08115	6.08113
	(0.0061)	(0.0040)	(0.0030)	(0.0018)	(0.0014)	(0.00091)	(0.00060)	(0.00041)	(0.00030)	(0.00020)
	{0.51}	{-0.15}	{-1.6}	{1.8}	{-0.77}	{1.6}	{1.2}	{-0.56}	{-0.10}	{-0.25}
[71]	**32**	**75**	**155**	**342**	**562**	**1227**	**2418**	**5070**	**5994**	**11641**
50000	6.0706	6.0859	6.0787	6.0787	6.0802	6.08095	6.08018	6.08107	6.08079	6.08088
	(0.0063)	(0.0040)	(0.0027)	(0.0018)	(0.0014)	(0.00091)	(0.00061)	(0.00040)	(0.00030)	(0.00020)
	{-1.7}	{1.2}	{-0.91}	{-1.3}	{-0.70}	{-0.25}	{-1.7}	{-0.28}	{-1.9}	{-1.5}
[286]	**7**	**19**	**39**	**83**	**145**	**321**	**708**	**1563**	**2411**	**5093**
250000	6.0879	6.0802	6.0817	6.0837	6.0799	6.08097	6.08085	6.08147	6.08066	6.08093
	(0.0060)	(0.0040)	(0.0027)	(0.0018)	(0.0014)	(0.00090)	(0.00061)	(0.00041)	(0.00030)	(0.00020)
	{1.1}	{-0.25}	{0.19}	{1.4}	{-1.0}	{-0.23}	{-0.54}	{0.71}	{-1.7}	{-1.3}
[1436]	**2**	**4**	**8**	**17**	**29**	**66**	**148**	**327**	**577**	**1266**
1000000	6.0853	6.0828	6.0817	6.0812	6.0798	6.08024	6.08113	6.08171	6.08171	6.08087
	(0.0062)	(0.0041)	(0.0027)	(0.0018)	(0.0014)	(0.00091)	(0.00061)	(0.00041)	(0.00030)	(0.00020)
	{0.66}	{0.39}	{0.19}	{0.01}	{-1.0}	{-1.0}	{-0.08}	{1.3}	{1.8}	{-1.5}
[5742]	**0.4**	**1**	**2**	**4**	**7**	**16**	**37**	**84**	**149**	**331**

Table 5.7. Two-phase method to value Bermudan put options. $X = 100$. Benchmark value is 6.08118.

$X = 105$										
M_1 [time]	M_2 / [time]									
	2500		12500		50000		250000		1000000	
	c_τ^p [0.30]	$c_\tau^p + c_\tau^{b2,e}$ [0.36]	c_τ^p [1.08]	$c_\tau^p + c_\tau^{b2,e}$ [1.5]	c_τ^p [4.0]	$c_\tau^p + c_\tau^{b2,e}$ [5.6]	c_τ^p [20]	$c_\tau^p + c_\tau^{b2,e}$ [28]	c_τ^p [79]	$c_\tau^p + c_\tau^{b2,e}$ [111]
2500	8.7322	8.7255	8.7301	8.7258	8.7243	8.7298	8.72935	8.72780	8.72807	8.72762
	(0.0075)	(0.0051)	(0.0034)	(0.0023)	(0.0017)	(0.0012)	(0.00075)	(0.00052)	(0.00038)	(0.00026)
	{0.54}	{-0.51}	{0.58}	{-1.0}	{-2.2}	{1.5}	{1.6}	{-0.59}	{-0.10}	{-1.9}
[16]	**168**	**364**	**704**	**1432**	**2522**	**5022**	**6763**	**11697**	**10143**	**16300**
12500	8.7420	8.7259	8.7299	8.7263	8.7271	8.7267	8.72583	8.72750	8.72668	8.72669
	(0.0076)	(0.0050)	(0.0034)	(0.0023)	(0.0017)	(0.0012)	(0.00076)	(0.00052)	(0.00038)	(0.00026)
	{1.8}	{-0.44}	{0.52}	{-0.78}	{-0.60}	{-1.2}	{-3.0}	{-1.2}	{-3.8}	{-5.5}
[79]	**33**	**75**	**147**	**304**	**594**	**1244**	**2394**	**4732**	**6016**	**10654**
50000	8.7294	8.7271	8.7279	8.7271	8.7274	8.7265	8.72711	8.72765	8.72826	8.72777
	(0.0076)	(0.0053)	(0.0034)	(0.0023)	(0.0017)	(0.0012)	(0.00076)	(0.00052)	(0.00038)	(0.00026)
	{0.17}	{-0.19}	{-0.06}	{-0.44}	{-0.41}	{-1.4}	{-1.3}	{-0.88}	{0.39}	{-1.3}
[316]	**8**	**17**	**38**	**79**	**156**	**334**	**709**	**1479**	**2419**	**4769**
250000	8.7329	8.7192	8.7282	8.7305	8.7267	8.7297	8.72721	8.72799	8.72782	8.72784
	(0.0075)	(0.0053)	(0.0034)	(0.0023)	(0.0017)	(0.0018)	(0.00076)	(0.00052)	(0.00038)	(0.00026)
	{0.63}	{-1.7}	{0.02}	{1.0}	{-0.83}	{1.4}	{-1.2}	{-0.2}	{-0.8}	{-1.0}
[1579]	**2**	**3**	**7**	**15**	**31**	**66**	**149**	**316**	**575**	**1208**
1000000	8.7304	8.7244	8.7240	8.7310	8.7264	8.7277	8.72691	8.72772	8.72797	8.72798
	(0.0077)	(0.0052)	(0.0034)	(0.0023)	(0.0017)	(0.0012)	(0.00076)	(0.00052)	(0.00038)	(0.00026)
	{0.29}	{-0.71}	{-1.2}	{1.2}	{-1.0}	{-0.32}	{-1.6}	{-0.75}	{-0.36}	{-0.50}
[6301]	**0.4**	**1**	**2**	**4**	**8**	**17**	**37**	**80**	**149**	**316**

Table 5.8. Two-phase method to value Bermudan put options. Benchmark value is 8.72811

noise. Note that when $M_2 < 50,000$, the bias cannot be seen because of the higher standard errors in option values.

When comparing the performance between the single put-tau control variates and the combination of the put-tau and the early Bermudan control variates, one sees that the combination of put-tau and Bermudan early pricing control variate yields much greater variance reductions across all values of M_1, M_2, and X. More specifically, gains from the combination of control variates are twice the size of those from the single put-tau control variate. This shows that combination of the Bermudan early with the put-tau control variate is more effective than the single put-tau control variate.

The variance reductions obtained from the combination of the put-tau and the early Bermudan control variates are very substantial. For instance, consider a pair $(M_1, M_2) = (12,500,250,000)$ in table 5.6. It takes approximately 86 seconds to compute option values with these M_1 and M_2 to achieve a gain of 4,000 over the plain LSLS method. To obtain the same level of standard errors with one-phase method with the same M_1, it will take ~1,300 seconds.

5.5.2.3 Extrapolated American put options

Tables 5.9 and 5.10 report extrapolated American put options using the method described in section 5.4 with different strikes, X. The simulation is the one phase method where $M_1 = M_2 = M$. A number of sample paths, M, is 50,000.

Table 5.9 shows extrapolated American put values using the combination of the early Bermudan and the put-tau pricing controls. This is the pricing only control variate in equation (5.4.16) in which control variates are applied only to individual Bermudan put values used for extrapolating. Table 5.10 adds the $v(0)$ control to the combination of control variates used in table 5.9. Individual Bermudan put values are computed using the put-tau combined with the Bermudan put-tau rollback method. This is given by (5.3.3).

Tables 5.9 and 5.10 show Bermudan put option values with different exercise dates; $N = 64, 32$ and 16. Benchmark values, v, are computed from a lattice method with $204,800$ time steps. Monte Carlo values, $\widehat{v}$, are computed with time steps corresponding to their exercise dates. Tables 5.9 and 5.10 also report option values with infinite exercise dates computed from both lattice method and Monte Carlo simulation. These values correspond to American put option values that are extrapolated from Bermudan put values with $64, 32$ and 16 exercise dates as described in section 5.4. Standard errors (se), computational times [t] and biases {b} (with respect to values from a lattice method) are reported.

N		∞	64	32	16
Lattice values		4.01309	4.00646	4.00000	3.98744
MC values	$\widehat{v}$	4.01175	4.00564	3.99949	3.98703
$X = 95$	(se)	(0.00087)	(0.00070)	(0.00071)	(0.00076)
$[t] = 116$	{b}	{-1.5}	{-1.2}	{-0.71}	{-0.53}
Lattice values		6.09037	6.08118	6.07214	6.05442
MC values	$\widehat{v}$	6.0915	6.08112	6.07149	6.05448
$X = 100$	(se)	(0.0011)	(0.00091)	(0.00094)	(0.00098)
$[t] = 165$	{b}	{1.0}	{-0.06}	{-0.70}	{0.06}
Lattice values		8.74017	8.72811	8.71616	8.69254
MC values	$\widehat{v}$	8.7392	8.7275	8.7158	8.6927
$X = 105$	(se)	(0.0014)	(0.0012)	(0.0012)	(0.0012)
$[t] = 212$	{b}	{-0.65}	{-0.56}	{-0.32}	{0.15}

Table 5.9. Extrapolated American put values with pricing control variate only

N		∞	64	32	16
Lattice values		4.01309	4.00646	4.00000	3.98744
MC values	$\widehat{v}$	4.01293	4.00647	4.00016	3.98801
$X = 95$	(se)	(0.00045)	(0.00069)	(0.00072)	(0.00076)
$[t] = 117$	{b}	{-0.34}	{0.01}	{0.22}	{0.75}
Lattice values		6.09037	6.08118	6.07214	6.05442
MC values	$\widehat{v}$	6.09118	6.08258	6.07391	6.05631
$X = 100$	(se)	(0.00062)	(0.00090)	(0.00093)	(0.00098)
$[t] = 168$	{b}	{1.3}	{1.6}	{1.9}	{1.9}
Lattice values		8.74017	8.72811	8.71616	8.69254
MC values	$\widehat{v}$	8.74028	8.7270	8.7144	8.6916
$X = 105$	(se)	(0.00077)	(0.0012)	(0.0012)	(0.0012)
$[t] = 216$	{b}	{0.15}	{-1.0}	{-1.5}	{-0.72}

Table 5.10. Extrapolated American put values with pricing and $v(0)$ control variate

Results in tables 5.9 and 5.10 show no evidence of bias in extrapolated American put values. Without the use of $v(0)$ as an additional control variate, extrapolated American puts show slightly higher standard errors than those from individual Bermudan puts.

The standard errors can be further reduced by combining the pricing control with the $v\,(0)$ control. Table 5.10 illustrates this. When adding $v\,(0)$ control variate, the standard errors of extrapolated American put values decrease. Table 5.11 shows that the efficiency gains from adding the $v(0)$ control is $\sim$3.5 over the method where only the pricing control variate is applied.

Lastly, note that the computational cost mainly comes from the fitting procedure which is $64 + 32 + 16 = 112$ times in this case. This is substantially cheaper compared with the plain simulation method where it takes $\sim$10,000 time steps for Bermudan put values to converge to American put values.

X	95	100	105
e	**3.7**	**3.3**	**3.4**

Table 5.11. Efficiency gains from adding $v(0)$ control

5.6 Conclusion

This chapter has described a novel control variate method to value Bermudan put options. The control variate is the Bermudan put option with two exercise dates. In general there exists no explicit solution to evaluate Bermudan puts. However, with two exercisable times, one can write an analytical solution for the Bermudan put as a compound call option. Different variants for the Bermudan control variate are described.

The method is implemented with a two-phase Monte Carlo simulation in which the option's early exercise boundary (EEB) is estimated in the first phase and then the second phase uses this EEB to value a Bermudan put option. The Bermudan control variate is first applied to the option's continuation values to improve the estimation of the early exercise boundary (EEB). The numerical results suggest that by implementing the Bermudan control variate together with the put-tau rollback control variate, errors are significantly smaller relative to those obtained from using the put-tau control variate by itself.

With an accurate EEB, the Bermudan control variate is applied to pricing Bermudan put options. Results in section 5.5 suggest that there is no bias in

the Bermudan put values. Also, when combining one particular Bermudan control variate, namely, the early Bermudan, with the put-tau control variate, the method achieves very substantial efficiency gains. All gains produced from this control variate combination are greater than those from the put-tau alone, by at least a factor of two.

To approximate the value of the American put option, a three-point Richardson extrapolation technique is implemented. The method extrapolates three Bermudan put options with different reset dates. The results show no evidence of bias in extrapolated American put values. The American put value can be estimated accurately with much lower computational time relative to valuing a Bermudan option with a large value of exercise dates. To further reduce standard errors of the extrapolated values, individual Bermudan put values are applied as control variates. These control variates can help the method to run $\sim$3 times faster.

Chapter 6

Valuing American Put Options using the Sequential Contour Monte Carlo Method

This chapter is concerned with the valuation of American put options by using Monte Carlo simulation. The chapter describes a novel simulation method, the sequential contour Monte Carlo (SCMC) method, based on hitting times to a set of predefined exponential contours, that can be used to value American put options accurately.

The method generalises the standard LSLS method in such a way that it allows one to use a broader family of contours (instead of only vertical contours in the LSLS method) to value American options. Instead of iterating backward on a fixed time step, one iterates backward on each contour.

Because the method is based on hitting times to a set of contours, the method can provide values of barrier options on each contour that can be used as control variates to the American put option.

The structure of the chapter is as follows. Section 6.1 describes the sequential contour Monte Carlo method. The contour construction is described in section 6.2. Section 6.3 introduces the control variate method obtained from the SCMC method. Numerical results are reported in section 6.4. Section 6.5 concludes.

6.1 The sequential contour Monte Carlo (SCMC) Method

Consider a fixed set of exponential contours, $\beta\left(t \mid \alpha^i\right)$, of a form

$$\beta\left(t \mid \alpha^i\right) = \frac{1}{\alpha^i} e^{gt}, \tag{6.1.1}$$

where g is a contour growth rate, $i = 1, \ldots, N-1$, and N is a total number of exercise opportunities for an option. Set $\beta^i(t) = \beta\left(t \mid \alpha^i\right)$. Suppose that $\beta^{i_1}(t) > \beta^{i_2}(t)$ for all t and all $i_1 < i_2$, so that the set of contours forms a decreasing sequence. Write S_{t_0} for the asset value that follows a geometric Brownian motion process in (1.0.1). Suppose that $S_{t_0} > \beta^{i_1}(t_0)$ so that no contour has yet been hit.

The main idea of the SCMC method is to generate a sequence of hitting times to an indexed set of contours and then to apply the LSLS algorithm to compute the value of the option. These values of the option are used to estimate the value of an American put option by Richardson extrapolation. The LSLS method is generalised and will be described in section 6.1.3. Unlike the contour bridge method described in chapter 3 where contours are generated while the algorithm proceeds, a set of contours in the SCMC method is fixed before the algorithm runs.

Single hit contour

It is important to require that the contour in (6.1.1) is single-hit to the option's EEB. In other words, a contour intersects the option's EEB at most once. This ensures that the option can be exercised only once on each contour. That is, the option with N exercise opportunities will require N contours. Since the option's EEB has no analytical form, its maximum slope cannot be obtained. Hence, the condition in section 3.2.6 cannot be applied in order to ensure that the contour is single hit. In this case, to have the single-hit contour, one must choose a steep contour.

Write $S^i = S_{t_i}$ for the asset value at time t_i. The hitting time to the ith contour, τ^i, is defined to be

$$\tau^i = \inf\{t \in (0, T] : S^i \leq \beta^i(t)\}, \tag{6.1.2}$$

where T is the option's time to maturity. The distribution of a hitting time in (6.1.2) is known and can be sampled from. The hitting time simulation method is described in section 3.2.2.

It is important to notice the difference between the hitting time, τ, in (6.1.2) and the exercise time, τ^*, of the option with early exercise feature, such as Bermudan options and American options (see equation (4.1.3) for the definition of τ^* for an American put option). On the jth sample path, a set $\tau_j = \left\{\tau_j^i\right\}_{i=1,\ldots,N}$ represents hitting times to the ith contour. However, the option can only be exercised optimally once at, τ_j^*, on the jth path. In the SCMC method, the optimal exercise time, τ^*,

for a put option becomes

$$\tau^* = \inf\{\tau \geq 0 : S_\tau \leq S_\tau^E\}, \tag{6.1.3}$$

where S_t^E is the EEB and τ is given by (6.1.2).

Before the SCMC method is described, it is important to define what options the method is valuing. In particular, the option that is exercised at time τ^* in (6.1.3). This is described in the next section.

6.1.1 Sequential contour (SC) options

A sequential contour (SC) option is an option whose holder can exercise at any time up to maturity T. However, the exercise decision can only be made when the asset value lies on one of a predetermined set of exponential contours. Specifically, the option can be exercised at time τ if and only if $S_\tau = \beta^i(\tau)$ for $i = 1, \ldots, N$ where $\beta^i(\tau)$ is given by (6.1.1).

To see the difference between a SC and a Bermudan option, let $\hat{\beta}^i$ be a set of points (x, y) in a two-dimensional plane on the ith contour such that

$$\text{SC option:} \qquad \hat{\beta}^i = \left\{(x,y) \mid y = \beta^i(x)\right\}, \tag{6.1.4}$$

$$\text{Bermudan option:} \qquad \hat{\beta}^i = \left\{(x,y) \mid x = t_i\right\}. \tag{6.1.5}$$

One can view (6.1.5) as simulating a hitting time to a vertical contour (where t_i is a fixed time). In particular, let $\Upsilon(t,T)$ be a set of exercise times of a Bermudan option and a SC option. It is defined as follows:

$$\text{SC option:} \qquad \Upsilon(t,T) = \left\{\tau^i \mid \tau^i \text{ is given by (6.1.2)}\right\}, \tag{6.1.6}$$

$$\text{Bermudan option:} \qquad \Upsilon(t,T) = \left\{\tau^i \mid \tau^i = t_i \text{ for } t \leq t_i \leq T\right\}. \tag{6.1.7}$$

τ^i for a standard Bermudan option is essentially a predetermined set of exercise dates $\{t_i\}_{i=1,\ldots,N}$.

Therefore, one can see that τ^i for a Bermudan option are non-random. On the other hand, τ^i for a SC option is a hitting time to the ith contour and is therefore random. In other words, one does not know in advance where τ^i will be on the ith contour.

Also, write $\Delta\tau^i = \tau^{i+1} - \tau^i$ for time intervals between successive exercise time of a SC option. Since τ^i for a SC option is random, $\Delta\tau^i$ is also random. In fact, $\Delta\tau^i$ will depend on how the contours are constructed. This will be discussed in section 6.2.

Figure 6.1 shows exercise opportunities for a Bermudan and a SC option. Panel (a) shows exercise opportunities for a Bermudan option. Exercise can be made in a predetermined set $\{t_i\}_{i=1,\ldots,N}$.[1] Panel (b) shows exercise opportunities for a SC option. In this case, exercise can be made in a random set of hitting times $\{\tau_i\}_{i=1,\ldots,N}$. Generally, $\Delta\tau^i$ are not equal for $i = 1, \ldots, N$.

Write $v_0^{b,N}$ and $v_0^{s,N}$ as time $t = 0$ values of a Bermudan option and a SC option with N exercise dates respectively. Clearly one has

$$v_0^{b,N} \neq v_0^{s,N}. \tag{6.1.8}$$

The value of a Bermudan put option with N exercise dates is generally not equal to the value of a SC option with the same N exercise dates. As $N \to \infty$, both values will converge to the American option value.

The objective of the SCMC method is to use values of a SC put option to approximate the value of an American put option. The overview of the SCMC method is given as follows.

1. Construct asset value sample paths by generating hitting times to a predetermined set of contours.

2. Apply the (generalised) LSLS algorithm with control variate methods to obtain values of SC options.

3. Implement Richardson extrapolation on these SC option values to approximate the value of an American option.

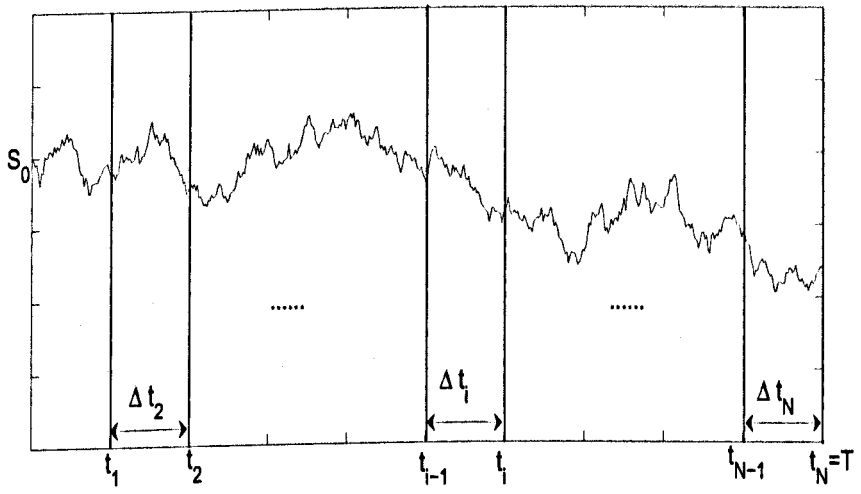
The advantage of the SCMC method is that simulated hitting times on each contour can be used to value barrier options and these options can be used as control variates for a SC option. Steps 1 to 3 will be described in detail next.

6.1.2 Sequential contour path construction

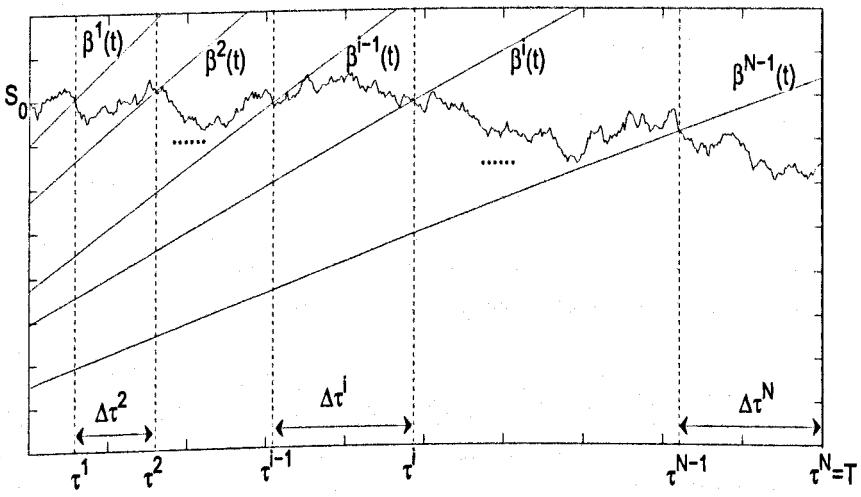
Unlike the standard LSLS method where asset values are simulated directly, asset values in the SCMC method are determined from simulated hitting times. Write S_j^i for an asset value at the ith contour on the jth path, it is the value of a contour in (6.1.1) evaluated at a hitting time τ_j^i.

Suppose the SC option has N (random) exercise dates. For a path $j = 1, \ldots, M$, the method simulates a hitting time sequence $\left\{\tau_j^i\right\}_{i=1,\ldots,N-1}$, to a set of

[1] Note that Δt_i do not necessarily have to be equal.



(a) Bermudan option's exercise opportunities, $\{t_i\}_{i=1,\ldots,N}$.



(b) SC option's exercise opportunities, $\{\tau_i\}_{i=1,\ldots,N}$.

Figure 6.1. Bermudan and SC option's exercise opportunities

contours $\{\beta^i(t)\}_{i=1,\ldots,N-1}$, so that τ_j^1 is a first hitting time of S to $\beta^1(t)$, starting from $S_{t_0} = S_0$, and τ_j^i is a first hitting time to $\beta^i(t)$ starting at $S_{\tau_j^{i-1}} = \beta^{i-1}\left(\tau_j^{i-1}\right)$ at time τ_j^{i-1}. At this stage there is no restriction on τ_j^i. In particular, one allows $\tau_j^i > T$ the maturity date of the SC option. Figure 6.2 illustrates hitting times and asset values in the SCMC method.

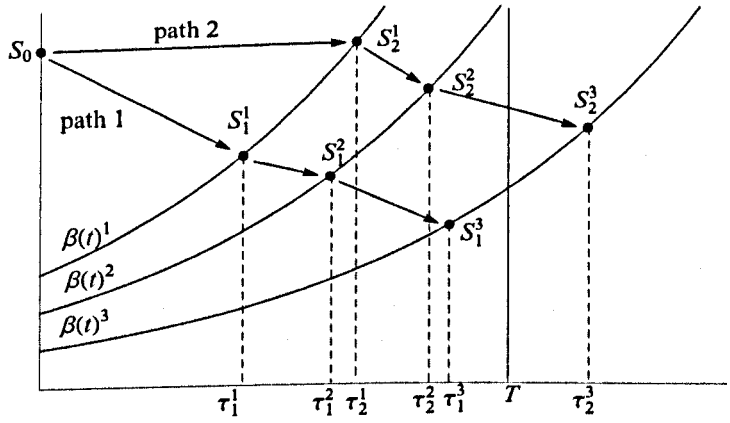


Figure 6.2. Illustration of path construction

Figure 6.2 shows two simulated paths with three exponential contours. An initial asset value is S_0. Points on the contours represent asset values S_j^i that are obtained from simulated values of hitting times τ_j^i. τ_j^i can be greater than the option's maturity T. This is the τ_2^3 in figure 6.2.

Note that the SCMC method uses forward path sampling. It generates sample paths forward through time. Then it stores an entire set of sample paths, both hitting times and asset values, and then iterates backward from contour $\beta^i(t)$ to contour $\beta^{i-1}(t)$, $i = N, \ldots, 1$.

The hitting time path will be described in more detail next.

6.1.2.1 Generalisation: The $\mathscr{T}^T$ path

In this section the definition of hitting times path and slice is modified so that it includes sample points only for times $t \leq T$, and ends with a sample from time T.

Let $\tau^i, i = 0, \ldots, N-1$, be a sequence of strictly increasing hitting times as defined in (6.1.2), with $\tau^0 \equiv 0$. A hitting time path is sample $\tau = (\tau^0, \ldots, \tau^{N-1})$ so that for $i_1 < i_2$ one has $\tau^{i_1} < \tau^{i_2}$. The SCMC method constructs a set of paths $\tau_j = \left(\tau_j^0, \ldots, \tau_j^{N-1}\right)$, $j = 1, \ldots, M$, as described in section 6.1.2. Write

$\mathscr{T} = \{\tau_j\}_{j=1,\ldots,M}$ for the complete set of hitting times paths. The ith slice is the set $\{\tau_j^i\}_{j=1,\ldots,M}$. This is a sample of hitting times τ^i to the contour $\beta^i(t)$.

Write τ^T for the constant hitting time T. Given a path $(\tau^0, \ldots, \tau^{N-1})$, suppose that $\tau^i \neq T$ for any i and $\tau^Q < \tau^T < \tau^{Q+1}$ for some $Q \in \{0, \ldots, N-1\}$. The T-modified path is the sample $(\tau^0, \ldots, \tau^Q, \tau^T, \ldots, \tau^T)$ where there are $N-Q$ independent samples from τ^T. From a set of paths $\mathscr{T} = \{\tau_j\}_{j=1,\ldots,M}$, set

$$\bar{\tau}_j^i = \tau_j^i \wedge \tau^T, \tag{6.1.9}$$

then the T-modified path is a set $\mathscr{T}^T = \left\{\bar{\tau}_j^i\right\}_{i=1,\ldots,N-1}$. This is illustrated in table 6.1.

τ path	$(0, \overbrace{0.2, 0.3, 0.5, 0.7, 0.9}^{N-1})$
T-modified path	$(0, \underbrace{0.2, 0.3, 0.5, 0.7, 0.9}_{Q})$

Panel (a): $Q = N-1$

τ path	$(0, \overbrace{0.2, 0.5, 0.9, 1.2, 1.5}^{N-1})$
T-modified path	$(0, \underbrace{0.2, 0.5, 0.9}_{Q}, \underbrace{1.0, 1.0}_{N-1-Q})$

Panel (b): $Q < N-1$

Table 6.1. T-modified path illustration, $T = 1$ and $N = 6$

Table 6.1 shows values of hitting times $\left\{\tau_j^i\right\}_{i=1,\ldots,N-1}$ on the jth path for different values of Q. Panel (a) illustrates a situation where $Q = N-1$. In this case, by using (6.1.9), the T-modified path is identical to the τ path. Panel (b) shows the T-modified path where $Q < N-1$. On the τ path, one can see that $Q = 3$ and $\tau^Q < \tau^T < \tau^{Q+1}$ $(0.9 < 1 < 1.2)$. By (6.1.9), the T-modified path is obtained.

6.1.2.2 Simulating S_T

On the τ path, one has simulated a set of points $(\tau^i, S^i), i = 1, \ldots, N-1$. Then one needs to simulate a value of an asset value at time to maturity, S_T. Because $\tau^T = T$, one can view simulating S_T as simulating a hitting time τ^T to a vertical contour. This implies that the Nth contour, $\beta^N(t)$, is always vertical. The Nth contour is defined as follows.

Definition 5. *The Nth contour, β^N, is vertical. It lies from 0 to $\beta^{N-1}(T)$.*

The value of Q will determine how S_T is generated. There are two cases to consider. These are:

Case 1: $Q = N - 1$

This is a case shown in panel (a) in table 6.1. When $Q = N - 1$, it implies that $\tau^i < T \; \forall i = 1, \ldots, N-1$. Let $\Delta\tau^Q = T - \tau^Q$. In this case, S_T is sampled conditional on S_{τ^Q}. This can be done by:

$$S_T = S_{\tau^Q} \exp\left(\left(r - \frac{1}{2}\sigma^2\right)\Delta\tau^Q + \sigma\sqrt{\Delta\tau^Q}\epsilon\right), \tag{6.1.10}$$

where $\epsilon \sim N(0,1)$.

Case 2: $Q < N - 1$

In this case, one truncates the τ path at Q, and samples $S_T \mid S_{\tau^Q}, S_{\tau^{Q+1}}$. This can be approximated by using linear interpolation between a pair (τ^Q, S_{τ^Q}) and $(\tau^{Q+1}, S_{\tau^{Q+1}})$. That is,

$$S_T = S_{\tau^Q} + (T - \tau^Q)\frac{S_{\tau^{Q+1}} - S_{\tau^Q}}{\tau^{Q+1} - \tau^Q}. \tag{6.1.11}$$

This situation corresponds to panel (b) in table 6.1. Note that (6.1.11) is only an approximation. The approximation in (6.1.11) will be used to value exotic American options in the next chapter, chapter 7. It will be discussed further in section 7.3.2.

On the jth path, there are a total of $Q + 1$ values of asset, $\left\{S_j^i\right\}_{i=1,\ldots,Q+1}$. When $Q = N-1$, a set of asset values is complete; that is there are N values of asset values for an option with N exercise dates. When $Q < N - 1$, a set of asset values is not complete. In other words, there are less than N asset values on the jth path. Since an option matures at time T, any values that are beyond time T are not needed.

When valuing a SC put option, a set of contours can be constructed such that asset values from the approximation in (6.1.11) will be greater than the option's strike, X, and thus contributes zero to the option's payoff. Since one uses only in-the-money sample paths to approximate option's continuation values, the computation in (6.1.11) is not needed for the case of valuing a put option. The contour construction will be discussed in section 6.2.

6.1.3 The SCMC algorithm: the generalisation of the LSLS algorithm

The standard LSLS algorithm iterates backwards from time t_N to time t_0 on vertical contours, approximating continuation values of a Bermudan option. Recall that τ^T is the constant hitting time T. Define $\tau^N \equiv \tau^T$ and $\tau^0 \equiv t_0 = 0$. The SCMC method generalises the LSLS method by instead iterating back from stopping time τ^N to stopping time τ^0 to exponential contours.

Let $h_j^i \equiv h\left(S_j^i\right) = \left(X - S_j^i\right)^+$ be a SC put option payoff with strike X at the ith contour on the jth path. The SCMC algorithm proceeds as follows:

1. Fix a set of contours $\beta = \{\beta^i(t)\}_{i=1,\ldots,N}$. Set $S_j^0 = S_0$ and $\tau_j^0 = 0$. Construct T-modified sample paths $\mathscr{T}_j^T = \left\{\bar{\tau}_j^0, \ldots, \bar{\tau}_j^N\right\}_{j=1,\ldots,M}$ and asset value paths $S_j = \left\{S_j^0, \ldots, S_j^N\right\}_{j=1,\ldots,M}$. Set $\widehat{v}_j^N = h_j^N$.

2. At each contour $\beta^i(t)$, $1 \leq i \leq N-1$, write $\widehat{C}_j^i$ for the continuation value. Write $\Delta\tau_j^i = \tau_j^{i+1} - \tau_j^i$. Set $\widehat{C}_j^i = \mathrm{e}^{-r\Delta\tau_j^i}\widehat{v}_j^{i+1}$.

3. Write $\widetilde{C}_j^i$ as an approximated continuation value. Find parameters $a^i = \{a_k^i\}_{k=1,\ldots,K}$ so that $\widehat{C}_j^i \sim \widetilde{C}_j^i = f\left(S_j^i \mid a^i\right)$, where

$$f\left(S_j^i \mid a^i\right) = \sum_{k=1}^{K} a_k^i \psi_k\left(S_j^i\right), \tag{6.1.12}$$

for a chosen set of basis functions $\{\psi_k\}_{k=1,\ldots,K}$. Parameters a^i can be found by

$$a^i = \arg\min_a \left\| \widehat{C}_j^i - f\left(S_j^i \mid a\right) \right\|. \tag{6.1.13}$$

4. The SC option is exercised if $h_j^i > \widetilde{C}_j^i$. On each contour, $\beta^i(t)$, set $\mathbb{S}^E = \left\{S_j^i \mid h_j^i > \widetilde{C}_j^i\right\}$. $\mathbb{S}^E$ is the set of asset values at contour $\beta^i(t)$ at which the SC option is exercised. Then one sets

$$S_i^E = \max_S \left\{\mathbb{S}^E\right\}. \tag{6.1.14}$$

S_i^E is the greatest asset value at the ith contour that the SC option is exercised. The option's EEB is the set $S^E = \left\{S_i^E\right\}, i = 1, \ldots, N$.

5. The SC put option value along the ith contour is

$$\widehat{v}_j^i = \begin{cases} h_j^i, & S_j^i \le S_i^E, \\ \mathrm{e}^{-r\Delta\tau_j^i}\widehat{v}_j^{i+1}, & \text{otherwise.} \end{cases} \tag{6.1.15}$$

If $\tau_j^i > T$, then at each iteration step one constructs $\widehat{v}_j^{i-1}$ only for $i \le Q+1$. At time τ^0, the SC option value is $\widehat{v}_j^0 = \max\left(\frac{1}{M}\sum_{j=1}^{M}\mathrm{e}^{-r\tau_j^1}\widehat{v}_j^1, h^0\right)$.

Figure 6.3 illustrates an iteration of the SCMC method. It shows that the method works backward along contours. The approximation of continuation values and the exercise decision are made on the contour.

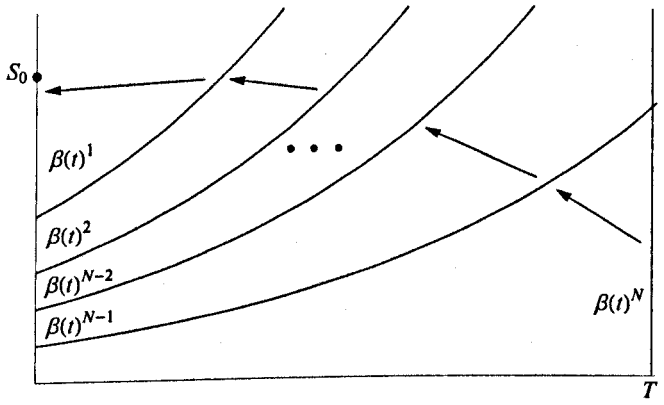


Figure 6.3. Iteration of the SCMC algorithm

6.1.4 Using SC put options to approximate American put options: Richardson extrapolation

The SCMC algorithm will approximate the value of a SC put option with N exercise dates. To obtain the approximated value of an American put option, the three-point Richardson extrapolation of Geske and Johnson (1984) [104] is applied to the values of a SC option with different exercise opportunities. The method is described in section 5.4. The only difference is that in section 5.4, values of a Bermudan put option are used. In this chapter, values of a SC option are used. In this case, the approximated value of an American put option, $\widehat{v}(0)$, is

$$\widehat{v}(0) = \widehat{v}^{64} + \frac{5}{3}\left(\widehat{v}^{64} - \widehat{v}_j^{32}\right) - \frac{1}{3}\left(\widehat{v}_j^{32} - \widehat{v}_j^{16}\right), \tag{6.1.16}$$

where $\widehat{v}^{64}$, $\widehat{v}^{32}$, and $\widehat{v}^{16}$ are values of a SC option with 64, 32, and 16 exercise opportunities respectively (see section 5.4 for the derivation of (6.1.16)).

6.2 Sequential contour construction

The shape of contours determines the value of a SC option. A SC option is specified by (i) a specification of the set of contours that determine the exercise opportunities and (ii) a specification of the payoff at exercise dates. A set of contours is determined by a set of pair $P = \{(\alpha^i, g^i)\}_{i=1,\ldots,N}$. α^i determines the time t_0 value of the ith contour, $\beta^i(0)$, and g^i is the contour growth rate and determines a steepness of the contour.

The SCMC method is used to value American options. To facilitate this, one chooses a suitable SC option. To value an American put option, the payoff of a SC option is a put. To choose a set P appropriately, assume that there are four parameters to consider. These are

1. α^0: time 0 value of the first contour.
2. α^{N-1}: time 0 value of the $(N-1)$th contour.
3. $g^i, i = 1, \ldots, N-2$: the growth rate of the ith contour.
4. g^{N-1}: the growth rate of the $(N-1)$th contour.

This section starts by first discussing g^{N-1}.

6.2.1 Choice of g^{N-1}

For the American put option valuation problem, one chooses (α^{N-1}, g^{N-1}) such that

$$\beta^{N-1}(T) = X. \tag{6.2.1}$$

This is because any values of τ_j^N, $j = 1, \ldots, M$, that is greater than T will contribute zero to a put payoff and hence values computed from the approximation in (6.1.11) are not used. To choose g^{N-1}, write

$$\beta^i(0) = \frac{1}{\alpha^i}, \tag{6.2.2}$$

then by using (6.1.1) and (6.2.1), one has

$$g^{N-1} = \ln\left(\frac{X}{\beta^{N-1}(0)}\right)\frac{1}{T}. \tag{6.2.3}$$

Other parameters are described next.

6.2.2 Choices of α^0, α^{N-1}, and $g^i, i = 1, \ldots, N-2$

A set $P = \{(\alpha^i, g^i)\}_{i=1,\ldots,N}$ can be categorised into four types depending on α^i and g_i. Since a SC option is specified by a set of contours (and a payoff at exercise dates), one can view different types of contour as different types of SC options. Four types of SC options are summarised in table 6.2.

SC option types		g^i	
		fixed	varied
α^i	fixed	-	3,4
	varied	1	2

Table 6.2. Summary of SC option types

Note that cases 3 and 4 are different ways of varying g^i. Each type of SC options is discussed are follows.

Type (1): α^i varied and g^i fixed $(g^i = g)$

First, the choice of α^1 is described. All contours should cross the American put's EEB because every sample path can potentially contribute to the exercise decision. Thus, $\beta^1(0) = \frac{1}{\alpha^1}$ must be less than the EEB. Even though the American put's EEB is not known in advance, the perpetual American put EEB, S_p^E, is known. It is in the form:

$$S_p^E = \frac{X}{1 + \sigma^2/2r}. \tag{6.2.4}$$

Because the holder of a perpetual American put can exercise the option anytime up to infinite, the EEB is flat and lower than the standard (finite horizon) American put. Thus one can set $\alpha^1 = 1/S_p^E$.

Now α^{N-1} is discussed. Since g^i is fixed, α^{N-1} should be further away from α^1 so that each contour intersects along the American put's EEB. Choose some value for $\alpha^{N-1} \gg \alpha^1 \in \mathbb{R}^+$ such that $\beta^{N-1}(0) \ll \beta^1(0)$. The subsequent values α^i, $i = 2, \ldots, N-2$, are set so that

$$\alpha^i = \frac{\alpha^{i-1}}{1 - \Delta\alpha^{i-1}}, \tag{6.2.5}$$

where

$$\Delta = \frac{\alpha^{N-1} - \alpha^1}{\alpha^1\alpha^{N-1}N - 1}. \tag{6.2.6}$$

Since g is fixed in this type, set $g = g^1 = g^2 = \ldots = g^{N-1}$ in (6.2.3).

Type (2): α^i varied and g_i varied

α^1 and α^{N-1} are set as in type (1). However, the growth rate g can be varied in this case. One chooses g^i such that

$$\beta^i(t_i^*) = X, \tag{6.2.7}$$

where

$$t_i^* = \frac{i}{N-1}. \tag{6.2.8}$$

Equation (6.2.7) means that t_i^* is the time that the ith contour, β^i, has a value X. This level of X is roughly used to approximate where contours intersect the American option's EEB. Write t_i^E as the time that the β^i intersects the American option's EEB. Clearly, as contours get steeper, t_i^* is closer to t_i^E.[2] Even though the use of t_i^* is only a rough approximation, it provides a general idea of the location of exercise opportunities of the option. Hence, by setting t_i^* as in (6.2.8), one roughly spreads exercise opportunities evenly along the American put's EEB.

For each $\beta^i(t), i = 1, \ldots, N-2$ and $t_i^* = \frac{i}{N-1}, i = 1, \ldots, N-1$, g^i is set as:

$$g^i = \ln\left(\frac{X}{\beta^i(0)}\right)\frac{1}{t_i^*}, \tag{6.2.9}$$

and set g^{N-1} as in (6.2.3).

Type (3): α^i fixed $(\alpha^i = \alpha)$ and g^i varied: (I)

First, set α such that $\beta(0) \ll S_p^E$. The reason $\beta(0)$ should be a low value because if $\beta(0)$ is too high, contours will be too shallow and hence there may be not sufficient in-the-money paths.[3] g^i is set as in (6.2.9).

Type (4): α^i fixed $(\alpha^i = \alpha)$ and g^i varied: (II)

This type of contour is a variant of type (3). Unlike a type 3 contour where t_i^* is given by (6.2.8), one can control a set $t^* = \{t_i^*\}_{i=1,\ldots,N-2}$ more freely in this case. This means one has more freedom in controlling a shape of contours. Also, by being able to choose a set t^*, one can roughly control the location of SC option's exercise opportunities along the time horizon.

[2]In particular, for vertical contours (standard LSLS method), $t_i^* = t_i^E$.
[3]In numerical examples, $\beta(0) = 5$ for $X \sim 100$.

To do this, set the growth rate on the ith contour, g^i, to be a function of function, $k(i)$,

$$k(i) = k_{\max} - i\Delta k^-, \tag{6.2.10}$$

with

$$\Delta k^- = \frac{k_{\min} - k_{\max}}{N-1}, \tag{6.2.11}$$

where $k_{\min} \leq k_{\max}$ and $k_{\min}, k_{\max} \in \mathbb{R}^+$. $k(i)$ is a decreasing function of i.

Then the growth rate function, $g(k(i))$, is defined to be:

$$g(k(i)) = \left(\frac{N-1}{i}\right)^{k(i)} g^{N-1}, \tag{6.2.12}$$

where g^{N-1} is defined in (6.2.3).

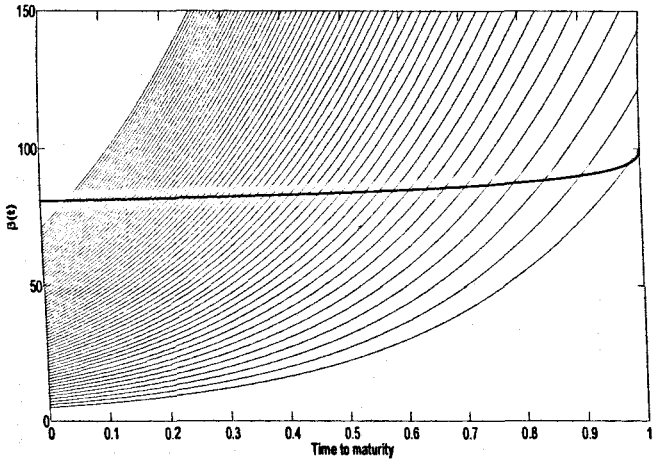
The value of a set $t^* = \{t_i^*\}_{i=1,\ldots,N-2}$ is determined through parameters $k_{\max}$ and $k_{\min}$. Recall a definition of t_i^* in (6.2.7). The parameter $k_{\max}$ controls the value of t_1^*. If $k_{\max}$ is small, then t_1^* will be large which means contours will be shifted toward an option's maturity. The parameter $k_{\min}$ controls the compactness of contours. The smaller the value of $k_{\min}$ is, the more clustered contours are toward an option's maturity. An analysis of $k_{\max}$ and $k_{\min}$ is in section 6.4.[4]

Plots of examples of four different types of contour are shown in figure 6.4. Figure 6.4 shows the EEBs of an American put option in bold lines. They are computed by using a lattice method with 20,480 time steps. Parameter values are $S_0 = 100, X = 100, r = 0.05, \sigma = 0.2$ and $T = 1$. Panel (a) shows type 1 contour where $\alpha = \{\alpha^i\}_{i=1,\ldots,N-1}$ is varied and g^i is fixed. One can see that contours are clustered at the initial segment of the option's EEB and spread out at the near to maturity segment. Panel (b) illustrates type 2 contour where both α^i and g^i are varied. Now contours are more evenly spread out horizontally along the EEB. Panel (c) shows type 3 contour where α^i is fixed and g_i is varied. Contours also spread out more evenly along the EEB in this case. Lastly, panel (d) illustrates type 4 contour where g^i is now a function of $k(i)$ in (6.2.10). One can see that contours spread out almost evenly along the EEB. They are slightly more clustered at the near-maturity segment of the EEB.

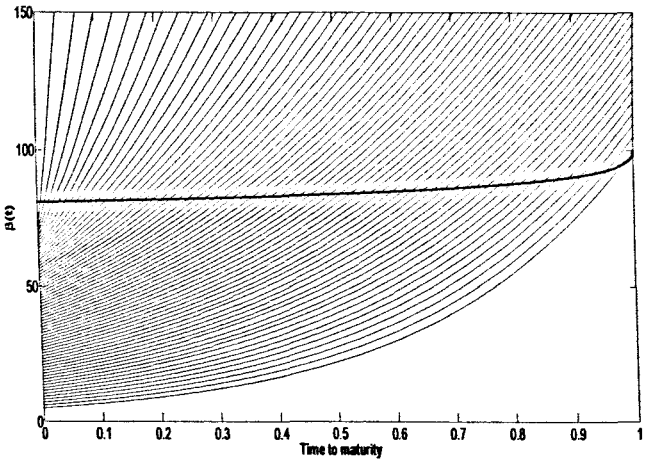
Simulated asset values along the contours of type 4 are plotted in figure 6.5. Parameter values are those used to produce a plot in panel (d) in figure 6.4.

In figure 6.5, since hitting times to the first contour are all conditional on the initial value S_0, most asset values are very close to each other on the first contour. As sample paths are generated from one contour to the next contour, asset values spread

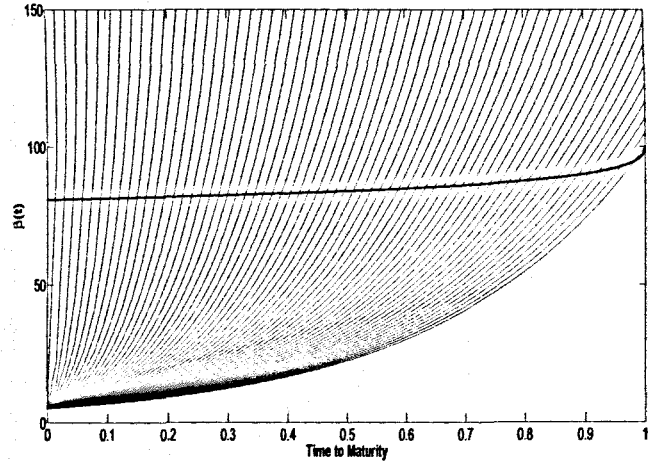
[4]The different values of $k_{\max}$ and $k_{\min}$ define different values of SC options.



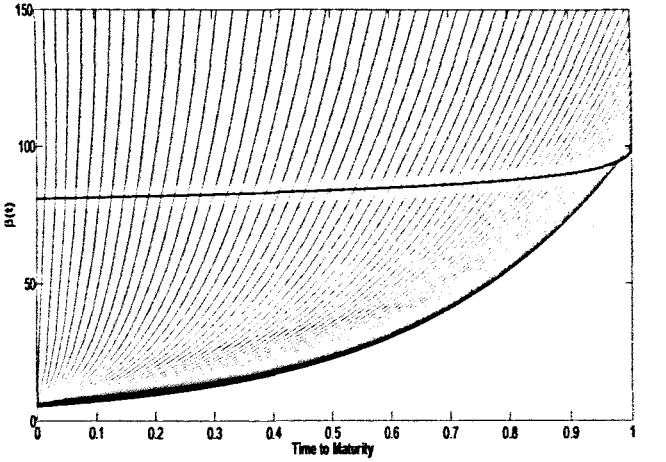
(a) Type (1): $\beta_1(0) = S_p^E = 71.43, \beta_{N-1}(0) = 5$ and $g = 2.996$



(b) Type (2): $\beta_1(0) = S_p^E = 71.43, \beta_{N-1}(0) = 5$ and g^i as in (6.2.9)



(c) Type (3): $\beta(0) = 5$ and g^i as in (6.2.9)



(d) Type (4): $\beta(0) = 5$ and g^i as in (6.2.12) where $k_{\max} = 1$ and $k_{\min} = 0.1$

Figure 6.4. The four types of sequential contours

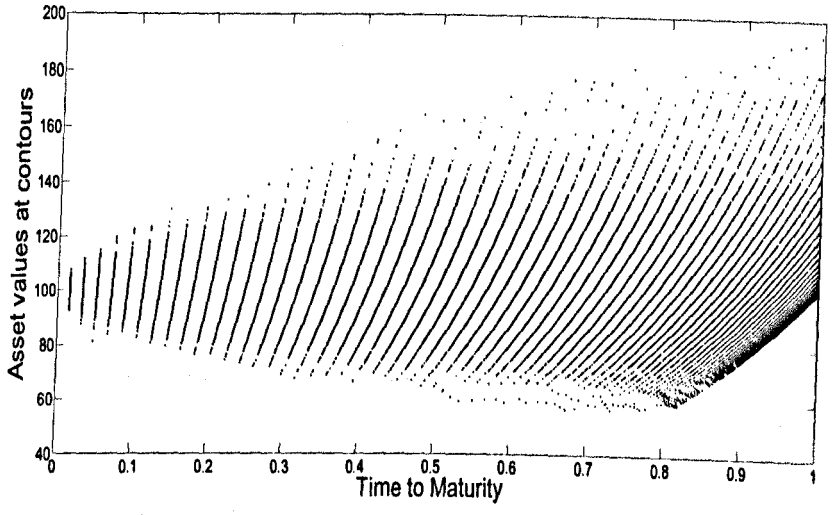


Figure 6.5. Asset value simulation from the SCMC method

out more along the contour. At time to maturity T, one can see that at the $\beta^N(t)$ contour (the vertical contour where $T = 1$), there are simulated asset values along the contour. These points are generated by methods described in section 6.1.2.2.

6.3 Control variates from the SCMC method

For $i = 1, \ldots, N-1$, the SCMC method generates hitting times to the ith contour, $\beta^i(t)$. These hitting times can be used to value barrier options with a barrier $b^i(t) = \beta^i(t)$. Section 6.3.1 discusses the valuation of a barrier option using hitting time simulation. Then, section 6.3.3 describes how barrier option's values are obtained from the SCMC method.

6.3.1 Valuing barrier options by using hitting time simulation

Define S_t to be an asset value at time t. Let $b(t)$, $t \in [t_0, T]$, be a barrier level and suppose that the hitting time distribution of S_t to $b(t)$ is known and can be sampled from, then a rebate option (to a barrier $b(t)$) expiring at T can be valued easily by Monte Carlo simulation. Recall from chapter 3 that a rebate option pays a rebate R at the hitting time τ to the barrier $b(t)$, conditional on $\tau \leq T$.

Write $f^b(\tau)$ for the hitting time distribution of S_t to $b(t)$, conditional on S_0. Let R_t be the rebate paid if S_t hits $b(t)$ at time t. Then the value v_t^R of a rebate

option at time t is

$$v_t^R = \int_t^T \mathrm{e}^{-r(s-t)} R_s \mathrm{d}f^b(s). \tag{6.3.1}$$

Let M be the total number of sample paths. For path $j = 1, \ldots, M$, sample hitting times $\tau_j \sim f^b$ and set

$$\widehat{v}_t^R = \frac{1}{M} \sum_{j=1}^{M} \mathrm{e}^{-r(\tau_j - t)} R_{\tau_j}, \quad t < \tau_j^i, \tag{6.3.2}$$

$\widehat{v}_t^R$ is a Monte Carlo estimate of v_t^R. The use of hitting times to value barrier options is important to the SCMC method because it can be used to compute control variates along the contours.

6.3.2 Hitting times and options

Given a stopping time τ and maturity time $\tau^T = T$, set $\hat{\tau} = \min\{\tau, \tau^T\}$. Let P_T be an option's payoff at time T. Then the set $\mathcal{O} = (\tau, \tau^T, R, P)$ defines several options. These are shown in table 6.3.

	Cash flows paid at		
	$\hat{\tau} = \tau$		$\hat{\tau} = \tau^T$
Option type	τ	T	
Knock-out option with rebate	R_τ	-	P_T
Knock-out option	-	-	P_T
Knock-in option with rebate		P_T	R_T
Knock-in option	-	P_T	-
Rebate-τ option	R_τ	-	-
Rebate-T option	-	R_T	-

Table 6.3. Options defined by hitting times

Let $b : [0, T] \to \mathbb{R}^+$ be a deterministic barrier level and suppose that $\tau = \tau^b = \min\{t \in [0, T] : S_t \leq b(t)\}$ is the first hitting time of S_t to $b(t)$, then $\mathcal{O}$ is a barrier option and the knock-out, knock-in and rebate options listed in table 6.3 correspond to the vanilla versions.

A knock-out option with rebate pays out a rebate value, R_τ, at a hitting time τ if a barrier is hit before time T. If a barrier is not hit; that is $\hat{\tau} = \tau^T$, it will pay out a payoff P_T at time T. If a knock-out option has no rebate, it will pay out only P_T at time T conditional on a barrier not being hit. A knock-in option with rebate pays out a payoff P_T at time T if a barrier is not hit before time T. A rebate R_T is

paid at time T if a barrier is not hit. If a knock-in option has no rebate attached, it will pay out P_T only when a barrier is hit.

A rebate-τ option pays out R_τ when a barrier is hit at τ. It can be viewed as a rebate attached to a knock-out option. A rebate-T option pays out R_T at time T when a barrier is hit before time T, $\hat{\tau} = \tau$. This can be viewed as a rebate attached to a knock-in option.

Also, suppose that $R_t = (X - S_t)^+$ and $P_T = R_T$ then the stopping time $\hat{\tau}$ defines an exercise strategy for an American put option.

The advantage of the SCMC method is that if the stopping times τ^i can be augmented with payoff functions R^i so that for each i the set $\mathcal{O}^i = (\tau, \tau^T, R, P)$ defines an option whose value can be computed explicitly then the set of $\mathcal{O}^i$ can be used as control variates. Their effectiveness as control variates depends upon the closeness of their values to that of the underlying American option.

Three examples are

1. $\tau^i = t_i$ with $R_{t_i} = (X - S_{t_i})^+$. This is a European put maturing at t_i.
2. $\tau^i = \tau^{b^i}$, where $b^i = b$ a constant, and $R_{\tau^i} \equiv R$, also constant. This corresponds to valuing a flat barrier rebate option with a rebate R.
3. $\tau^i = \tau^{b^i}$, where $b^i = \frac{1}{\alpha^i}\exp(g^i t)$ is an exponential barrier, and $R_{\tau^i} \equiv R$ is constant. This is an exponential barrier rebate option with rebate R^i.

Because contours in the SCMC method are exponential, it is straightforward to value exponential barrier rebate options in example 3 along the contours. This is described next.

6.3.3 Rebate options from the SCMC method

The SCMC method generates a set of hitting times $\tau^i = \left\{\tau^i_j\right\}_{j=1,\ldots,M}$ along the ith contour, where $\tau^i = \min\left\{t \mid S_t \leq b^i(t)\right\}$ and $b^i(t) = \beta^i(t)$ is an exponential contour. Define an ith rebate-τ option with a payoff, h^R_τ

$$h^R_\tau = R\mathbb{I}_{\tau^i < T}, \tag{6.3.3}$$

where $\mathbb{I}$ is an indicator function and $R \in \mathbb{R}^+$. From section 6.3.1, the time t_0 value of a rebate option on the ith contour can be computed by Monte Carlo simulation. It is $\widehat{v}^{i,R}_0 = \frac{1}{M}\sum_{j=1}^{M} \mathrm{e}^{-r\tau^i_j} h^R_\tau$. Effectively, as the method constructs a whole hitting time path $\left\{\tau^i_j\right\}^{i=1,\ldots,N-1}_{j=1,\ldots,M}$, one has a set of rebate option values $\widehat{v}^R_0 = \left\{\widehat{v}^{R,i}_0\right\}_{i=1,\ldots,N-1}$.

6.3.4 Rebate option control variate

Recall that $c^x = \widehat{x} - \mathbb{E}[x]$ is a control variate, where $\widehat{x}$ is a chosen quantity and $\mathbb{E}[x]$ is known. A set of rebate option values $v_0^R = \left\{v_0^{R,i}\right\}_{i=1,\ldots,N-1}$ can be used as additional control variates to control variates discussed in section 5.2.2. In this case one has multiple controls:

$$\widehat{v}^{cv} = \widehat{v} - \sum_{l=0}^{L} \omega^l c^l, \tag{6.3.4}$$

where $\omega^l = \text{Cov}\left[\widehat{v}, \widehat{x}^l\right] / \text{Var}\left[\widehat{v}\right]$, $L \leq N+1$ is the number of control variates and $\left\{c^l\right\}^{l=0,\ldots,L}$ is a set of selected control variates. $\widehat{v}$ is a uncontrolled value of the SC option. In this chapter, set c^0 to be the put-tau control variate, $c^0 = c_\tau^p$ in equation (4.3.44) on page 83. Also set c^l, $l = 1, \ldots, L$ to be a rebate option (to the ith contour) control variate. First, define $c_T^{R,i}$ to be a rebate option to the ith contour control variate. It can be found by

$$c_T^{R,i} = \widehat{v}_0^{R,i} - v_0^{R,i}, \tag{6.3.5}$$

where $v_0^{R,i}$ is the time t_0 analytical value of a rebate option to the ith contour.[5] When an option barrier is exponential, the value $v_0^{R,i}$ is known. This was given by Ingersoll (1998) [134] as follows.

$$v_0^{R,i} = R\left[\left(\frac{b_0^i}{S}\right)^{\nu'-\alpha'} \mathrm{N}\left(\phi_1\left(\frac{b_0^i}{S}, \alpha'\right)\right) - \left(\frac{b_0^i}{S}\right)^{\nu'+\alpha'} \mathrm{N}\left(\phi_2\left(\frac{b_0^i}{S}, \alpha'\right)\right)\right], \tag{6.3.6}$$

where

$$\begin{aligned}
\mathrm{N}(.) &\equiv \text{Cumulative normal distribution function} \\
\phi_1(x,y) &\equiv \frac{\ln(x) - y\sigma^2 T}{\sigma\sqrt{T}} \\
\phi_2(x,y) &\equiv \phi_1(x,y) + 2y\sigma\sqrt{\phi_1(x,y)} \\
\nu' &\equiv \frac{r - g - \frac{1}{2}\sigma^2}{\sigma^2} \\
\alpha' &\equiv \sqrt{\nu'^2 + \frac{2r}{\sigma^2}}.
\end{aligned}$$

It is important that one should not include too many options (large L) in (6.3.4) since this may introduce multicollinearity among the control variates and

[5] A subscript T in $c_T^{R,i}$ means that its value is obtained from hitting times to exponential contour, not from exercise times of a SC option.

will reduce the effectiveness of the control variate method. In fact, the numerical results reported in this chapter are produced using $L = 1$. That is,

$$\widehat{v}^{cv} = \widehat{v} - \omega^0 c^0 - \omega^1 c^1, \tag{6.3.7}$$

$$= \widehat{v} - \omega^p c_\tau^p - \omega^{R,i} c_T^{R,i} \tag{6.3.8}$$

6.4 Numerical Results

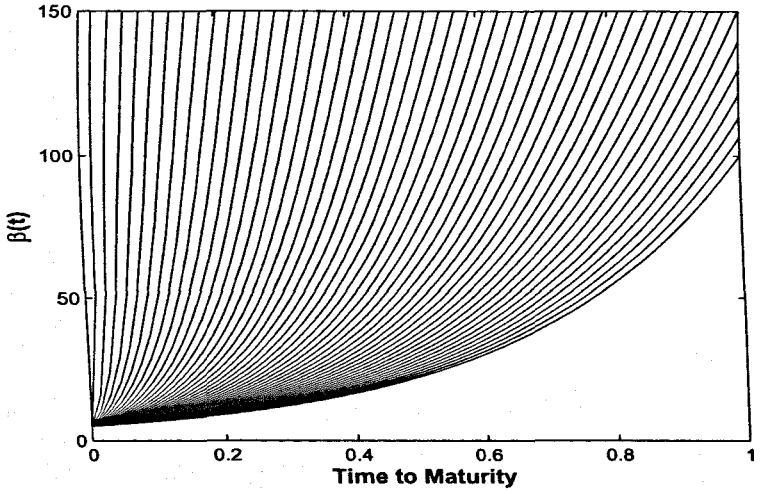
In this section the SCMC method is implemented to value SC options that pay $(X - S_\tau)^+$, where X is a strike price and S_τ is an asset value, if exercised at time τ. Then, to obtain the value of standard American put options, three-point Richardson extrapolation is applied to these values for SC options of the same type but different values of exercise dates, N.

Parameter values, unless stated otherwise, are $X = 100$, $r = 0.05$, $\sigma = 0.2$ and $T = 1$. The number of sample paths, M, is 50,000. Benchmark values are computed using a lattice method with $N = 204,800$ time steps. The SCMC method uses scaled Laguerre basis function (section 5.3.1) where number of basis functions $K = 12$.

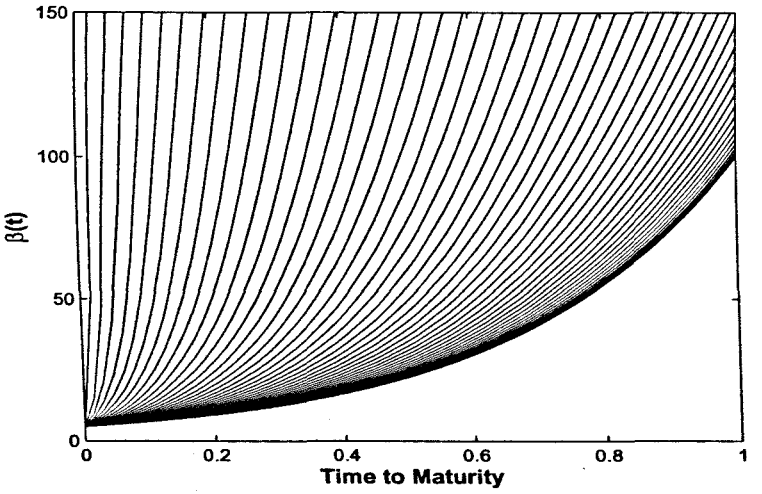
6.4.1 Choice of contours

First, the effect of $k_{\max}$ and $k_{\min}$ on the value of type 4 SC option and on extrapolated American put values is investigated. Table 6.4 illustrates this. It reports extrapolated option values, standard errors in brackets, computational times in square brackets and biases in curly brackets. Biases are computed using (3.3.1). Pricing and rollback control variates are the put-tau (section 4.3.6).

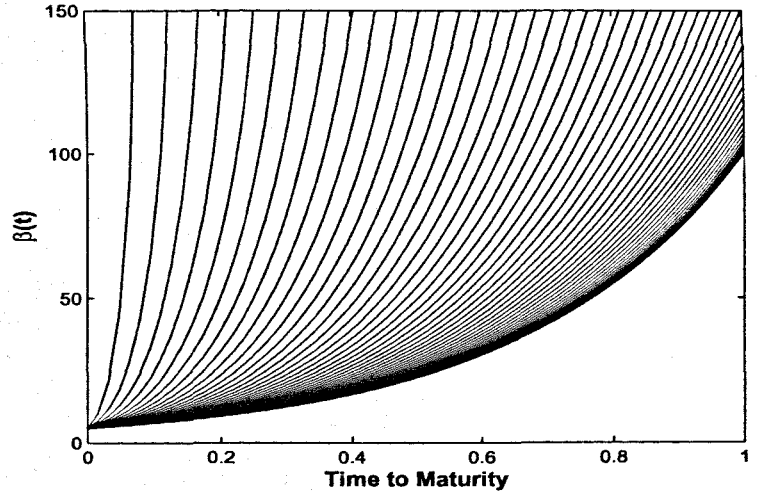
Tables 6.4 reports extrapolated American put value with different values of $k_{\max}$ and $k_{\min}$. The value of $k_{\max}$ ranges from 0.7 to 1. Since $k_{\max}$ varies inversely with t_1^*, smaller values of $k_{\max}$ push contours toward an option's maturity. This is illustrated in figure 6.6. When $k_{\max} = 0.7$, t_1^* is $\sim$0.1 which is nearly 10% of an option's maturity ($T = 1$). Since further reducing $k_{\max}$ means an option holder will have less chance of exercising in the initial segment of an option's lifetime, $k_{\max}$ will not be further reduced. $k_{\min}$ ranges from 0.1 to $k_{\max}$. Results in table 6.4 show that there is no evidence of bias in American put values. As $k_{\max}$ decreases, the bias is larger for values of $k_{\min}$ that are close to $k_{\max}$. However, these values do not seem to be significant. Therefore, to use type 4 contour, $k_{\max}$ is set to be a high value that is close to 1 while $k_{\min} \leq k_{\max}$.



(a) $k_{\max} = 1, k_{\min} = 1$



(b) $k_{\max} = 1, k_{\min} = 0.1$



(c) $k_{\max} = 0.7, k_{\min} = 0.1$

Figure 6.6. Contour shape with different $k_{\max}$ and $k_{\min}$. $N = 50$, $\beta^{N-1}(0) = 5$ and $g^{N-1} = 2.99$.

$k_{\max}$	$k_{\min}$									
	1	0.9	0.8	0.7	0.6	0.5	0.4	0.3	0.2	0.1
1	6.0891 (0.0016) [191] {-0.78}	6.0890 (0.0016) [190] {-0.87}	6.0894 (0.0016) [190] {-0.63}	6.0904 (0.0016) [189] {0.03}	6.0887 (0.0015) [189] {-1.1}	6.0905 (0.0016) [190] {0.08}	6.0904 (0.0016) [190] {0.03}	6.0912 (0.0016) [189] {0.49}	6.0915 (0.0016) [188] {0.69}	6.0911 (0.0016) [188] {0.48}
0.9		6.0878 (0.0016) [189] {-1.6}	6.0885 (0.0016) [189] {-1.2}	6.0879 (0.0016) [190] {-1.6}	6.0905 (0.0016) [189] {0.10}	6.0904 (0.0016) [190] {-0.01}	6.0894 (0.0015) [188] {-0.65}	6.0884 (0.0016) [190] {-1.3}	6.0899 (0.0016) [188] {-0.31}	6.0911 (0.0016) [189] {0.47}
0.8			6.0882 (0.0015) [190] {-1.4}	6.0888 (0.0016) [190] {-1.0}	6.0889 (0.0016) [190] {-0.92}	6.0892 (0.0016) [191] {-0.74}	6.0895 (0.0016) [188] {-0.54}	6.0914 (0.0016) [190] {0.66}	6.0914 (0.0016) [188] {0.63 }	6.0887 (0.0015) [188] {-1.1}
0.7				6.0885 (0.0016) [188] {-1.2}	6.0877 (0.0016) [188] {-1.7}	6.0918 (0.0016) [188] {0.93}	6.0922 (0.0015) [188] {1.2}	6.0912 (0.0016) [189] {0.56}	6.0897 (0.0016) [188] {-0.43}	6.0917 (0.0016) [187] {0.84}

Table 6.4. Extrapolated American put values from type 4 SC option with different values of $k_{\max}$ and $k_{\min}$. $\beta^{N-1}(0) = 5$ and $g^{N-1} = 2.99$. Benchmark value is 6.09037

Next, four different type of contours (section 6.2) are investigated. Contour parameters are reported in table 6.5.

Type	Parameters
1: $\beta_i(0), g$	$\alpha^1 = 1/S_p^E = 1/85.71, \alpha^{N-1} = 1/5, g = 2.996$
2: $\beta_i(0), g(i)$	$\alpha^1 = 1/S_p^E = 1/85.71, \alpha^{N-1} = 1/5, g^i$ is given by eq. (6.2.9)
3: $\beta(0), g(i)$	$\alpha^{N-1} = 1/5, g^i$ is given by eq. (6.2.9)
4: $\beta(0), g(i, k(i))$	$\alpha^{N-1} = 1/5, k_{\max} = 1$, and $k_{\min} = 0.1$

Table 6.5. Contour parameter values

Table 6.6 reports extrapolated American put values computed from the SCMC method with different types of contours. It reports extrapolated American put values $\widehat{v}^{\infty}$, standard errors (se), computational times [t] and biases {b}. The put-tau pricing and rollback control variates are applied to individual SC option values.

It is clear that the type 1 contour is heavily high-biased. This is because most of the contours are in the initial segment of the option's life. This means an option can be exercised predominately in the the early part of its lifetime. Payoffs will be discounted back to time $\tau^0 = t_0 = 0$ with lower discount factor. This results in a high bias in option values. The bias becomes greater as an option gets deeper in-the-money. This is because more paths will be exercised.

When the exercise decision spreads more evenly along the option's EEB, as for types 2-4, the biases are all within the range ± 2. Recall from (6.2.7) that t_i^* is the time that $\beta^i(t)$ has a value X, and it is used to roughly estimate where β^i intersects the American option's EEB. Since it is easy to modify the location of t_i^* along the option's EEB when using type 4 contours through $k_{\max}$ and $k_{\min}$, type 4 contours is chosen to be the contour type of choice in this chapter.

$S0, \sigma = 0.2$ Benchmark		LSLS	Contour Type			
			1	2	3	4
90	$\widehat{v}^{\infty}$	11.4909	11.5258	11.4949	11.4886	11.4943
11.4927	(se)	(0.0023)	(0.0023)	(0.0023)	(0.0022)	(0.0022)
	$[t]$	[298]	[325]	[312]	[303]	[297]
	{b}	{-0.8}	{14}	{0.93}	{-1.8}	{0.72}
95	$\widehat{v}^{\infty}$	8.4455	8.4683	8.4556	8.4494	8.4540
8.4510	(se)	(0.0019)	(0.0022)	(0.0024)	(0.0019)	(0.0019)
	$[t]$	[249]	[264]	[258]	[255]	[248]
	{b}	{-2.9}	{8.0}	{1.9}	{-0.84}	{1.6}
100	$\widehat{v}^{\infty}$	6.0895	6.1016	6.0938	6.0876	6.0908
6.0904	(se)	(0.0016)	(0.0018)	(0.0021)	(0.0016)	(0.0016)
	$[t]$	[184]	[201]	[201]	[195]	[193]
	{b}	{-0.53}	{6.1}	{1.6}	{-1.8}	{0.25}
105	$\widehat{v}^{\infty}$	4.3036	4.3068	4.3063	4.3035	4.3049
4.3044	(se)	(0.0013)	(0.0016)	(0.0017)	(0.0013)	(0.0012)
	$[t]$	[129]	[139]	[141]	[137]	[141]
	{b}	{-0.64}	{1.5}	{1.1}	{-0.76}	{0.40}
110	$\widehat{v}^{\infty}$	2.9866	2.9871	2.9862	2.9847	2.9857
2.9865	(se)	(0.0010)	(0.0012)	(0.0014)	(0.0010)	(0.0010)
	$[t]$	[92]	[92]	[101]	[98]	[103]
	{b}	{0.09}	{0.43}	{-0.23}	{-1.8}	{-0.88}

Table 6.6. Extrapolated American put option values from the SCMC method with different types, $\sigma = 0.2$

6.4.2 Early exercise boundary of sequential contour options

EEBs of SC options are shown in this section. Unlike a standard Bermudan option (with constant Δt where $\Delta t = T/N$) in which the EEB can be benchmarked against values computed from the lattice method, there is no benchmark for sequential contour option's EEB. Therefore, the EEB of the SC options will be compared with that of an American put option from a lattice method. Since the holder of an American option can choose to exercise (continuously) at any time, one expects the

EEB of the sequential contour option to lie above that of the American put.

To obtain the option's EEB, the put-tau rollback control variate (section 5.3.1) is implemented. The EEB is shown in figure 6.7.

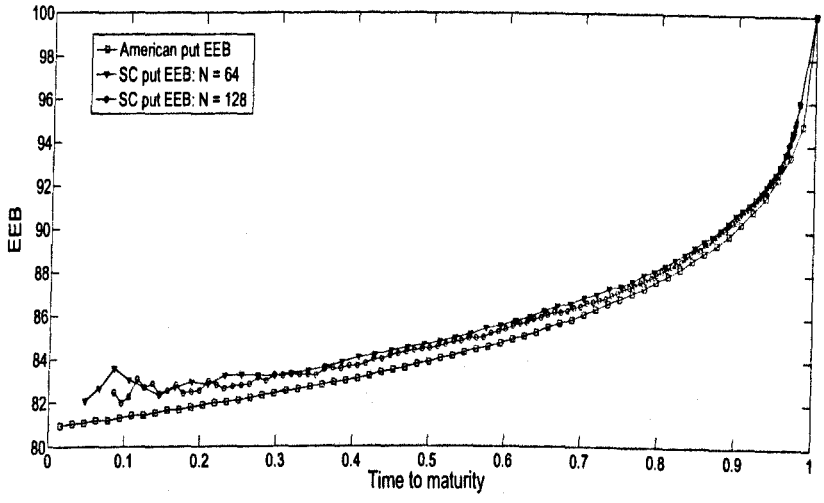


Figure 6.7. Early exercise boundaries of the sequential contour puts and the American put

One can see, from figure 6.7, that both of the sequential contour's EEB lie above the American put's EEB. It has a convex shape as one would expect for a put option. Also as the total number of contours N is increased from 64 to 128, the EEB of SC options get closer to that of the American put. The SC option's EEB shifts down slightly toward the American option's EEB.

6.4.3 Valuing standard American put options

In this section, the SCMC method is implemented to approximate values of the American put options. First, its convergence and accuracy are investigated and compared with those from the standard LSLS method. Second, the effectiveness of rebate option control variates is assessed, and the efficiency gain over the LSLS method with the put-tau pricing control variate is investigated.

6.4.3.1 Convergence and Method's Accuracy

First the convergence of SC options to the American option is investigated. Results are compared with the rate of convergence of Bermudan options using the LSLS method (Longstaff and Schwartz (2001) [176]). Asset values are chosen to be $S_0 =$

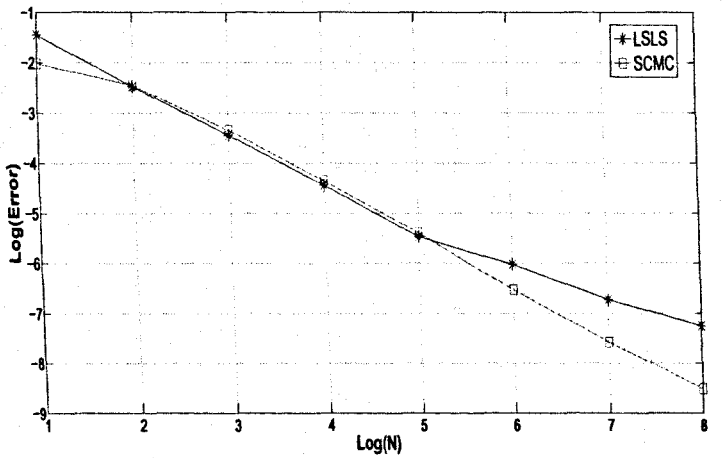
$\{90, 95, 100\}$. For each individual SC put and Bermudan put options, the put-tau control variate is used both for continuation values and for option values. Both methods use scaled Laguerre basis function where number of basis functions $K = 12$.

Figure 6.8 shows errors $\epsilon = |v^{\infty} - \widehat{v}^{N}|$, where v^{∞} and $\widehat{v}^{N}$ are lattice American put benchmark values and Monte Carlo estimates respectively, against the number of contours (the time step in the LSLS case) in logarithmic scaling. One can see that, for all values of S_0, the SCMC method converges faster than the LSLS method does, although approximately at the same rate.

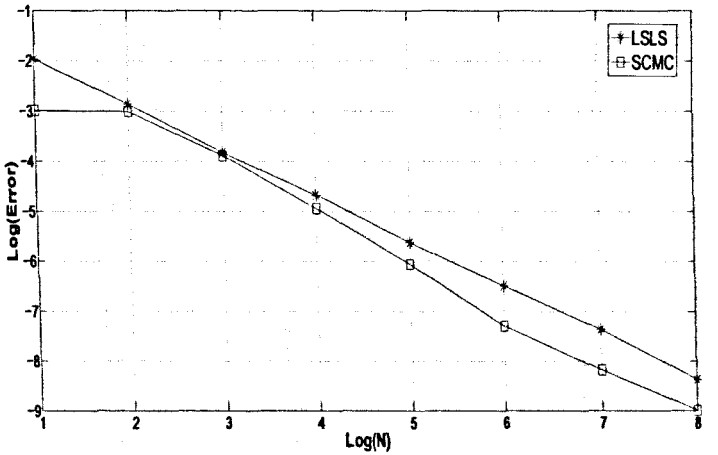
Next American put values computed from Richardson extrapolation are reported. Three-point Richardson extrapolation is performed on SC put options with a number of contours, $N = \{16, 32, 64\}$ respectively. Tables 6.7 and 6.8 report extrapolated American values $\widehat{v}_{t_0}^{\infty,\mathrm{sc}}$, individual sequential contour option values $\widehat{v}^{N}$, standard errors (se), computational times in seconds $[t]$ and biases $\{\mathrm{b}\}$. The tables also report extrapolated American put values from the LSLS method $\widehat{v}_{t_0}^{\infty,\mathrm{ls}}$. Table 6.7 shows extrapolated American put values with pricing control only. Table 6.8 reports values of an extrapolated American put option with both the pricing control and the $v(0)$ control (section 5.4.3). $\widehat{v}^{64}$ is chosen to be the $v(0)$ control variate.

$X, \sigma = 0.2$ Benchmark	LSLS $\widehat{v}_{t_0}^{\infty,\mathrm{ls}}$	SCMC $\widehat{v}_{t_0}^{\infty,\mathrm{sc}}$	SCMC $\widehat{v}_{t_0}^{64}$	SCMC $\widehat{v}_{t_0}^{32}$	SCMC $\widehat{v}_{t_0}^{16}$
95 4.0131	4.0135 (0.0012) [131] {0.30}	4.0148 (0.0012) [136] {1.5}	4.0098 (0.0010)	4.0048 (0.0010)	3.9945 (0.0010)
100 6.0904	6.0893 (0.0016) [187] {-0.65}	6.0905 (0.0016) [189] {0.07}	6.0830 (0.0013)	6.0753 (0.0014)	6.0588 (0.0014)
105 8.7402	8.7409 (0.0020) [242] {0.37}	8.7415 (0.0020) [241] {0.69}	8.7303 (0.0017)	8.7185 (0.0017)	8.6933 (0.0017)
110 11.9728	11.9740 (0.0024) [289] {0.48}	11.9703 (0.0024) [285] {-1.1}	11.9546 (0.0020)	11.9380 (0.0020)	11.9021 (0.0020)

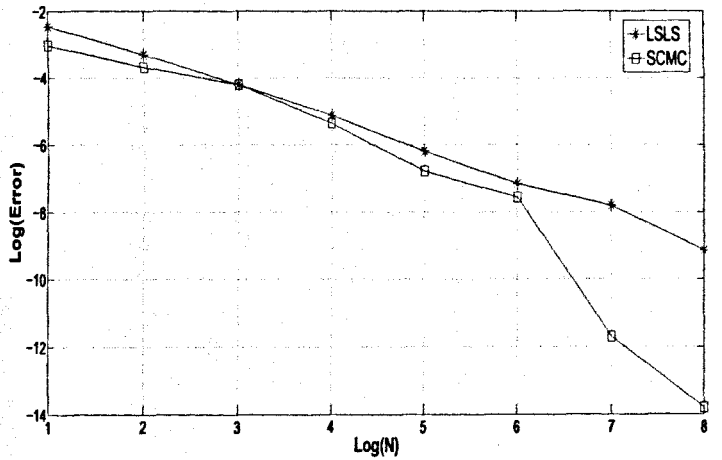
Table 6.7. Extrapolated American put values from SC option values without the $v(0)$ control



(a) $S_0 = 95$



(b) $S_0 = 100$



(c) $S_0 = 105$

Figure 6.8. Convergence of SC put option values to American put values for various S_0

$X, \sigma = 0.2$ Benchmark	LSLS $\widehat{v}_{t_0}^{\infty,\text{ls}}$	SCMC			
		$\widehat{v}_{t_0}^{\infty,\text{sb}}$	$\widehat{v}_{t_0}^{64}$	$\widehat{v}_{t_0}^{32}$	$\widehat{v}_{t_0}^{16}$
95 4.0131	4.01220 (0.00058) [132] {-1.5}	4.01210 (0.00050) [136] {-1.9}	4.0078 (0.0010)	4.0030 (0.0010)	3.9919 (0.0010)
100 6.09037	6.08911 (0.00076) [187] {-1.6}	6.09070 (0.00070) [190] {0.47}	6.0838 (0.0014)	6.0762 (0.0014)	6.0592 (0.0014)
105 8.74017	8.73937 (0.00099) [240] {-0.80}	8.74130 (0.00091) [241] {1.2}	8.7303 (0.0017)	8.7185 (0.0017)	8.6933 (0.0017)
110 11.9728	11.9736 (0.0012) [287] {0.63}	11.9741 (0.0012) [285] {1.1}	11.9583 (0.0020)	11.9415 (0.0020)	11.9052 (0.0020)

Table 6.8. Extrapolated American put values from SC option values with $v(0)$ control

The results in table 6.7 and 6.8 suggest that there is no evidence of bias. This is shown by small bias estimates $|\{b\}| \in 2$. The smallest bias is less then 0.1. The computational time is on average around 200 seconds which is similar to the computational time of the LSLS method.

It is also worth pointing out the reductions in Monte Carlo variance from using the $v(0)$ control on the extrapolated values as in (5.4.19) in table 6.8 over table 6.7. In fact, the variance reduction factor, as defined in (3.3.2), from using the $v(0)$ control is $\sim$5.

6.4.3.2 Variance reduction

This section investigates variance reductions from using put-tau and rebate option control variates. The variance reduction factors are computed by comparing the extrapolated American values from the SCMC method with those computed from the LSLS method with the put-tau rollback and pricing control variate.

Table 6.9 reports a correlation between $v(0)$ control variate in section 5.4.3 with the extrapolated American put value. One can see that the more exercise opportunities the option has, the higher the correlation between it and extrapolated

American put option value. This is true for both SC and Bermudan put options. Correlations between individual SC put options and an American put option are all slightly higher than those computed from individual Bermudan put options and an American put option, although not significantly.

Methods	$\rho\left(\widehat{v}^{\infty}, \widehat{v}^{64}\right)$	$\rho\left(\widehat{v}^{\infty}, \widehat{v}^{32}\right)$	$\rho\left(\widehat{v}^{\infty}, \widehat{v}^{16}\right)$
SBMC	0.907	0.749	0.785
LSLS	0.881	0.701	0.740

Table 6.9. Correlations between extrapolated American put and $v(0)$ controls, $\rho\left(\widehat{v}^{\infty}, \widehat{v}^{N}\right)$

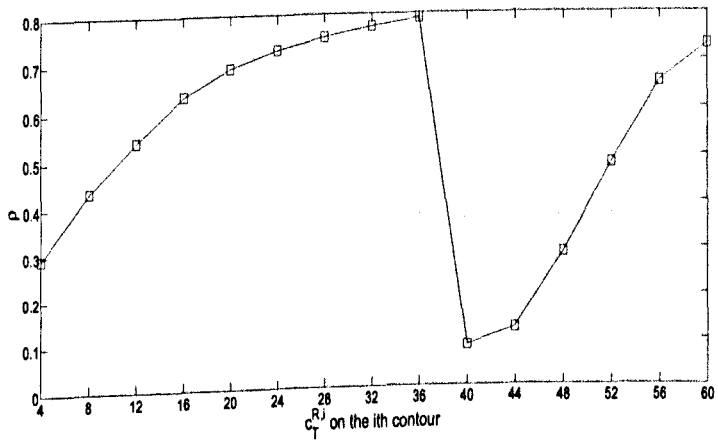
Figure 6.9 shows the plot of correlations between rebate option control variates, $c_T^{R,i}$, on each contour $\beta\left(t \mid \alpha^i\right)$ with SC option values with 64 exercise dates, $\widehat{v}^{64}$. Their covariances and standard deviations are also shown. The number of sample paths is $M = 5,000$.

Panel (a) in figure 6.9 shows that the correlation increases as later contours are used as an option barrier for the rebate option control variate. However, the correlation significantly drops when moving the 36th contour to the 40th contour.

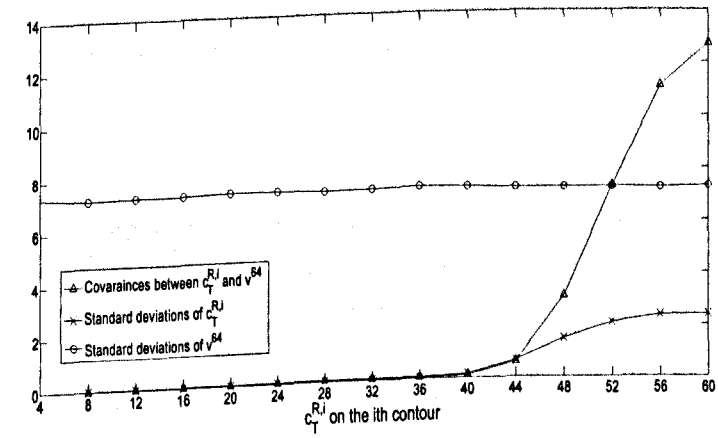
To see this, consider panel (b) where covariances between $c_T^{R,i}$ and $\widehat{v}^{64}$ and their standard deviations are shown. Since each value is taken from different runs, standard deviations of $\widehat{v}^{64}$ are consistent across all contours. $\mathrm{Cov}\left(c_T^{R,i}, \widehat{v}^{64}\right)$ and standard deviations of $c_T^{R,i}$ seem to increase significantly at later contours. An increase in standard deviations of $c_T^{R,i}$ comes from the fact that simulated hitting times have values clustering along the initial segment of a set of contours. In figure 6.5, one can see that hitting times are very close to each other on the first few contours. This causes the rebate option value to have very low standard errors. As hitting times spread out across the (later) contour, standard deviations of rebate option values increase.

Recall that $\rho_{c_T^{R,i}, \widehat{v}^{64}} = \mathrm{Cov}\left(c_T^{R,i}, \widehat{v}^{64}\right) / \left(\sigma_{c_T^{R,i}} \sigma_{\widehat{v}^{64}}\right)$. Since standard deviations of $\widehat{v}^{64}$ seems to be consistent, the correlation between them will depend mainly on their covariance and standard deviation of $c_T^{R,i}$. This relationship can be explained through their relative rate of change:

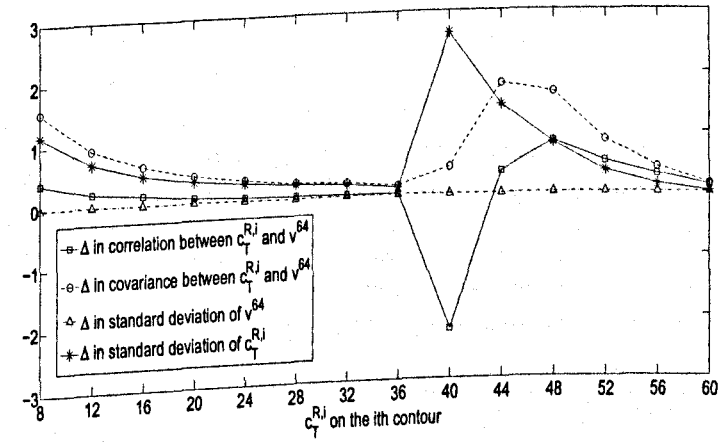
$$\xi = \frac{\Delta \mathrm{Cov}\left(c_T^{R,i}, \widehat{v}^{64}\right)}{\Delta \sigma_{c_T^{R,i}}}, \tag{6.4.1}$$



Panel (a) Correlations between $c_T^{R,i}$ with $\widehat{v}^{64}$ for different contour $\beta^i(t)$.



Panel (b) Covarainces between $c_T^{R,i}$ and $\widehat{v}^{64}$ and their standard deviations.



Panel (c) Changes in correlations between $c_T^{R,i}$ and $\widehat{v}^{64}$, in their covariances, and in their standard deviations.

Figure 6.9. Correlations between $c_T^{R,i}$ with $\widehat{v}^{64}$, their covariances, and standard deviations for different contours $\beta^i(t)$

where $\Delta x = (x_i - x_{i-1})/x_{i-1}$. At the ith contour, one has

$$\xi_i > 1 \rightarrow \rho_i > \rho_{i-1}, \tag{6.4.2}$$

$$\xi_i < 1 \rightarrow \rho_i < \rho_{i-1}. \tag{6.4.3}$$

This means that the correlation decreases if a rate of change in covariance is less than a rate of change in standard deviation of $c_T^{R,i}$ ($\xi_i < 1$). In this case, $\xi_{40} < 1$ which causes a decrease in the correlation from 36th to 40th contour. Panel (c) shows Δ of correlations between $c_T^{R,i}$ and $\widehat{v}^{64}$, of their covariances and their standard deviations for each contour. One can see that from the 36th to the 40th contour, the change in standard deviation of $c_T^{R,i}$ exceeds the change in standard deviation of covariance between $c_T^{R,i}$ and $\widehat{v}^{64}$. In particular, the 40th contour is the only contour that a rate of change in standard deviation of $c_T^{R,i}$ exceeds a rate of change in the covariance between $c_T^{R,i}$ and $\widehat{v}^{64}$. This causes a decrease in correlation from the 36th to the 40th contour as shown in panel (a) and (c).

Next, the effect of rebate option control variate, $c_T^{R,i}$, as an addition to the put-tau, c_τ^p, control variate is illustrated. This is reported in tables 6.10 and 6.11. Note that since American option values are extrapolated from SC option values with 64, 32 and 16 exercise opportunities, the index of the contour for the rebate option selected to be the control variate is chosen to be a multiple of 4.

Table 6.10 reports extrapolated American put values without the use of $v(0)$ control in section 5.4.3. The table reports gains from using only the put-tau control SCMC (c_τ^p) and the put-tau and rebate option control SCMC $\left(c_\tau^p + c_T^{R,i}\right)$. The values from the LSLS method with the put-tau control variate are reported in column 2.

There is no evidence of bias for option values in table 6.10. All biases are within the range ± 2. For the at-the-money and in-the-money case, adding the rebate option control variate seems to make a very marginal improvement on the efficiency gain. The maximum is only around $\sim$1.2. These marginal gains are consistent across all values of the control variate contour index i. For the out-of-the-money case, the slight improvement vanishes and the LSLS method seems to perform better in some cases.

Table 6.11 reports extrapolated American put values with the use of $v(0)$ control in section 5.4.3. The $v(0)$ control is $\widehat{v}^{64}$ due to its highest correlation with extrapolated American option values.

The option values in table 6.11 show no evidence of bias. Since $\rho\left(\widehat{v}^{\infty}, \widehat{v}^{64}\right)$ in the sequential contour case is slightly greater than that of the LSLS case (table 6.9),

X	LSLS	SCMC	SCMC $(c_\tau^p + c_T^{R,i}), i$							
Benchmark	c_τ^p	c_τ^p	60	56	52	48	44	40	36	32
95	4.0135	4.0148	4.0145	4.0144	4.0133	4.0132	4.0127	4.0111	4.0139	4.0117
4.01309	(0.0012)	(0.0012)	(0.0011)	(0.0011)	(0.0012)	(0.0012)	(0.0012)	(0.0012)	(0.0012)	(0.0012)
	[131]	[136]	[137]	[137]	[136]	[137]	[138]	[137]	[137]	[139]
	{0.30}	{1.5}	{1.3}	{1.2}	{0.16}	{0.12}	{-0.37}	{-1.7}	{0.68}	{-1.1}
	1.0	**0.95**	**1.1**	**1.1**	**1.0**	**0.99**	**0.98**	**1.0**	**1.0**	**0.96**

SCMC $\left(c_\tau^p + c_T^{R,i}\right), i$						
28	24	20	16	12	8	4
4.0123	4.0137	4.0127	4.0137	4.0120	4.0135	4.0127
(0.0012)	(0.0012)	(0.0012)	(0.0011)	(0.0012)	(0.0012)	(0.0012)
[138]	[138]	[137]	[139]	[137]	[139]	[137]
{-0.71}	{0.50}	{-0.35}	{0.50}	{-0.93}	{0.38}	{-0.34}
0.99	**0.99**	**1.0**	**1.0**	**0.98**	**0.97**	**0.97**

X	LSLS	SCMC	SCMC $\left(c_\tau^p + c_T^{R,i}\right), i$							
Benchmark	c_τ^p	c_τ^p	60	56	52	48	44	40	36	32
100	6.0893	6.0905	6.0918	6.0894	6.0906	6.0897	6.0902	6.0884	6.0889	6.0899
6.09037	(0.0016)	(0.0016)	(0.0015)	(0.0015)	(0.0015)	(0.0015)	(0.0016)	(0.0016)	(0.0015)	(0.0015)
	[187]	[189]	[189]	[189]	[189]	[188]	[188]	[188]	[188]	[189]
	{-0.65}	{0.07}	{0.94}	{-0.62}	{0.14}	{-0.47}	{-0.10}	{-1.3}	{-0.94}	{-0.30}
	1.0	**1.1**	**1.2**	**1.1**	**1.1**	**1.1**	**1.1**	**1.1**	**1.1**	**1.1**

SCMC $\left(c_\tau^p + c_T^{R,i}\right)), i$						
28	24	20	16	12	8	4
6.0910	6.0910	6.0919	6.0888	6.0906	6.0902	6.0896
(0.0016)	(0.0016)	(0.0015)	(0.0015)	(0.0015)	(0.0015)	(0.0016)
[189]	[189]	[188]	[189]	[189]	[190]	[189]
{0.40}	{0.41}	{0.96}	{-1.0}	{0.16}	{-0.14}	{-0.50}
1.07	**1.06**	**1.10**	**1.12**	**1.11**	**1.09**	**1.07**

X	LSLS	SCMC	SCMC $\left(c_\tau^p + c_T^{R,i}\right), i$							
Benchmark	c_τ^p	c_τ^p	60	56	52	48	44	40	36	32
105	8.7409	8.7415	8.7411	8.7407	8.7399	8.7415	8.7386	8.7394	8.7407	8.7404
8.74017	(0.0020)	(0.0020)	(0.0019)	(0.0019)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)	(0.0020)
	[242]	[241]	[244]	[243]	[241]	[242]	[243]	[242]	[242]	[243]
	{0.37}	{0.69}	{0.48}	{0.29}	{-0.16}	{0.65}	{0.81}	{-0.39}	{0.28}	{0.14}
	1.0	**1.0**	**1.19**	**1.13**	**1.08**	**1.03**	**1.04**	**1.04**	**1.05**	**1.03**

SCMC $\left(c_\tau^p + c_T^{R,i}\right), i$						
28	24	20	16	12	8	4
8.7412	8.7423	8.7421	8.7405	8.7392	8.7418	8.7405
(0.0020)	(0.0020)	(0.0020)	(0.0019)	(0.0019)	(0.0019)	(0.0019)
[242]	[242]	[242]	[243]	[242]	[242]	[242]
{0.50}	{1.1}	{0.98}	{0.15}	{-0.52}	{0.84}	{0.17}
1.01	**1.04**	**1.07**	**1.09**	**1.11**	**1.12**	**1.09**

Table 6.10. Extrapolated American put values from the sequential contour options with only pricing control

X Benchmark	LSLS c_τ^p	SCMC c_τ^p	SCMC $(c_\tau^p + c_T^{R,i}), i$							
			60	56	52	48	44	40	36	32
95	4.01221	4.01210	4.01291	4.01233	4.01345	4.01256	4.01235	4.01235	4.01251	4.01318
4.01309	(0.00058)	(0.00050)	(0.00048)	(0.00048)	(0.00049)	(0.00050)	(0.00049)	(0.00051)	(0.00050)	(0.00050)
	[132]	[136]	[138]	[138]	[137]	[138]	[139]	[138]	[138]	[139]
	{-1.5}	{-1.9}	{-0.37}	{-1.6}	{0.74}	{-1.1}	{-1.5}	{-1.5}	{-1.2}	{0.18}
	1.00	**1.3**	**2.8**	**2.8**	**2.8**	**2.6**	**2.7**	**2.6**	**2.6**	**2.6**

SCMC $(c_\tau^p + c_T^{R,i})), i$						
28	24	20	16	12	8	4
4.0135	4.01228	4.01359	4.01318	4.01312	4.01271	4.01294
(0.00054)	(0.00051)	(0.00051)	(0.00050)	(0.00048)	(0.00050)	(0.00049)
[138]	[140]	[138]	[140]	[138]	[139]	[138]
{0.75}	{-1.6}	{0.97}	{0.18}	{0.06}	{-0.76}	{-0.31}
2.3	**2.5**	**2.5**	**2.7**	**2.8**	**2.7**	**2.8**

X Benchmark	LSLS c_τ^p	SCMC c_τ^p	SCMC $(c_\tau^p + c_T^{R,i}), i$							
			60	56	52	48	44	40	36	32
100	6.08914	6.08910	6.09028	6.09069	6.09166	6.09047	6.08957	6.09065	6.09080	6.09023
6.09037	(0.00076)	(0.00070)	(0.00070)	(0.00068)	(0.00071)	(0.00070)	(0.00069)	(0.00069)	(0.00069)	(0.00071)
	[187]	[190]	[192]	[191]	[191]	[191]	[189]	[189]	[189]	[191]
	{-1.6}	{0.47}	{-0.13}	{0.47}	{1.8}	{0.14}	{-1.2}	{0.40}	{0.63}	{-0.20}
	1.00	**1.2**	**1.1**	**1.0**	**1.1**	**1.1**	**1.1**	**1.1**	**1.1**	**1.0**

SCMC $(c_\tau^p + c_T^{R,i}), i$						
28	24	20	16	12	8	4
6.09046	6.08944	6.0896	6.09152	6.09047	6.09115	6.09077
(0.00072)	(0.00070)	(0.00072)	(0.00071)	(0.00072)	(0.00069)	(0.00071)
[190]	[191]	[190]	[192]	[190]	[192]	[191]
{0.13}	{-1.3}	{-1.1}	{1.6}	{0.14}	{1.1}	{0.56}
1.0	**1.1**	**1.0**	**1.1**	**1.0**	**1.1**	**1.0**

X Benchmark	LSLS c_τ^p	SCMC c_τ^p	SCMC $(c_\tau^p + c_T^{R,i}), i$							
			60	56	52	48	44	40	36	32
105	8.73937	8.74130	8.73898	8.74073	8.73986	8.74164	8.73898	8.74039	8.73880	8.74131
8.74017	(0.00099)	(0.00091)	(0.00091)	(0.00091)	(0.00091)	(0.00091)	(0.00094)	(0.00092)	(0.00089)	(0.00091)
	[245]	[246]	[246]	[246]	[245]	[245]	[245]	[246]	[245]	[246]
	{-0.80}	{1.2}	{-1.3}	{0.62}	{-0.34}	{1.6 }	{-1.3}	{0.24}	{-1.5}	{1.3}
	1.0	**1.0**	**1.2**	**1.2**	**1.2**	**1.2**	**1.1**	**1.1**	**1.2**	**1.2**

SCMC $(c_\tau^p + c_T^{R,i}), i$						
28	24	20	16	12	8	4
8.74149	8.74092	8.73882	8.7416	8.74138	8.74014	8.74001
(0.00094)	(0.00097)	(0.00097)	(0.00096)	(0.00097)	(0.00093)	(0.00095)
[245]	[246]	[245]	[245]	[245]	[245]	[245]
{1.4}	{0.77}	{-1.4}	{1.5}	{1.3}	{-0.03}	{-0.17}
1.1	**1.0**	**1.1**	**1.1**	**1.0**	**1.1**	**1.1**

Table 6.11. Extrapolated American put values from the sequential contour options with pricing and $v(0)$ control

one expects the standard errors from the SCMC method to be slightly lower than those from the LSLS method. This is confirmed by results in table 6.11. For the out-of-the-money case, gains are increasing approximately to 3. This level of gain is also consistent across all contour indexes i. However, for the in-the-money and the at-the-money cases, although the SCMC produces a lower standard error, variance reductions are still very marginal.

The rebate option can add only marginal improvements because it is used as an additional control to the put-tau control which has a very high correlation with $\widehat{v}^N, N = 64, 32$ and 16. Hence, the contribution from the rebate option control is overshadowed by that from the put-tau control variate.

6.5 Conclusion

This chapter has described a new simulation method, the SCMC method, to value the American put option. The method generalises the existing LSLS method by using a more general family of contours. The method generates hitting times to a set of exponential contours and approximates the value of the option which has random exercise opportunities. This type of option is called the sequential contour (SC) option. The American put value is estimated by extrapolating these SC option values.

Results suggest that, with a suitable choice of contour parameters, biases in American put option values found by the method are negligible. Also, standard SC option values converge to the American option value slightly faster than standard Bermudan option values do.

The method also provides a set of barrier options that can be used as control variates for individual SC option values. Numerical examples illustrate the use of a rebate option (with exponential barrier) as an additional control variate to the put-tau control. Numerical results in section 6.4.3.2 suggest that the SCMC, when using the combination of the rebate option control variate, the put-tau control variate, and the $v(0)$ control, can achieve gains of $\sim$3 when valuing out-of-the-money options. However, for the at-the-money and the in-the-money options it turns out that the efficiency gains achieved from adding the rebate option control variate is marginal. This may be because the effect of rebate option is overshadowed by the put-tau control which is highly correlated with the SC option.

All in all, the new method can value American put option without biases but gains over the standard LSLS method are not substantial.

Part III

Projection Techniques and Exotic American Options

Chapter 7

Valuing Exotic American Options using the Sequential Contour Monte Carlo Method

This chapter illustrates the use of the sequential contour Monte Carlo (SCMC) method to price exotic American options; an American fractional call option and a linear knock-in barrier American call option with a fractional power payoff. The proposed method combines the contour bridge method (chapter 3) and the SCMC method.

This chapter also demonstrates how the contour bridge sampling method (section 3.2.2) can be combined with the sequential contour method to generate sample paths backwards in time.

Different projection techniques that can be used to approximate continuation values are proposed. Instead of using asset values to approximate option's continuation values, the hitting time to a predetermined set of contours ($\mathcal{T}$-projection) and the distance along contours ($\mathcal{D}$-projection) can be used. These projection techniques are applied to value standard American put options, American fractional power call options and linear knock-in American barrier options.

The chapter is structured as follows. Section 7.1 discusses the use of the contour bridge sampling method with the SCMC method. Section 7.2 introduces and explores the different projection techniques. Section 7.3 applies the method to price American fractional power call options. The contour construction in the call option case is discussed and differences in contour shape between the call option and the put option cases are pointed out. An application to value linear knock-in American barrier options with a fractional power payoff is discussed in section 7.4.

Numerical results are reported in section 7.5. Section 7.6 concludes.

7.1 Sequential contour bridge (SCB) method

Consider exponential contours of the form

$$\beta(t \mid \alpha^i) = \frac{1}{\alpha^i} \exp(g^i t), \tag{7.1.1}$$

where $\frac{1}{\alpha^i}$ is the time t_0 value of the ith contour, g^i is the contour growth rate, and i is a contour index, $i = 1, \ldots, N$. Write $\beta^i(t) = \beta(t \mid \alpha^i)$. The idea of the SCB method is to generate sample paths backwards in time from contour $\beta^{i+1}(t)$ to contour $\beta^i(t)$ and also apply the generalised LSLS method (section 6.1.3) at the same time. It is equivalent to the Brownian bridge method for the vertical contours case (section 3.1.2). This method is in contrast to the forward path sampling method used in the SCMC method in chapter 6 in which one stores an entire set of hitting times and asset values, and then works backwards from each contour to the previous contour.[1]

Chaudhary (2005) [63] applied the Brownian bridge method with vertical contours (standard LSLS) with quasi Monte Carlo simulation. The main advantage of the backward simulation method is the reduction in memory requirement from $\mathcal{O}(M \times N \times q)$ to $\mathcal{O}(M \times q)$ where M is a total number of sample paths, N is a total number of contours and q is a total number of assets. This lower memory requirement facilitates the valuation of options in higher dimensioned.[2] Also, if one has to store all sample paths, one will get severe restrictions on M.

7.1.1 Contour construction for the SCB method

To implement the sequential contour bridge method, one needs to be able to sample from the hitting time bridge density given by (3.2.9). However, to do this, growth rates g between the two bounding contours must be equal (see section 3.2.2). This implies that, to implement the method, every contour's growth rate must be the same. Write g for this common value of growth rate. Recall the definition of t_i^* from chapter 6. It is the time on the ith contour such that

$$\beta^i(t_i^*) = X. \tag{7.1.2}$$

[1]The only difference between the SCMC and the SCB methods is the way sample paths are generated.

[2]Multiple asset American options are not considered in this thesis.

Using (7.1.1) and write $\beta^i(0) = \frac{1}{\alpha^i}$, one obtains

$$t_i^* = \frac{1}{g} \ln \left(\frac{X}{\beta^i(0)} \right). \tag{7.1.3}$$

(7.1.3) is used to approximate where contours intersect the option's EEB. In chapter 6, the growth rate is defined to be a function such that values of t_i^* can be chosen for each contour (type 4 contour). However, since g must be fixed in this case, the type 4 contour, which g is varied, cannot be applied. To be able to control t_i^* with g fixed, α^i must be varied and chosen according to values of t_i^*. This contour construction is referred to as the type 5 contour.

Type 5 contour: α^i varied as a function of t_i^* and g fixed

The overview of type 5 contour construction is as follows.

1. Fix g.
2. Predetermine $t^* = \{t_i^*\}_{i=1,\ldots,N-1}$.
3. Choose $\alpha = \{\alpha^i\}_{i=1,\ldots,N-1}$ according to t^*.

These steps are described now. First, for a SC put option, set $g = \ln \left(\frac{X}{\beta^{N-1}(0)} \right) \frac{1}{T}$. Then, to control values of t_i^*, one substitutes a function g in (6.2.12) into (7.1.3) to obtain

$$t_i^* = \frac{1}{g} \ln \left(\frac{X}{C} \right) \left(\frac{i}{N-1} \right)^{k(i)}, \tag{7.1.4}$$

where $C \in \mathbb{R}^+$ is a constant, $k(i)$ is defined in (6.2.10) and N is a total number of contours. With this set of $t^* = \{t_i^*\}_{i=1,\ldots,N-1}$, the contours can be constructed with a set of parameter $\alpha = \{\alpha^i\}_{i=1,\ldots,N-1}$ as follows

$$\alpha^i = \frac{\exp(g t_i^*)}{X}. \tag{7.1.5}$$

Since there is a function $k(i)$ in (7.1.4), $k_{\max}$ and $k_{\min}$, which are used for a type 4 contour, are also parameters for a type 5 contour. The other additional parameter is a constant C. If one wants to have an exact same set of t^* like the one used for a type 4 contour in chapter 6, one will set C to be equal to a fixed $\beta(0) = 1/\alpha$ for a type 4 contour.

7.1.2 The SCB algorithm

Let τ_j^i be the first hitting time to the ith contour β^i on the jth path as defined in (6.1.1) and $S_j^i = S_{\tau_j^i}$ be the asset value at the ith contour on the jth path. Write $S_0 = S^0$ and let $h_j^i \equiv \left(X - S_j^i\right)^+$ be the payoff of a put option. The algorithm proceeds as follows.

The SCB iteration algorithm:
For path $j = 1, \ldots, M$,

1. Construct a hitting time $\tau_j^1 \mid \tau^0, S_0$ to the first contour $\beta^1(t)$.
2. Conditional on τ_j^1, construct $\tau_j^{N-1} \mid \tau_j^1$ to β^{N-1}.
3. If $\left(\tau_j^{N-1} \mid \tau_j^1\right) < T$, set $S_j^i = \frac{1}{\alpha^{N-1}} e^{g\tau_j^{N-1}}$ and compute $S_j^N \mid S_j^{N-1}$ using (6.1.10).[3] Set $i = N$ and apply step 4.
4. Iterate back from the contour $\beta^i(t)$ to $\beta^{i-1}(t)$ using the generalised LSLS algorithm (section 6.1.3) to obtain SC option values $\widehat{v}_j^i$. Decrement $i \leftarrow i - 1$ and apply step 5.
5. At the β^i contour, sample conditional hitting times $\tau_j^{i-1} \mid \tau_j^i, \tau_j^1$ to β^{i-1} by using the contour bridge sampling method discussed in section 3.2.2. Go back to step 4.
6. Keep repeating step 4 and 5 until the contour β^0, where $\tau^0 = 0$, is reached. Then set $\widehat{v}^0 = \max\left(\frac{1}{M}\sum_{j=1}^M e^{-r\tau_j^1}\widehat{v}_j^1, h^0\right)$.

7.2 Different projection techniques

Write $\mathcal{P}^i = \left\{\left(\tau_j^i, S_j^i\right)\right\}_{j=1,\ldots,M}$. This is a set of simulated points on the ith contour. Let

$$\pi : \mathcal{P}^i \longmapsto \mathbb{R}^+. \tag{7.2.1}$$

be a projection operator for $p = (\tau, S) \in \mathcal{P}^i$, one could set $\pi = \pi^S$ or $\pi = \pi^\tau$ where

$$\pi^S(\tau, S) = \pi^S(p) = S \tag{7.2.2}$$

$$\pi^\tau(\tau, S) = \pi^\tau(p) = \tau \tag{7.2.3}$$

[3]Note that since $\beta^{N-1}(T) = X$ and one is valuing a put option, values of S_j^{N-1} where $\tau_j^{N-1} > T$ will not be used and hence are not generated.

(7.2.2) is referred to as the $\mathcal{S}$-projection and (7.2.3) is referred to as the $\mathcal{T}$-projection.

In the standard LSLS method (section 4.3.4), one uses asset values, $\left\{S_j^i\right\}_{j=1,\ldots,M}$ at each time step t_i, with the chosen basis functions to approximate option continuation values, $\widetilde{C}_j^i$. This is the $\mathcal{S}$-projection in which continuation values are mapped vertically.

The SCB and the SCMC method extends the choices for the projection used in the continuation values regression. It does not restrict one to use only π^S to specify the independent variables in the regression. In particular, because contours are not vertical, time and distance along contours can now be defined and can be used as the independent variables in the regression. $\mathcal{T}$-projection defines time and $\mathcal{D}$-projection defines distance on contours. The $\mathcal{D}$-projection technique is discussed in section 7.2.3.

For the standard LSLS method which corresponds to simulating hitting times to vertical contours, there exists no $\mathcal{T}$-projection since, on the ith contour (time step), all asset values are observed at the same fixed time.

The different projection techniques are illustrated in figure 7.1. It shows a contour $\beta\left(t \mid \alpha^i\right)$ and a simulated point (τ, S). The vertical distance from 0 to S is defined by the $\mathcal{S}$-projection. The horizon distance from 0 to τ is defined by the $\mathcal{T}$-projection, and the distance along the contour from 0 to (τ, S) is defined by the $\mathcal{D}$-projection.

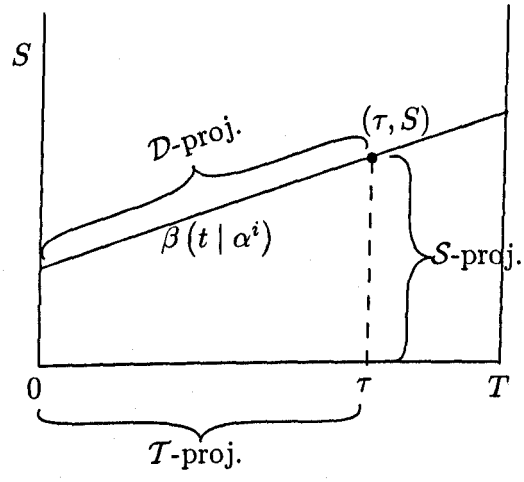


Figure 7.1. Illustration of different projection techniques

7.2.1 Generalised LSLS Algorithm with projection operator

This section provides a generalised LSLS algorithm to value the SC option in terms of the projection operator $\pi(\tau, S)$. At this stage, the method is general in a sense that the type of projection is not yet specified. Then $\mathcal{T}$-projection and $\mathcal{D}$-projection will be described later in detail.

Define the option's continuation value function to be $C \longmapsto \mathbb{R}^+$. The projection is used to approximate option's continuation values by $C \sim \widetilde{C} = f(\pi(\tau, S)|a)$, where $\pi(\tau, S)$ is defined in (7.2.2) and (7.2.3) and a is the parameter to be found.

Write

$$\hat{\beta}^i = \left\{\left(\tau, \beta^i\left(\tau \mid \alpha^i\right)\right)\right\}_{\tau\in[0,T]}, \tag{7.2.4}$$

$i = 1, \ldots, N$ for the set of contours. The projection operator restricts to a map $\pi : \hat{\beta} \longmapsto \mathbb{R}^+$. Let $h_j^i \equiv h\left(S_j^i\right)$ be the SC option's payoff at the ith contour on the jth path. For paths $j = 1, \ldots, M$, the algorithm proceeds as follows.

The algorithm with projection operator:

1. Sample S_j^N at the contour β^N where time $\tau_N = T$. This can be done either by using forward-evolution (section 6.1.2) or using backward-evolution (section 7.1). Set $\widehat{v}_j^N = h_j^N$. Set $i = N - 1$ and apply step 2.

2. At the ith contour, write $\widehat{C}_j^i$ for the continuation value, $\widehat{C}_j^i = \mathrm{e}^{-r(\tau^{i+1}-\tau^i)}v_j^{i+1}$.

3. Write $\widetilde{C}_j^i$ for approximated continuation values. Find a set of parameters $a^i = \left\{a_k^i\right\}_{k=1,\ldots,K}$ such that $\widehat{C}_j^i \sim \widetilde{C}_j^i = f\left(\pi\left(p_j^i\right) \mid a^i\right)$, where

$$f\left(\pi\left(p_j^i\right) \mid a^i\right) = \sum_{k=1}^{K} a_k^i \psi_k\left(\pi\left(p_j^i\right)\right), \tag{7.2.5}$$

   for a chosen set of basis functions $\{\psi_k\}_{k=1,\ldots,K}$. Parameters a^i can be obtained by

$$a^i = \arg\min_a \left\| \widehat{C}_j^i - f\left(\pi\left(p_j^i\right) \mid a^i\right) \right\|. \tag{7.2.6}$$

4. Exercise the option if $h_j^i > \widetilde{C}_j^i$. Then the option values along the ith contour are

$$\widehat{v}_j^i = \begin{cases} h_j^i, & \text{if exercised,} \\ \mathrm{e}^{-r(\tau^{i+1}-\tau^i)}\widehat{v}_j^{i+1}, & \text{otherwise.} \end{cases} \tag{7.2.7}$$

Decrement $i \leftarrow i-1$ and apply step 2.

5. At the contour β^0 where $\tau^0 = 0$, the SC option value is $\widehat{v}^0 = \frac{1}{M}\sum_{j=1}^{M} \max\left(\mathrm{e}^{-r\tau_j^1} v_j^1, h^0\right)$.

7.2.2 Hitting times projection ($\mathcal{T}$-projection)

At the ith contour, one has a set of asset values, $S^i = \left\{S_j^i\right\}_{j=1,\ldots,M}$, and a set of hitting time, $\tau^i = \left\{\tau_j^i\right\}_{j=1,\ldots,M}$. Instead of mapping

$$S_j^i = \beta_{\alpha^i} : \tau_j^i \longmapsto \frac{1}{\alpha^i}\mathrm{e}^{g\tau_j^i}, \tag{7.2.8}$$

and regressing option values against the values of S_j^i, one can have a map

$$\tau_j^i = \beta_{\alpha^i}^{-1} : S_j^i \longmapsto \frac{\ln\left(S_j^i \alpha^i\right)}{g}, \tag{7.2.9}$$

and regress option values against the hitting times τ_j^i to estimate option continuation values along each contour.[4]

Since only in-the-money paths will contribute to the exercise decision, one needs to determine in-the-money paths. To do this, one makes use of predetermined values of t_i^* from (7.1.4). For a put option, $\tau_j^i < t_i^*$ implies $S_j^j < X$ which is an in-the-money value. Therefore, when valuing the American put option, $\tau_j^i < t_i^*$ is the in-the-money value while $\tau_j^i > t_i^*$ is the out-of-the-money value.

The algorithm proceeds as specified in section 7.2.1. To implement $\mathcal{T}$-projection technique, one sets

$$\pi^{\tau}\left(p_j^i\right) = \tau_j^i \tag{7.2.10}$$

in (7.2.5).

To obtain the EEB, let $\mathbb{T}_i^E = \left\{\tau_j^i \mid h_j^i > \widetilde{C}_j^i\right\}$, which is the set of τ^i at which a holder exercises the option. Then, for a put option, set

$$\tau_i^E = \max_{\tau}\left\{\tau \in \mathbb{T}_i^E\right\}. \tag{7.2.11}$$

τ_i^E is the greatest simulated hitting time on the ith contour that the option is exercised. The set $\left\{\tau_i^E\right\}_{i=1,\ldots,N}$ represents the EEB of the option. Then the option

[4]Since the SCMC method samples hitting times, when using the $\mathcal{T}$-projection, equation (7.2.9) does not have to be computed.

value $\widehat{v}_j^i$ in step 4 on page 161 can be found by

$$\widehat{v}_j^i = \begin{cases} h_j^i, & \tau_j^i \leq \tau_i^E, \\ \mathrm{e}^{-r(\tau_j^{i+1}-\tau_j^i)}\widehat{v}_j^{i+1}, & \text{otherwise.} \end{cases} \tag{7.2.12}$$

7.2.3 Contour distance projection ($\mathcal{D}$-projection)

Instead of projecting horizontally ($\mathcal{S}$-projection) or vertically ($\mathcal{T}$-projection), a third possibility is to project onto a distance along a contour itself. Let $d^i\left(p^1, p^2\right)$ be a distance between points p^1 and p^2 along $\hat{\beta}^i$. The idea is to regress $\widehat{C}^i$ against $d^i\left(p^1, p^2\right)$, $p^1, p^2 \in \mathcal{P}$. $d^i\left(p^1, p^2\right)$ can be explicitly computed.

Write $d^i\left(p^1, p^2\right) = d(\tau^1, \tau^2)$ where $p^1 = \left(\tau^1, S^1\right)$ and $p^2 = \left(\tau^2, S^2\right)$, then $d(\tau^1, \tau^2)$ is given by (see appendix B)

$$\begin{aligned} d(\tau^1, \tau^2) \;=\; & \frac{1}{g}\Bigg\{\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau^2)} - \sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau^1)} \\ & + g(\tau^2-\tau^1) - \ln\left(1+\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau^2)}\right) \\ & + \ln\left(1+\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau^1)}\right)\Bigg\}, \end{aligned} \tag{7.2.13}$$

where g is the growth rate in (7.1.1). When implementing, set $\tau^2 = \tau_j^i$ and $\tau^1 = 0$.

To determine the in-the-money value for the $\mathcal{D}$-projection, one computes $d(0, t_i^*)$ where t_i^* is defined in (7.1.4). For each $j = 1, \ldots, M$, one determines $d(0, \tau_j^i)$. For a put option, an in-the-money value for a put option is determined by the set $\{d : d(0, \tau_j^i) \in (0, d(0, t_i^*))\}$ and an out-of-the-money value is determined by the set $\{d : d(0, \tau_j^i) \in [d(0, t_i^*), \infty)\}$. This is because $d(0, \tau_j^i) < d(0, t_i^*)$ implies $S_j^i < X$. Note that this procedure is the same as the one described for the $\mathcal{T}$-projection where one compares τ_j^i with t_i^*, but in this case the distance on the contour is used instead.

The algorithm proceeds as illustrated in section 7.2.1. One sets

$$\pi^d\left(p_j^i\right) = d_j^i\left(0, \tau_j^i\right) \tag{7.2.14}$$

in (7.2.5).

The option's EEB can be found as follows. Let $\mathbb{D}_i^E = \left\{d_j^i \mid h_j^i > \widetilde{C}_j^i\right\}$, which

is the set of d^i at which a holder exercises the option. For a put option, set

$$d_i^E = \max_d \left\{ d \in \mathbb{D}_i^E \right\}. \tag{7.2.15}$$

d_i^E is the greatest distance along the ith contour that the option is exercised. The set $\left\{ d_i^E \right\}_{i=1,\ldots,N}$ represents the EEB of the option. The value d_i^E comprises a pair of the asset value S_i^E in (6.1.14) and time τ_i^E in (7.2.11).

In this case, the option value $\widehat{v}_j^i$ in step 4 can be found by

$$\widehat{v}_j^i = \begin{cases} h_j^i, & d_j^i \le d_i^E, \\ \mathrm{e}^{-r(\tau_j^{i+1} - \tau_j^i)} \widehat{v}_j^{i+1}, & \text{otherwise.} \end{cases} \tag{7.2.16}$$

7.3 Application to American fractional power call options

This section illustrates how the sequential contour method with $\mathcal{S}$-projection, $\mathcal{T}$-projection and $\mathcal{D}$-projection can be applied to value an American power call option with a payoff, $h(s)$,

$$h(s) = [\max(s - X, 0)]^\kappa \tag{7.3.1}$$

where $\kappa \in (0, 1)$.

Without dividends, the standard vanilla American call option value is equal to the corresponding European call option since it is not optimal to exercise an American call early. However, this is not the case for an American call with the fractional power payoff in (7.3.1). This is illustrated in Table 7.1.

S_0	American	European
90	1.25443	1.18252
100	2.26822	2.15269
110	3.39444	3.24124

Table 7.1. Comparison of American and European fractional power call options

Table 7.1 shows values of American fractional power call options and those of their European counterparts. Parameter values are $X = 100$, $r = 0.05$, $\sigma = 0.2$, and $T = 1$. $\kappa = 0.5$. All values are computed using a trinomial lattice method with $N = 204,800$ time steps. It is clear that American call values are all higher

than those of European call which means American fractional power call can be (optimally) exercised early.

Figure 7.2 shows the convergence of Bermudan fractional power call option values to American fractional call option values with the same set of parameter values used in table 7.1. These values are computed using a lattice method with $N = 204,800$ time steps with different exercise dates, E. Panel (a) shows Bermudan fractional call values with different E on a log scale, $E = 1, 2, \ldots, 204\,800$. As expected, as E increases the Bermudan value converges to the American value of 2.26822. Panel (b) shows the convergence in errors, $\ln(v_E^c - v^\infty)$, where v_E^c is the Bermudan fractional call option value with E exercise dates and v^∞ is the value of an American fractional call. It shows that the Bermudan value converges to the American value at a rate of ~ 1.

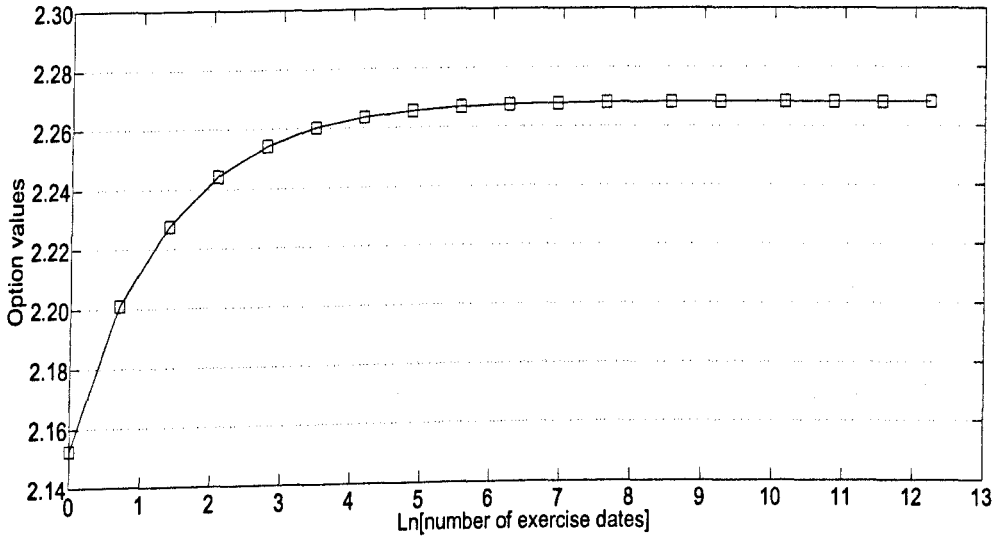
Figure 7.3 plots American fractional power call option's EEB using a lattice method and the same set of parameter values used in table 7.1. The EEB is concave rather than convex for the put case. The EEB is non-increasing in time to maturity of the option. At maturity, the EEB has a value equal to option's strike, X. The region above the EEB is the immediate exercise region while the region below the EEB is the continuation region. Let τ^* be the optimal stopping time of a American fractional call option and let S_t^E be the EEB. Write S_t as an asset value at time t. In this case, τ^* is given by

$$\tau^* = \inf\left\{t > 0 : S_t \geq S_t^E\right\}. \tag{7.3.2}$$

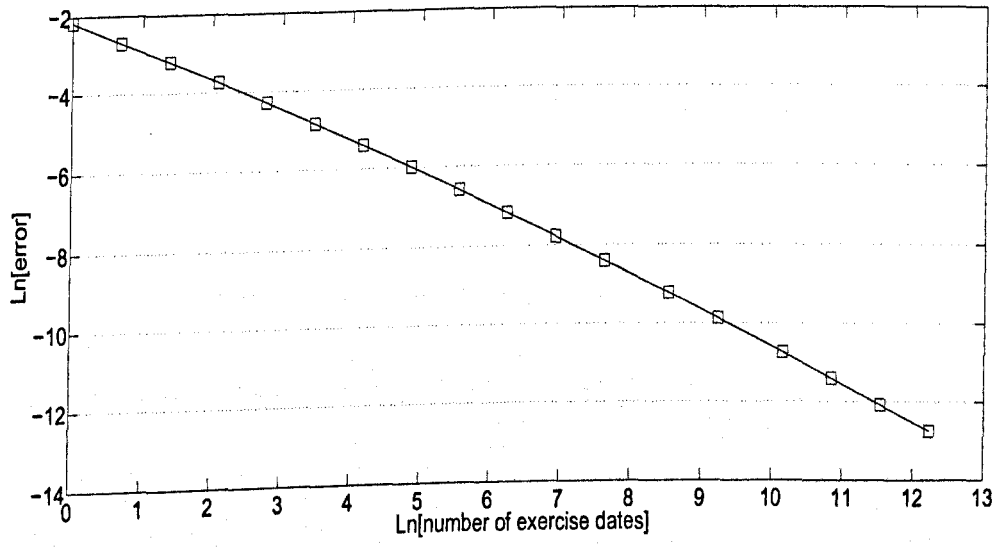
7.3.1 Valuation of an American fractional power call option

The SCMC method with projection operators described in section 7.2.1 can be easily applied to value an American fractional power call option.

Two modifications are needed for the method to value an American fractional power call. The first modification is trivial; the payoff function is now given by (7.3.1). The second modification is concerned with the estimation of the option's EEB. As shown in figure 7.3, the exercise region is above the EEB and the continuation region is below the EEB. Therefore, definitions of S_i^E in (6.1.14), τ_i^E in (7.2.11), and d_i^E in (7.2.15) must be changed. In particular, at the ith contour, they



Panel (a): Convergence of Bermudan fractional call option values to the American value.



Panel (b): Convergence: ln(error) against ln(number of exercise dates).

Figure 7.2. Convergence of Bermudan fractional call option values to the American value from a lattice method

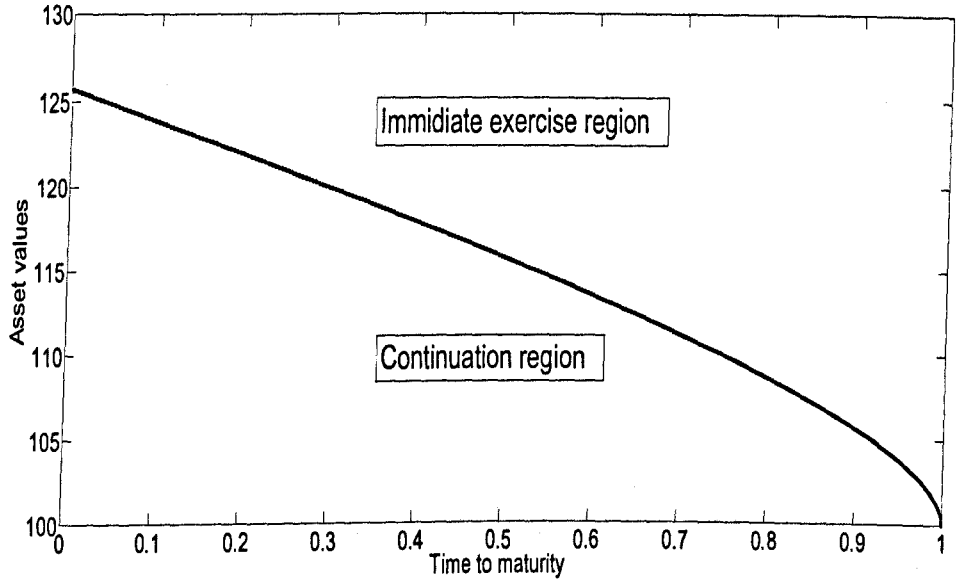


Figure 7.3. American fractional power call options' EEB

are

$$S_i^E = \min_S \{S \in \mathbb{S}_i^E\} \tag{7.3.3}$$

$$\tau_i^E = \min_\tau \{\tau \in \mathbb{T}_i^E\} \tag{7.3.4}$$

$$d_i^E = \min_d \{d \in \mathbb{D}_i^E\}, \tag{7.3.5}$$

$\mathbb{S}_i^E, \mathbb{T}_i^E$, and $\mathbb{D}_i^E$ are sets of asset values, hitting times and contour distances at which a holder exercises the option. S_i^E, τ_i^E, and d_i^E are now the smallest asset value, hitting time, and distance at the ith contour that the option is exercised.

To estimate the value of an American fractional power call options, the method approximates values of individual SC fractional power call options with different exercise dates, N. Then, the value of an American fractional call option is obtained by using Richardson extrapolation (section 6.1.4).

7.3.2 Contour construction for an American fractional power call option

In chapter 6 where American puts are valued, one was able to choose the growth rate g^{N-1} by setting $\beta^{N-1}(T) = X$. For American fractional power call case, it is required that $\beta^{N-1}(T) > X$. This is because, at time T (the vertical contour,

β^N) and for paths $j = 1, \ldots, M$, one wants to have sufficient in-the-money paths $S_j^N > X$. Also, since the EEB of American fractional power call options lie above an option's strike X (see figure 7.3), having $\beta^{N-1}(T) > X$ implies that sample paths on the $(N-1)$th contour contribute to the exercise decision of the option's holder. Therefore, in this case g^{N-1} must be chosen such that $\beta^{N-1}(T) > X$.

It is important that contour parameters must be chosen such that there is no bias in option values and are robust to different values of S_0 with the same X, r, σ and T.

To see the effect of changing $\beta^{N-1}(T)$, table 7.2 reports values of fractional power American call options with different values of $\beta^{N-1}(T)$. Type 5 contour is used with the SCMC method. Except for the cases where $S_0 = 90$ and $\beta^{N-1}(T) \in \{130, 140\}$, contour parameters are $k_{\max} = 0.85$, $k_{\min} = 0.1$, $C = 5$, $g = 11.69$, and $\beta^{N-1}\left(0 \mid \alpha^{N-1}\right) = 0.001$. When $S_0 = 90$ and $\beta^{N-1}(T) \in \{130, 140\}$, $k_{\max}$ is changed to 0.70 in order to ensure that $\beta^1(0) < S_0$. Benchmark values are computed from a lattice method with $N = 204,800$ time steps.

Table 7.2 shows that when $\beta^{N-1}(T)$ is 130 or 140, there is a slight low bias in option values. When $\beta^{N-1}(T)$ is 110 or 120, bias is within the range ± 2 and contour parameters are robust with different values of S_0. Since these are options that will be considered in this chapter, $\beta^{N-1}(T)$ is chosen to be either 110 or 120.

S_0		$\beta^{N-1}(T)$			
Benchmark		110	120	130	140
120	$\widehat{v}^{\infty}$	4.4881	4.4879	4.4843	4.4896
4.48915	(se)	(0.0019)	(0.0021)	(0.0020)	(0.0019)
	$[t]$	[371]	[381]	[385]	[391]
	{b}	{-0.54}	{-0.60}	{-2.4}	{0.22}
110	$\widehat{v}^{\infty}$	3.3939	3.3958	3.3923	3.3906
3.39444	(se)	(0.0025)	(0.0024)	(0.0024)	(0.0024)
	$[t]$	[329]	[337]	[343]	[349]
	{b}	{-0.23}	{0.57}	{-0.90}	{-1.6}
100	$\widehat{v}^{\infty}$	2.2661	2.2691	2.2631	2.2627
2.26822	(se)	(0.0023)	(0.0023)	(0.0024)	(0.0023)
	$[t]$	[231]	[237]	[239]	[243]
	{b}	{-0.94}	{0.38}	{-2.1}	{-2.4}
90	$\widehat{v}^{\infty}$	1.2557	1.2552	1.2496	1.2513
1.25443	(se)	(0.0021)	(0.0021)	(0.0019)	(0.0019)
	$[t]$	[135]	[138]	[143]	[150]
	{b}	{0.61}	{0.37}	{-2.5}	{-1.6}

Table 7.2. American fractional call values with different $\beta^{N-1}(T)$

Figure 7.4 illustrates the type 5 contours. Parameter values are $\alpha^{N-1} = \frac{1}{0.001}$, $\beta^{N-1}(T) = 120$, $g = 11.69$, $C = 5$, $k_{\max} = 0.85$, and $k_{\max} = 0.1$. The option has 64 exercise dates.

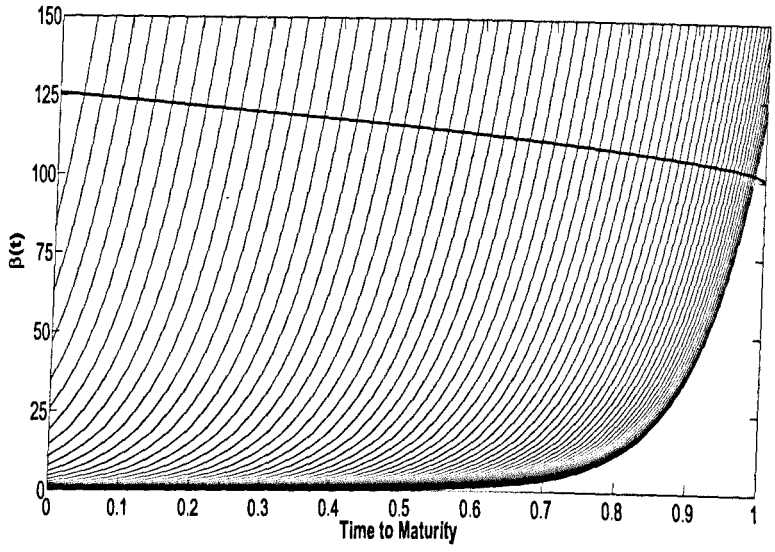


Figure 7.4. Contour construction for an American fractional power call with its EEB from Richardson extrapolation

7.3.2.1 Values of S_T on the ith contour

Section 6.1.2.2 discusses two methods to compute S_T. These are cases where $\tau_j^{N-1} < T$ in equation (6.1.10) and $\tau_j^{N-1} > T$ in equation (6.1.11). When valuing an American put, values of S_T where $\tau_j^{N-1} > T$ are not needed (see section 6.2). However, for an American call option, they have to be taken into account since they are in-the-money values. This section describes how these values of S_T are used in the method.

Figure 7.5 illustrates the values of S_T from the two cases where $\tau_j^{N-1} < T$ and $\tau_j^{N-1} > T$ with two exponential contours $\beta^{N-1}(t)$ and $\beta^{N-2}(t)$. Figure 7.5 shows three regions; (a), (b), and (c). In region (c), S_T takes values in $[0, X)$. In region (a), S_T has values in $[X, \beta^{N-1}(T))$. In region (b), S_T takes values in $[\beta^{N-1}(T), \beta^{N-2}(T))$. Regions (a) and (c) correspond to the case where $\tau_j^{N-1} < T$ and values of S_T in these regions are computed by using (6.1.10). Since values of S_T in region (c) are the out-of-the-money values (for a call option) and are not used in the algorithm, they need not be computed. Region (b) corresponds to the case where $\tau_j^{N-1} > T$ and thus values of S_T in this region are computed by using (6.1.11).

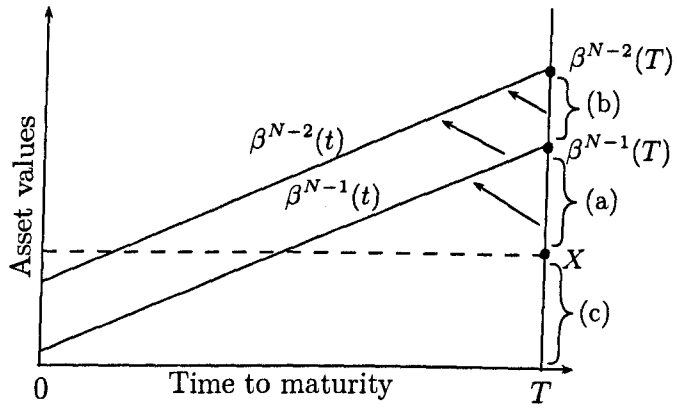


Figure 7.5. The values of S_T (area (a), (b) and (c))

Numerical examples suggest that the approximation in (6.1.11) are used only ~2% of total sample paths on the ith contour, and approximately for the last 5% of N contours. Thus, noises that come from the approximation in (6.1.11) do not have a great impact on the option value.[5]

When computing values of SC fractional power call options, these values of S_T must be used. On the Nth contour (a vertical contour from 0 to $\beta^{N-1}(T)$), the option values are those lying along regions (a) and (c). On the $(N-1)$th contour, the option values are those lying on $\beta^{N-1}(t)$ and those in region (b) which is the interval $[\beta^{N-1}(T), \beta^{N-2}(T))$. On the $(N-2)$th contour, the option values are those on $\beta^{N-2}(t)$ and those in the interval $[\beta^{N-2}(T), \beta^{N-3}(T))$ (not shown in figure 7.5). In general, on the ith contour, the option values are those on $\beta^i(t)$ and also those lying in the interval $[\beta^i(T), \beta^{i-1}(T))$.

7.4 Application to American linear barrier fractional power call options

This section illustrates how the SCMC method can be extended to value a (down) American knock-in linear barrier option. This is a barrier option that is knocked in to an American fractional power option. The linear barrier b_t is

$$b_t = l + mt, \tag{7.4.1}$$

where $l \in \mathbb{R}^+$ and $m \in \mathbb{R}$.

Suppose the barrier option matures at time T, and the underlying American

[5]Numerical examples in section 7.5 shows that biases are within the range ± 2.

fractional power option matures at time $T_1 \geq T$. Let $\tau_b = \inf\{t \in (0,T] : S_t \leq b_t\}$. An American knock-in option comes into existence at time τ_b if $\tau_b < T$. Then a holder can exercise early at time τ where $\tau \in [\tau_b, T_1]$. In this case, set $T_1 = T$.

The valuation algorithm consists of two main phases: (1) the hitting times phase and (2) the valuation phase. They are discussed now.

Phase 1: Finding hitting times

When valuing an European knock-in option, knowledge of the value of τ_b is not required: one only needs to know $\iota = \mathbb{I}_{\{\tau_b \in [0,T]\}}$, that is, whether an asset hits the option barrier anytime between time 0 to T.

Define the interval $[\underline{\tau}^*, \bar{\tau}^*]$ such that $\underline{\tau}^* \leq \tau_b \leq \bar{\tau}^*$. For an American knock-in option, establishing the time interval $[\underline{\tau}^*, \bar{\tau}^*]$ is crucial.[6] Let τ^{a} be the exercise time of the American option. Since the American option comes to existence only at τ_b, it can be exercised early at times $\tau^{\mathrm{a}} \in [\tau_b, T]$.

The objective of this phase is to determine the interval $[\underline{\tau}^*, \bar{\tau}^*]$ that contains the first hitting time τ_b to the option barrier. On the jth path, the method generates a set $\tau_j = \left\{\tau_j^i\right\}_{i=1,\ldots,N}$, where τ_j^i is defined in section 7.1. The algorithm finds an i such that $\tau_j^{i-1} \leq \tau_{b_j} \leq \tau_j^i$, if such an i exists. Then set $\underline{\tau}^* = \tau_j^{i-1}$ and $\bar{\tau}^* = \tau_j^i$.

Let S_j^i be the simulated value for a hitting time τ_j^i to the ith contour on the jth sample path. Given a pair of asset values $\left(S_j^{i-1}, S_j^i\right)$, one has to determine if $\tau_j^{i-1} < \tau_{b_j} < \tau_j^i$. This is straightforward because the contour bridge method described in section 3.2 can be directly applied.

Fix a set $\alpha = \{\alpha^i\}_{i=1,\ldots,N}$. The method first generates hitting times to the fixed set of N contours $\left\{\beta\left(\tau \mid \alpha^i\right)\right\}_{\tau \in [0,T]}$, $i = 1, \ldots, N$, to obtain a set of sample paths of hitting times. To establish whether $\tau_j^{i-1} < \tau_{b_j} < \tau_j^i$ for a given i, a set of refined contours $\left\{\beta^{l,l}\right\}_{l=1,\ldots,L}$ lying between $\beta^{i-1}(t)$ and $\beta^i(t)$ is generated. Note that the value of L is not fixed but is determined by the algorithm. Write $b^i = b\left(\tau_j^i\right)$. The algorithm works forward from $\beta^i(t)$ to $\beta^N(t)$. It proceeds as follows.

The algorithm to find the interval $[\underline{\tau}^*, \bar{\tau}^*]$:

1. Fix a set of contours $\beta = \left\{\beta^i(t)\right\}_{i=1,\ldots,N}$. Set $\tau_j^0 = 0$ and $S_j^0 = S_0$. Construct a set of sample paths $\tau_j = \left\{\tau_j^0, \ldots, \tau_j^N\right\}_{j=1,\ldots,M}$ and $S_j = \left\{S_j^0, \ldots, S_j^N\right\}_{j=1,\ldots,M}$. Set $j = 1$ and $i = 1$. Proceed to step 2.

[6] Note that one needs to know only the interval $[\underline{\tau}^*, \bar{\tau}^*]$, not the exact value of τ_b.

2. At the ith contour, determine if $\tau_j^{i-1} \leq \tau_{b_j} \leq \tau_j^i$ by checking whether $S_j^i < b^i$. If $S_j^i < b^i$, then one establishes that $\tau_{b_j} < \tau_j^i$. Set $\underline{\tau}^* = \tau_j^{i-1}$ and $\bar{\tau}^* = \tau_j^i$. Then set $i = 1$ and increment $j \leftarrow j + 1$. Go back to step 2.

   If $S_j^i > b^i$, increment $i \leftarrow i + 1$ and apply step 2.

3. If $i = N$, this implies $S_j^i > b^i \ \forall i = 1, \ldots, N$. Must check whether an option barrier is breached between $\left(\tau_j^{i-1}, \tau_j^i\right) \ \forall i = 1, \ldots, N$. To do this, reset $i = 1$ and apply step (3a).

   (3.a) At the ith step, one has $S_{i-1} > b_{i-1}$ and $S_i > b_i$. Apply the algorithm in section 3.2 in between bounding contour $\beta^{i-1}(t)$ and $\beta^i(t)$ with a set of refined contours $\left\{\beta'^{,l}\right\}_{l=1,\ldots,L}$.

   (3.b) If one can establish that $\tau_j^{i-1} \leq \tau_{b_j} \leq \tau_j^i$, set $\underline{\tau}^* = \tau_j^{i-1}$ and $\bar{\tau}^* = \tau_j^i$. Then, set $i = 1$ and increment $j \leftarrow j + 1$. Go back to step 2.

   (3.c) If $\tau_j^{i-1} \not\leq \tau_{b_j} \not\leq \tau_j^i$, increment $i = i + 1$ and apply step 3a.

   (3.d) If $i = N$, this implies the option barrier has not been breached. Set $\underline{\tau}^* = \bar{\tau}^* = \infty$ for that jth path. Set $i = 1$. Increment $j \leftarrow j + 1$ and apply step (2).

At the end of the algorithm, one obtains a set of pairs $\left\{\left(\underline{\tau}_j^*, \bar{\tau}_j^*\right)\right\}_{j=1,\ldots,M}$.

Since the standard LSLS method is equivalent to sampling a hitting time to a vertical contour, the standard LSLS implements the box vertical method. For the sequential contour method, the single hit variant method is implemented (see section 3.2.6).

Phase 2: Knock-in option valuation

First, the value of the underlying American option is found. Then, the value for the barrier option can be found. The American option is valued using the algorithm described in section 7.2.1 (as the one used to find the option's EEB). The method uses the same set of sample paths $\tau_j = \left\{\tau_j^0, \ldots, \tau_j^N\right\}_{j=1,\ldots,M}$ and $S_j = \left\{S_j^0, \ldots, S_j^N\right\}_{j=1,\ldots,M}$ as the one used for obtaining the interval $\left[\underline{\tau}_j^*, \bar{\tau}_j^*\right]$ that contains τ_b. Write $\widehat{v}_j^{\mathrm{a},i}$ as the American option value at the ith contour on the jth path. After implementing the algorithm in section 7.2.1, at time $\tau^0 = 0$, one has a set $\left\{\widehat{v}_j^{\mathrm{a},0}\right\}_{j=1,\ldots,M}$.

To determine the value of the knock-in barrier option $\widehat{v}_j^{\mathrm{b}}$, one requires a knowledge of $\left\{\left(\underline{\tau}_j^*, \bar{\tau}_j^*\right)\right\}_{j=1,\ldots,M}$. On the ith contour, one needs to check if $\tau_j^i \geq \bar{\tau}_j^*$

or if $\tau_j^i \leq \underline{\tau}_j^*$. When $\tau_j^i \geq \bar{\tau}_j^*$, the option's barrier is already hit and a holder can choose to exercise the underlying American option. However, if $\tau_j^i \leq \underline{\tau}_j^*$, the option's barrier is not yet hit and the underlying American option does not come into existence.

Hence, at time $\tau^0 = 0$, set

$$\widehat{v}_j^{\mathrm{b},0} = \begin{cases} \mathrm{e}^{-r\tau_j^i}\widehat{v}_j^{\mathrm{a},i}, & \text{for } \tau_j^i = \bar{\tau}_j^* < T, \\ 0, & \text{for } \underline{\tau}_j^* > T \end{cases} \tag{7.4.2}$$

Then the value of the knock-in option is $\widehat{v}^{\mathrm{b},0} = \frac{1}{M}\sum_{j=1}^{M}\widehat{v}_j^{\mathrm{b},0}$.

On the $j$path, when $\bar{\tau}_j^* < T$, the American option comes into existence before time T. Its value can be observed for the first time at $\bar{\tau}_j^*$. Hence, the value of the knock-in option is the discounted value of the American option at time $\tau_j^i = \bar{\tau}_j^*$. If $\underline{\tau}_j^* > T$, the option's barrier is not hit and thus its value on that particular jth path is 0.

7.5 Numerical Results

This section reports numerical results for three different options: the standard American put option, the American fractional power call options and the barrier option that knocks in to an American fractional power call option. Parameter values are $X = 100$, $\sigma = 0.2$, $r = 0.05$ and $T = 1$. Monte Carlo parameter values are $M = 50,000$, $N \in \{16, 32, 64\}$. The method uses scaled Laguerre basis function where the number of basis functions K is 12. To approximate the American value, Richardson extrapolation is used on three SC option values $\widehat{v}_{64}$, $\widehat{v}_{32}$ and $\widehat{v}_{16}$. The roll-back control variate is call-tau (call option sampled at exercise times. See section 5.2.2) and the pricing control variate for each sequential contour options is also call-tau.

Let τ^a be the exercise time of the option. The call-tau control variate, $\mathrm{c}_{\tau^a}^c$ is found by

$$\mathrm{c}_{\tau^a}^c = \begin{cases} \mathrm{e}^{-r\tau^a}c_{\tau^a} - c_0, & \text{if } \tau^a < T, \\ \mathrm{e}^{-r\tau^a}(S_{\tau^a} - X)^+ - c_0, & \text{Otherwise,} \end{cases} \tag{7.5.1}$$

where c_0 is the time t_0 Black-Scholes value of an European call option.

7.5.1 Sequential contour bridge (SCB) method with American put options

Table 7.3 reports extrapolated American put option values from the SCB method for different values of S_0. The projection methods are $\mathcal{S}$-projection and $\mathcal{D}$-projection. Note that, when valuing a put option, there are biases in option values when using $\mathcal{T}$-projection. This issue will be investigated in section 7.5.1.1. The benchmark values are computed from a trinomial lattice with $N = 204,800$ time steps. The table reports extrapolated American put values $\widehat{v}^{\infty}$, standard errors (se), computational times [t], biases {b} (3.3.1) and efficiency gains **e** (3.3.2). Gains are computed against the standard LSLS method. Table 7.3 reports values from the standard SCMC method (forward-evolution) described in chapter 6 and the SCB method (backward-evolution) described in section 7.1.

In this case, type 5 contour is used (section 7.1.1). Contour parameters are $\beta^{N-1}(0) = \frac{1}{\alpha^0} = 0.001$, $C = 5$, $k_{\max} = 0.95$, $k_{\min} = 1$ and $g = 11.51$. Table 7.3 also shows extrapolated American put option values computed from the standard LSLS method with Brownian bridge (backward-evolution).

The results show no evidence of bias for the $\mathcal{S}$-projection and $\mathcal{D}$-projection methods. The biases are all within a range ± 2. Standard errors are constant across all methods. Regarding the sequential contour methods, the backward evolution (SCB) technique produces similar results to the forward evolution (SCMC) technique both for $\mathcal{S}$-projection and $\mathcal{D}$-projection.

Both the $\mathcal{S}$ and $\mathcal{D}$-projections yield similar results for all values of S_0. This is because, with the chosen set of contours, their steep shape causes the value of S_j^i and d_j^i to be quite close to each other.

When comparing the sequential contour results with those produced from the standard LSLS, no efficiency agains are observed. Efficiency gains, **e**, are approximately in the range $\mathbf{e} \in [0.90, 1.02]$. This suggests the performance of the sequential contour method (both forward and backward) with $\mathcal{S}$- and $\mathcal{D}$-projections is similar to that of the standard LSLS method when valuing American put options.

7.5.1.1 Bias in the American put value from the $\mathcal{T}$-projection

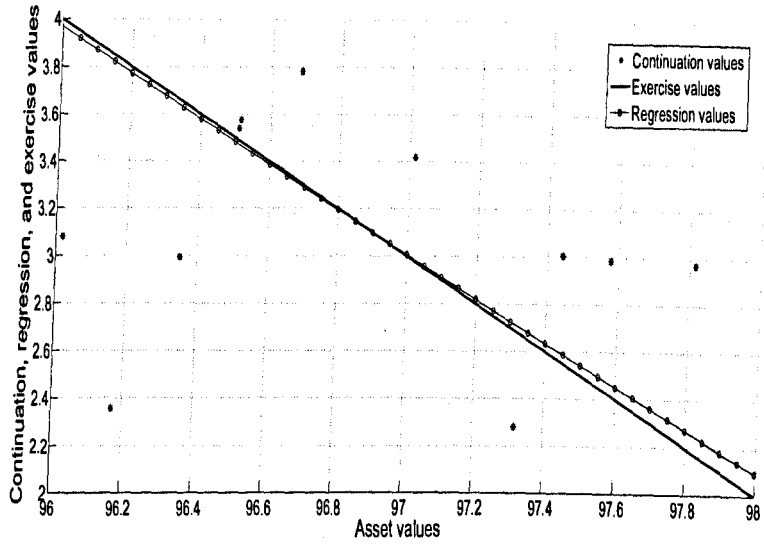
Although $\mathcal{S}$- and $\mathcal{D}$-projections are unbiased, it turns out that the $\mathcal{T}$-projection exhibits bias. This section investigates this bias.

The source of bias comes from the noise in estimating τ_i^E in (7.2.11). Figure 7.6 illustrates the source of such a noise.

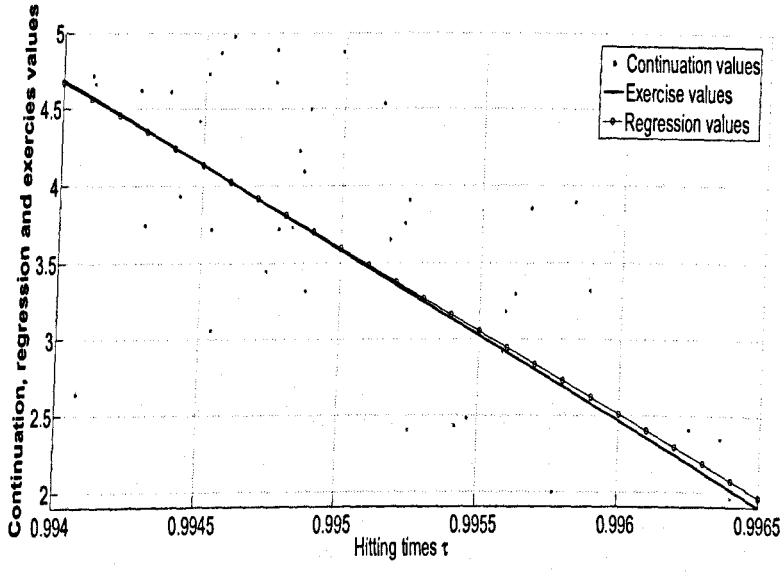
Panel (a) in figure 7.6 shows the continuation values, regression values and

S_0 Benchmark		LSLS Backward	$\mathcal{S}$-Projection Forward	$\mathcal{S}$-Projection Backward	$\mathcal{D}$-Projection Forward	$\mathcal{D}$-Projection Backward
110	$\widehat{v}^{\infty}$	2.9864	2.9871	2.9862	2.9864	2.9868
2.9865	(se)	(0.0010)	(0.0010)	(0.0010)	(0.0010)	(0.0010)
	$[t]$	[92]	[104]	[104]	[107]	[106]
	{b}	{-0.11}	{0.54}	{-0.30}	{-0.12}	{0.23}
	e	**1.0**	**0.91**	**0.92**	**0.95**	**0.90**
105	$\widehat{v}^{\infty}$	4.3024	4.3052	4.3054	4.3052	4.3065
4.3044	(se)	(0.0013)	(0.0013)	(0.0013)	(0.0012)	(0.0013)
	$[t]$	[129]	[138]	[137]	[143]	[140]
	{b}	{-1.6}	{0.59}	{0.74}	{0.59}	{1.6}
	e	**1.0**	**0.94**	**0.93**	**0.94**	**0.94**
100	$\widehat{v}^{\infty}$	6.0904	6.0901	6.0890	6.0892	6.0899
6.0904	(se)	(0.0016)	(0.0016)	(0.0016)	(0.0016)	(0.0016)
	$[t]$	[185]	[188]	[187]	[190]	[189]
	{b}	{0.02}	{-0.15}	{-0.85}	{-0.73}	{-0.28}
	e	**1.0**	**0.98**	**0.98**	**0.93**	**0.96**
95	$\widehat{v}^{\infty}$	8.4497	8.4504	8.4533	8.4495	8.4504
8.4510	(se)	(0.0019)	(0.0019)	(0.0019)	(0.0019)	(0.0019)
	$[t]$	[249]	[245]	[244]	[246]	[245]
	{b}	{-0.67}	{-0.3}	{1.2}	{-0.79}	{-0.30}
	e	**1.0**	**1.04**	**1.02**	**1.01**	**1.02**
90	$\widehat{v}^{\infty}$	11.4901	11.4896	11.4899	11.4922	11.4937
11.4927	(se)	(0.0023)	(0.0023)	(0.0023)	(0.0023)	(0.0023)
	$[t]$	[297]	[294]	[290]	[292]	[292]
	{b}	{-1.2}	{-1.4}	{-1.2}	{-0.22}	{0.43}
	e	**1.0**	**1.01**	**1.01**	**0.99**	**1.00**

Table 7.3. American put, $\sigma = 0.2$



(a) $\mathcal{S}$-projection



(b) $\mathcal{T}$-projection

Figure 7.6. Continuation values, regression values and exercise values from the $\mathcal{S}$-projection and the $\mathcal{T}$-projection method, for a SC put option, on the 62th contour

exercise values of the SC fractional call options from the $\mathcal{S}$-projection with 64 exercise opportunities. In panel (a), it can be observed that the value of S_i^E (where the regression line crosses the payoff line) is ~97.

Panel (b) shows the same plot produced by the $\mathcal{T}$-projection. In this case, it is not clear where the regression line crosses the payoff line. With the $\mathcal{T}$-projection, the slopes of the exercise value line and that of the regression value line are very close. Hence, it is difficult to estimate τ_i^E accurately. This introduces noise when estimating τ_i^E using $\mathcal{T}$-projection.

Furthermore, since S_{τ^i} is an exponential function of τ^i, $S_{\tau^i} = \frac{1}{\alpha^i}\exp(g\tau^i)$, a small simulation noise in estimating τ_i^E will result in a much larger noise in the option's EEB (in terms of the asset value S). For instance, with a high contour growth rate g ~11.5, a 0.1% difference in τ_i^* results in a 1% difference in option's EEB. This is an enormous 10 times increase in error.

One can see that the error should increase as the contour growth rate g increases. Hence, to reduce bias in option values, one should set g so that contours are shallower. However, contours cannot be too shallow because this means that there are fewer in-the-money paths which will cause the option value to be biased low.

S_0		$\mathcal{T}$-projection		
		g		
Benchmark		11.51	2.30	1.61
110	$\widehat{v}^{\infty}$	2.9843	2.9854	2.9835
2.9865	(se)	(0.0010)	(0.00095)	(0.00093)
	$[t]$	[90]	[98]	[102]
	{b}	{-2.2}	{-1.2}	{-3.2}
105	$\widehat{v}^{\infty}$	4.2968	4.3030	4.3004
4.3044	(se)	(0.0013)	(0.0012)	(0.0012)
	$[t]$	[127]	[132]	[134]
	{b}	{-5.9}	{-1.2}	{-3.4}
100	$\widehat{v}^{\infty}$	6.0829	6.0897	6.0852
6.0904	(se)	(0.0016)	(0.0015)	(0.0016)
	$[t]$	[173]	[175]	[197]
	{b}	{-4.7}	{-0.42}	{-3.3}
95	$\widehat{v}^{\infty}$	8.4408	8.4481	8.4455
8.4510	(se)	(0.0020)	(0.0019)	(0.0019)
	$[t]$	[194]	[223]	[232]
	{b}	{-5.1}	{-1.5}	{-2.9}

Table 7.4. Bias in extrapolated American put values using $\mathcal{T}$-projection

Table 7.4 illustrates the extent of biases in American put values from using the SCB method with the $\mathcal{T}$-projection. Values of growth rate are 11.51, 2.3, and 1.61 which correspond to $\frac{1}{\alpha^{N-1}} = 0.001$, 10, and 20 respectively. These values are chosen such that the contours become shallower.

When $g = 11.51$ (this value of g is used in table 7.3), biases are large. They are around 5 standard errors away from the benchmark value. When g is lowered to $\sim$2.30, the biases are reduced to be within the range ± 2. However, all bias signs are negative which implies there may still be bias, although low, in the option value. When further lowering g to $\sim$1.61, the biases increase. Note that these are biases from misplacing the contours. With shallower contours, the contours are pushed toward option's maturity which means most of the exercise decisions will be done further away from time t_0. This introduces bias in the option value.

Hence, it is possible to reduce a bias in the American put value computed $\mathcal{T}$-projection by making a suitable choice of g. Even though the biases are in ± 2, a small bias still persists.

7.5.2 American fractional power call option results

In this section American fractional power call option values are reported. The parameter κ in (7.3.1) is set to be 0.5. Benchmark values are computed from a lattice method with $N = 204,800$. Type 5 contour is used. Contour parameters are $\beta^{N-1}(0) = \frac{1}{\alpha^0} = 0.001$, $\beta^{N-1}(T) = 120$, $C = 5$, $k_{\max} = 0.85$, $k_{\min} = 1$ and $g = 11.69$. Note that since numerical results suggest that there is no significant difference in the performance between the SCMC (forward-evolution) and the SCB (backward-evolution) methods, the SCMC method is used in this case.

Results in table 7.5 show no evidence of bias for any of the three projections. Biases are all in the range ± 2. Computational times for the $\mathcal{T}$-projection are, on average, $\sim$10% less than those from $\mathcal{S}$-projection and $\mathcal{D}$-projection.

To see why using the $\mathcal{T}$-projection introduces no bias in this case, figure 7.7 shows continuation values, regression values and exercise values from the $\mathcal{T}$-projection. Comparing figure 7.7 to panel (b) in figure 7.6, one can observe that it is much easier to determine where the regression line intersects the exercise line in a call option case for the $\mathcal{T}$-projection ($\tau_i^E \sim 0.9838$ in figure 7.7). This is due to the difference in the shape of the payoff between an American put and an American fractional call. The concave shape of the payoff function of an American fractional call, due to $\kappa < 1$ in (7.3.1), make the point where an exercise point first crosses a regression point clearly visible.

Table 7.5 suggests that the SCMC method with $\mathcal{S}$-projection and $\mathcal{D}$-projection

S_0 Benchmark		LSLS	$\mathcal{S}$-proj.	$\mathcal{T}$-proj.	$\mathcal{D}$-proj.
120	$\widehat{v}^{\infty}$	4.4906	4.4879	4.4892	4.4914
4.48915	(se)	(0.0019)	(0.0021)	(0.0019)	(0.0022)
	$[t]$	[394]	[381]	[337]	[384]
	{b}	{0.75}	{-0.62}	{0.04}	{1.1}
	e	**1.0**	**0.88**	**1.2**	**0.82**
110	$\widehat{v}^{\infty}$	3.3942	3.3958	3.3938	3.3946
3.39444	(se)	(0.0023)	(0.0024)	(0.0023)	(0.0024)
	$[t]$	[347]	[337]	[299]	[338]
	{b}	{-0.09}	{0.58}	{-0.28}	{0.0}
	e	**1.0**	**0.99**	**1.2**	**0.96**
100	$\widehat{v}^{\infty}$	2.2656	2.2691	2.2688	2.2701
2.26822	(se)	(0.0024)	(0.0023)	(0.0023)	(0.0025)
	$[t]$	[237]	[237]	[213]	[240]
	{b}	{-1.1}	{0.38}	{0.26}	{0.76}
	e	**1.0**	**1.1**	**1.2**	**0.90**
90	$\widehat{v}^{\infty}$	1.2566	1.2552	1.2533	1.2525
1.25443	(se)	(0.0021)	(0.0021)	(0.0019)	(0.0020)
	$[t]$	[117]	[138]	[123]	[140]
	{b}	{1.1}	{0.38}	{-0.60}	{-1.0}
	e	**1.0**	**0.85**	**1.1**	**0.90**

Table 7.5. American fractional power call option with forward evolution (standard SCMC) method. $\sigma = 0.2$

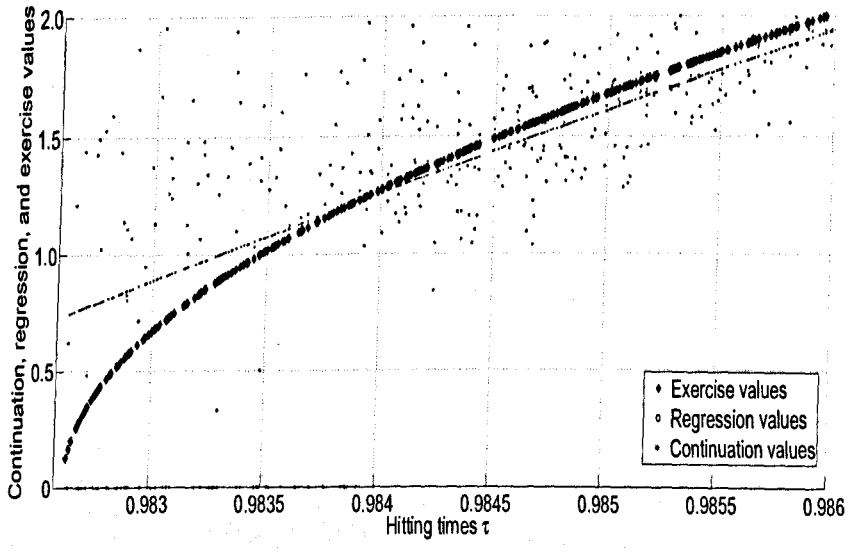


Figure 7.7. Continuation values, regression values and exercise values from the $\mathcal{T}$-projection method, for a SC fractional call option, on the 62th contour

techniques have no gain over the LSLS method. There is a gain of $\sim$1.2 from the $\mathcal{T}$ projection technique. However, this is not substantial.

7.5.3 American knock-in option results

This section reports numerical results for a barrier option that knocks in to an American fractional power call option using the SCMC method. The pricing and rollback control variate is call-tau.

The stopping condition parameters for the contour bridge algorithm (see section 7.4 and 3.2.4) are $\epsilon^1 = 10^{-10}$, $\epsilon^2 = 10^{-4}$, and $\epsilon^p = 10^{-10}$.

7.5.4 Benchmark options: flat barrier with power $\kappa = 1$

When an option barrier is flat and $k = 1$, there exists several approximation techniques in the literature. There are Haug (2001) [119], Dai and Kwok (2004) [79] and Gao et al. (2000) [100].

In this section, benchmark values are computed using the approximation technique from Haug (2001) [119] who applied the reflection principle to approximate the value American down-and-in call option where the barrier, b, is less than the option's strike price, X. The idea is that the number of asset value paths from S_0 to a point higher than X, that hit a barrier b at $\tau_b < T$ is equal to the number of

paths staring from $\frac{b^2}{S_0}$ to a point higher than X.

Write $v^b(S)$, $v^a(S)$ and b for the value of an American knock-in call option, a standard vanilla American call option, and the option's barrier respectively. The American knock-in call and the standard American call have the same parameters, X, r, σ, and T.

The approximation is

$$v^b(S) = \left(\frac{S}{b}\right)^{1-\frac{2(r-\delta)}{\sigma^2}} v^a\left(\frac{b^2}{S}\right), \tag{7.5.2}$$

where δ is the dividend yield. To apply (7.5.2), a trinomial lattice is used to compute $v^a\left(\frac{\beta^2}{S}\right)$ where $N = 204,800$ steps. When $b < X$, where X is the strike price of the American knock-in option, results from the approximation in (7.5.2) and those from the decomposition method of Dai and Kwok (2004) [79] coincide.

Parameter values in Table are $X = 100$, $r = 0.1$, $\delta = 0.08$, $\sigma = 0.3$, and $b = 95$. Values computed using (7.5.2) are shown below values of S_0 in table 7.6.

Table 7.6 presents numerical results for an American knock-in option with a flat barrier and standard call payoff ($\kappa = 1$). The table shows results from the SCMC with all projections, and those from the LSLS method. The gains are computed relatively to the LSLS method with the same call-tau roll-back and pricing control variate.

There is no evidence of bias in option values in table 7.6. Biases are all in the range ± 2. There are slight efficient gains across all values of S_0 and types of projection. One can see that for all values of S_0, the $\mathcal{T}$-projection achieves the highest gains (even though marginal) among the three projections. As S_0 moves away from the option barrier b, the gains increase slightly. This is because as S_0 increases, the vertical contour method will take longer to compute τ_b and hence results in slightly increased gains to the SCMC method.

7.5.5 Exotic American options

This section presents numerical results for American knock-in options with the fractional power payoffs and a linear barrier. Parameter values are $X = 100$, $r = 0.05$, $\sigma = 0.2$ and $T = 1$. Type 5 contour parameters are $k_{\max} = 0.85$, $k_{\min} = 0.1$, $\beta^{N-1}(T \mid \alpha^{N-1}) = 120$, $\beta^{N-1}(0) = \frac{1}{\alpha^0} = 0.001$, $C = 5$, and $g = 11.69$. The barrier $b(t) = l + mt$ is linear $l = b(0) = 95$ and $m = \{-15, -10, -5, 5\}$.

The option power parameter κ in (7.3.1) is set to be 0.5. Benchmark values are computed from a standard LSLS method with $M = 850,000$ sample paths and

S_0 Benchmark		LSLS	$\mathcal{S}$-Proj.	$\mathcal{T}$-Proj.	$\mathcal{D}$-Proj.
105	$\widehat{v}^{\infty}$	5.817	5.823	5.905	5.920
5.846	(se)	(0.052)	(0.051)	(0.049)	(0.052)
	$[t]$	[414]	[226]	[198]	[226]
	{b}	{-0.55}	{-0.46}	{1.2}	{1.4}
	{e}	**1.0**	**1.9**	**2.3**	**1.9**
102	$\widehat{v}^{\infty}$	6.793	6.827	6.701	6.780
6.746	(se)	(0.053)	(0.052)	(0.052)	(0.051)
	$[t]$	[347]	[203]	[180]	[208]
	{b}	{0.89}	{1.6}	{-0.86}	{0.66}
	{e}	**1.0**	**1.8**	**2.0**	**1.8**
100	$\widehat{v}^{\infty}$	7.419	7.436	7.350	7.376
7.414	(se)	(0.050)	(0.050)	(0.048)	(0.049)
	$[t]$	[291]	[188]	[167]	[193]
	{b}	{0.11}	{0.44}	{-1.3}	{0.77}
	e	**1.0**	**1.6**	**1.9**	**1.6**
98	$\widehat{v}^{\infty}$	8.138	8.161	8.152	8.122
8.141	(se)	(0.044)	(0.044)	(0.042)	(0.044)
	$[t]$	[236]	[173]	[156]	[177]
	{b}	{-0.08}	{0.45}	{0.25}	{-0.45}
	e	**1.0**	**1.4**	**1.7**	**1.4**

Table 7.6. Benchmark case: knock-in flat barrier American call options with $\kappa = 1$.

are extrapolated from the Bermudan fractional call values with $N \in \{32, 64, 128\}$ exercise opportunities.

Note that in the first columns in tables 7.7 to 7.9, the slopes of the option barrier and benchmark values are presented as triplets:

$$\begin{pmatrix} m \\ \widehat{v} \\ (se) \end{pmatrix} = \begin{pmatrix} -15 \\ 0.3746 \\ (0.0013) \end{pmatrix}. \tag{7.5.3}$$

In top entry is the slope of the option barrier, the middle entry is the Monte Carlo benchmark value and the bottom entry in round brackets is the standard error of the Monte Carlo benchmark value.

Benchmark values		Simulation values			
		LSLS	$\mathcal{S}$-Proj.	$\mathcal{T}$-Proj.	$\mathcal{D}$-Proj.
-15	$\widehat{v}^{\infty}$	0.3822	0.3764	0.3791	0.3688
0.3746	(se)	(0.0047)	(0.0047)	(0.0047)	(0.0046)
(0.0013)	$[t]$	[739]	[272]	[246]	[279]
	{b}	{1.6}	{0.39}	{1.0}	{-1.2}
	{e}	**1.0**	**2.8**	**3.0**	**2.7**
-10	$\widehat{v}^{\infty}$	0.5191	0.5142	0.5280	0.5130
0.5204	(se)	(0.0054)	(0.0053)	(0.0055)	(0.0054)
(0.0015)	$[t]$	[689]	[271]	[243]	[276]
	{b}	{-0.25}	{-1.2}	{1.4}	{-1.4}
	{e}	**1.0**	**2.6**	**2.8**	**2.5**
-5	$\widehat{v}^{\infty}$	0.7212	0.7212	0.7197	0.7329
0.7293	(se)	(0.0061)	(0.0061)	(0.0062)	(0.0062)
(0.0017)	$[t]$	[633]	[270]	[246]	[274]
	{b}	{-1.3}	{-1.3}	{-1.5}	{0.59}
	{e}	**1.0**	**2.3**	**2.5**	**2.2**
5	$\widehat{v}^{\infty}$	1.2813	1.2776	1.2856	1.2769
1.2893	(se)	(0.0076)	(0.0076)	(0.0076)	(0.0076)
(0.0021)	$[t]$	[520]	[268]	[240]	[273]
	{b}	{-1.0}	{-1.5}	{-0.49}	{-1.6}
	{e}	**1.0**	**2.0**	**2.2**	**1.9**

Table 7.7. Knock-in linear barrier fractional power American call options, $S_0 = 105$

Tables 7.7 to 7.9 report the values of knock-in linear barrier fractional power American call options. Results suggest that there is no evidence of bias in these values. All option values shown in these tables have biases within ± 2.

Benchmark values		Simulation values			
		LSLS	$\mathcal{S}$-Proj.	$\mathcal{T}$-Proj.	$\mathcal{D}$-Proj.
-15	$\widehat{v}^{\infty}$	0.8040	0.8013	0.7990	0.7880
0.7948	(se)	(0.0064)	(0.0064)	(0.0063)	(0.0063)
(0.0017)	$[t]$	[508]	[220]	[198]	[222]
	{b}	{1.4}	{1.0}	{0.66 }	{-1.1}
	{e}	**1.0**	**2.3**	**2.6**	**2.3**
-10	$\widehat{v}^{\infty}$	0.9287	0.9295	0.9237	0.9286
0.9329	(se)	(0.0067)	(0.0067)	(0.0067)	(0.0067)
(0.0016)	$[t]$	[469]	[219]	[199]	[221]
	{b}	{-0.62}	{-0.50}	{-1.4}	{-0.64}
	{e}	**1.0**	**2.1**	**2.4**	**2.1**
-5	$\widehat{v}^{\infty}$	1.0874	1.0889	1.0903	1.0856
1.0953	(se)	(0.0070)	(0.0070)	(0.0070)	(0.0070)
(0.0017)	$[t]$	[421]	[218]	[199]	[221]
	{b}	{-1.1}	{-0.92}	{-0.71}	{-1.4}
	{e}	**1.0**	**1.9**	**2.1**	**1.9**
5	$\widehat{v}^{\infty}$	1.4590	1.4531	1.4607	1.4577
1.4616	(se)	(0.0074)	(0.0075)	(0.0074)	(0.0074)
(0.0019)	$[t]$	[346]	[219]	[195]	[220]
	{b}	{-0.35}	{-1.1}	{-0.12}	{-0.52}
	{e}	**1.0**	**1.6**	**1.8**	**1.6**

Table 7.8. Knock-in linear barrier fractional power American call options, $S_0 = 100$

Benchmark values		Simulation values			
		LSLS	$\mathcal{S}$-Proj.	$\mathcal{T}$-Proj.	$\mathcal{D}$-Proj.
-15	$\widehat{v}^{\infty}$	1.0868	1.0885	1.0846	1.0744
1.0816	(se)	(0.0068)	(0.0068)	(0.0068)	(0.0068)
(0.0016)	$[t]$	[384]	[197]	[178]	[198]
	{b}	{0.76}	{1.0}	{0.44}	{-1.1}
	{e}	**1.0**	**2.0**	**2.2**	**2.0**
-10	$\widehat{v}^{\infty}$	1.1794	1.1933	1.1873	1.1901
1.1895	(se)	(0.0069)	(0.0069)	(0.0068)	(0.0069)
(0.0017)	$[t]$	[352]	[196]	[178]	[199]
	{b}	{-1.5}	{0.56}	{-0.31}	{0.10}
	{e}	**1.0**	**1.8**	**2.0**	**1.8**
-5	$\widehat{v}^{\infty}$	1.3067	1.3104	1.3131	1.2949
1.3050	(se)	(0.0070)	(0.0070)	(0.0069)	(0.0069)
(0.0017)	$[t]$	[322]	[196]	[177]	[196]
	{b}	{0.25}	{0.77}	{1.2}	{-1.5}
	{e}	**1.0**	**1.7**	**1.9**	**1.7**
5	$\widehat{v}^{\infty}$	1.5408	1.5423	1.5473	1.5461
1.5485	(se)	(0.0069)	(0.0069)	(0.0069)	(0.0069)
(0.0017)	$[t]$	[266]	[194]	[176]	[199]
	{b}	{-1.1}	{-0.91}	{-0.18}	{-0.35}
	{e}	**1.0**	**1.4**	**1.5**	**1.3**

Table 7.9. Knock-in linear barrier fractional power American call options, $S_0 = 98$.

Like the benchmark case, the highest gains come from the $\mathcal{T}$-projection across all values of m and S_0. As the slope of the option barrier decreases, gains are larger. Similarly, as S_0 moves further away from the option barrier $b(0)$ at time t_0, gains also increase. This is because, on the ith contour, the vertical barrier method takes a greater computational time to establish if $\tau^{i-1} \leq \tau_b \leq \tau^i$ than the single hit method does, and this difference becomes more pronounced as the barrier is decreasing and S_0 is larger.

The highest gain (of 3) over the standard method comes from the $\mathcal{T}$-projection technique when $S_0 = 105$ with $m = -15$. Gains from the $\mathcal{S}$- and $\mathcal{D}$-projections are slightly lower; these are 2.8 and 2.7 respectively. Even when S_0 is close to b_0 and $m = -15$, all three methods are at least twice as fast as the standard LSLS method. The least gains are $\sim$1.5 when S_0 is close to the option barrier ($S_0 = 98$) with a positive slope, $m = 5$.

It is clear that the overall gains when valuing American knock-in options are larger than those obtained from the same method when valuing American put and American fractional call options. The reason comes from the knock-in valuation phase as discussed earlier.

7.6 Conclusion

In this chapter, new simulation techniques to value several types of American options have been described. First, the sequential contour bridge method is introduced and discussed. Second, different projection techniques are described, namely the $\mathcal{T}$-projection and the $\mathcal{D}$-projection. These projections are applied to value standard American put options and American fractional power call options. Then a linear knock-in American barrier option is introduced. To value this American option, the generalised LSLS algorithm is combined with the contour bridge method described in chapter 3.

The results in section 7.5 suggest that there is no evidence of bias in American put option values when using the sequential contour bridge method with the $\mathcal{S}$- and $\mathcal{D}$-projections. However, there is a low bias with the $\mathcal{T}$-projection when valuing an American put. This seems to be because the convex shape of the American put option causes greater noise in estimating τ_i^E when using the $\mathcal{T}$-projection which results in a suboptimal estimate of the EEB. The results show that bias can be reduced by modifying the shape of the contours but a slight bias can still be observed.

When applying these projection techniques to value American fractional call options, biases in option values from using the $\mathcal{T}$-projection disappears. This is

because the concave shape of the American call's payoff reduces simulation noises from the $\mathcal{T}$-projection. In this case, $\mathcal{T}$-projection achieves very marginal gains over the standard LSLS method.

The SCMC method is extended to value exotic American options; an American linear barrier knock-in option with a fractional power payoff. In this case, the algorithm is modified so that one can determine if the option barrier is hit before the time to maturity T. To do this, the contour bridge method described in section 3.2.3 is incorporated into the SCMC method. Numerical results show no evidence of bias in the extrapolated American knock-in values. The SCMC method with different projections is more effective than the standard LSLS method especially when valuing American knock-in options with a large negative barrier slope with an initial asset value further away from barrier value at time t_0.

Chapter 8

Conclusion

This thesis is concerned with the development of new simulation methods to value exotic barrier options and American options.

Barrier options is one of the most popular exotic options. They have several forms. For instance, a knock-out option expires worthless if the predetermined barrier is hit before the time to maturity. On the contrary, a knock-in option comes into the existence only if the barrier is hit before time to maturity. Barrier options enable investors to avoid or to obtain exposure and they are cheaper than the European option. Even though several methods can value barrier options accurately, these methods may work only when a barrier is in a simple form, namely, a flat or an exponential barrier. Most standard methods can be used to price only a single option at a time. This can be extremely inefficient when pricing a large book of options. Even if a method can price a single option quite rapidly, its cost when applied to a book can be excessive.

The first part of the thesis discusses the valuation problem of barrier options. Chapter 2 provides the literature review of existing methods for valuing barrier options. It points out that although some of the methods can be applied to price standard barrier options accurately, these methods can fail to value a more complex barrier option.

Chapter 3 proposes a novel simulation method, the contour bridge method, that can be applied to the valuation of barrier options with complex non-constant barriers. The main tool of the method is a hitting time simulation. Instead of generating a set of asset values (standard method), the contour bridge method samples hitting times to amendable contours to approximate the hitting time to the option's barrier. The method is applied to value barrier options with different shapes of barrier, ranging from a simple linear barrier to a more complicated trigonometric

shaped barrier. Numerical examples in section 3.3 suggest that bias in option values is in an acceptable range.

In most cases, the contour bridge method achieves efficiency gains over the standard simulation method. Two variants of the method are proposed; the biggest bite and the single hit methods. Results suggest that the biggest-bite version of the method always out-performs the single hit version and most of the time achieves very substantial gains over a standard simulation method. Even though the single-hit version of the method does not achieve gains as large as the biggest-bite version, those gains are still respectable in a number of cases. The method is also argued to produce greater efficiency gains when it is used to value a book of options even though this is not illustrated in the chapter. The method cannot be applied to value knock-out options directly; however, knock-out option value can be computed from the in-out parity.

Parts two and three of the thesis are concerned with the valuation of American options by using Monte Carlo simulation. American options are hard to price because their early exercise feature complicates the valuation problem. The holder of American options can exercise anytime before their expiry date. The values of the underlying process, that makes it optimal for an option holder to exercise, form an early exercise boundary (EEB).

This boundary makes the valuation problem difficult because the early exercise boundary also needs to be found. There exists no closed-form solution to the American valuation problem, and the option value needs to be computed by approximation and numerical techniques.

The review of several valuation methods for an American option is provided in Chapter 4. It includes not only the case where an underlying process follows a geometric Brownian motion, but it also describes American option pricing methods in non-geometric motion models. The chapter also focuses on several simulation methods. Different control variate methods for valuing an American option is discussed. A brief review of an American barrier option and a power option is also given.

It is well known that Monte Carlo simulation can be used to price American options. However, Monte Carlo simulation may be computationally expensive. Therefore, variance reduction techniques must be used in order to implement Monte Carlo simulation efficiently.

Chapter 5 proposes a new variance reduction technique to value Bermudan and American put options. The method is based on the use of twice-exercise Bermu-

dan put options as a control variate. When a Bermudan option has only two exercise dates, its analytical solution can be found and hence it can be used as a control variate. The Bermudan control method is applied in conjunction with the functional form method of Longstaff and Schwartz (1995) [177] (LSLS method). A two-phase simulation is also used. It is a simulation in which the estimation of the option's EEB and the valuation of the option are done separately. In the first phase, the option's EEB is approximated. The second phase uses the EEB obtained from the first phase to value the option. The proposed Bermudan control variate is applied to both phases. In the first phase, it is applied to option's continuation values (roll-back control variate). In the second phase, it is applied to the option value (pricing control variate).

Numerical examples in section 5.5 show that there is negligible bias in Bermudan option values from a two-phase method, which implies that the option's estimated EEB is accurate. The new Bermudan control variate achieves significant gains over the plain LSLS method. In particular, when combining one particular Bermudan control variate, namely, the early Bermudan control variate with the put-tau control variate of Rasmussen (2005) [203], the method achieves very substantial efficiency gains. All gains produced from this combination are greater than those from the method of Rasmussen (2005) [203] alone, by at least a factor of two.

With an accurate value for Bermudan put options, the American put value can be approximated by using three-point Richardson extrapolation. The method extrapolates three Bermudan put options with different number of exercise opportunities. The results have low standard errors and negligible bias in extrapolated American put values. To achieve the same level of accuracy, the standard simulation method must use a very large number of time steps for a Bermudan put value to converge to an American put value, typically around ~10,000 time steps. Therefore, the American put value can be estimated accurately with much lower computational costs. Standard errors of extrapolated American values are further reduced by applying an individual Bermudan put as a control variate.

In Chapter 6, a new simulation method, the sequential contour Monte Carlo (SCMC) method, to value American put options, is introduced. The main idea of the method is to generate hitting times to a fixed set of exponential contours and the option value is obtained by applying the backward algorithm of the LSLS method along sequential contours. This method generalises the standard LSLS method by using a more general family of contours. The advantage of this method is that a set of hitting times on each contour can be used to compute barrier option values and these option values can be used as a control variate.

Unlike a Bermudan option whose exercise dates are predetermined, the exercise date of the option being valued by the SCMC method is not predetermined and depends on the shape of a set of pre-specified contours. This type of option is called the sequential contour (SC) option. The SCMC method approximate values of these options with different exercise dates and then uses these SC option values to estimate the value of an American put option by using Richardson extrapolation.

Results in section 6.4 suggest that, with an appropriate choice of contour parameters, the SCMC method can be used to value American put options with an acceptable range of bias. Further, the SC option value also convergences slightly faster to the American option values than the Bermudan value from the standard LSLS method does. The method uses a rebate option as an additional control variate to an existing put-tau control variate. Although the method can price American put options with negligible bias, efficiency gains from using a rebate option are marginal.

The last part of the thesis extends the SCMC method to value exotic American options. The method further generalises the LSLS method by introducing two new projections. These projections determine independent variables in the regression procedure used to approximate option continuation values. The $\mathcal{T}$-projection uses sampled hitting times as the independent variable in the regression. The $\mathcal{D}$-projection uses the distance between time t_0 to a pair of simulated hitting times and asset values on a contour. The standard method where asset values are used as the independent variable in the regression is the $\mathcal{S}$-projection.

These techniques were applied to the valuation of an American put and an American fractional power call option. Numerical examples in section 7.5 suggest that, for American put options, the $\mathcal{S}$- and $\mathcal{D}$-projections can value the options with negligible bias. However, there is bias observed in option values when using the $\mathcal{T}$-projection. This bias disappears when valuing American fractional power call options. This seems to be because the concave shape of a fractional power call payoff makes it clearer where to exercise the option for the $\mathcal{T}$-projection. There is no gain observed in an American put case. Even though there are gains from the $\mathcal{T}$-projection, these gains are not substantial.

The SCMC method with different projections is applied next to value American knock-in fractional power call options. The barrier for knock-in options is chosen to be both linear and flat. In this case, the SCMC method is modified so that it can value the knock-in part of an option. This is done by incorporating the contour bridge method proposed in Chapter 3. Numerical examples show that bias in option values is in an acceptable range. In this case, there are gains over the standard LSLS method. The highest gain is achieved using the $\mathcal{T}$-projection. Gains from the $\mathcal{S}$-

and $\mathcal{D}$-projections are, even though slightly smaller, respectable.

Note that the methods used in Chapter 3, 6, and 7 rely on sampling a hitting time to an exponential barrier. This means it might not be feasible to extend the methods to other models such as local or stochastic volatility models because the hitting time distribution to an exponential barrier for these models are not known. Even though there are hitting time distributions for a constant elasticity of variance (CEV) model and an Ornstein-Uhlenbeck (OU) process, they are hitting times to a constant barrier and cannot be used.

In future work, it would be interesting to see the efficiency gains when applying the contour bridge method to value a book of options. The gains are expected to be much higher than those from valuing a single option. Also, the method may be extended to value other types of barrier options, perhaps a double barrier option, if a hitting time density to a particular type of contour is known.

The Bermudan control variate can be extended to stochastic volatility models. In this case, it is clear that one may need an efficient numerical integration method to evaluate Bermudan put value in order to use it as a control variate. It is also interesting to see how the method can be extended to value higher-dimension American options.

For future work, the SCMC method can include more barrier options and use a portfolio of options as a control variate to value American options. It is interesting to compare the performance of the put-tau control variate with that of a portfolio of different barrier options.

Appendix A

The derivation of the hitting time bridge distribution

The derivation and the implementation of the inverse Gaussian bridge was done by Ribeiro and Webber (2003) [206]. They consider the use of Monte Carlo simulation to value path dependent options where an underlying process follows a Normal Inverse Gaussian process. The derivation is similar to this case since inverse Gaussian variables are considered. However, the parameterisation is slightly different because in this case hitting times to exponential contours are used.

Write $\beta^i(t) = \beta\left(t \mid \alpha^i\right)$. Suppose one has three exponential contours of the form:

$$\beta^1(t) = \frac{1}{\alpha_1} e^{gt} \tag{A.0.1}$$

$$\beta^\alpha(t) = \frac{1}{\alpha} e^{gt} \tag{A.0.2}$$

$$\beta^2(t) = \frac{1}{\alpha_2} e^{gt}, \tag{A.0.3}$$

where $\beta^1(t) > \beta^\alpha(t) > \beta^2(t)\ \forall t \in [0,T]$ and $\alpha^1 < \alpha < \alpha^2$.

Let τ^{α_1}, τ^α and τ^{α_2} be hitting times to these contours such that $\tau^{\alpha_1} < \tau^\alpha < \tau^{\alpha_2}$. Then let τ^x be a hitting time from τ^{α_1} to τ^α, τ^y be a hitting time from τ^α to τ^{α_2}, and τ^z be a hitting time from τ^{α_1} to τ^{α_2} so that $\tau^z = \tau^x + \tau^y$. These are shown in figure A.1. Write $\mu = r - \frac{1}{2}\sigma^2$, where r and σ are a risk-free rate and volatility

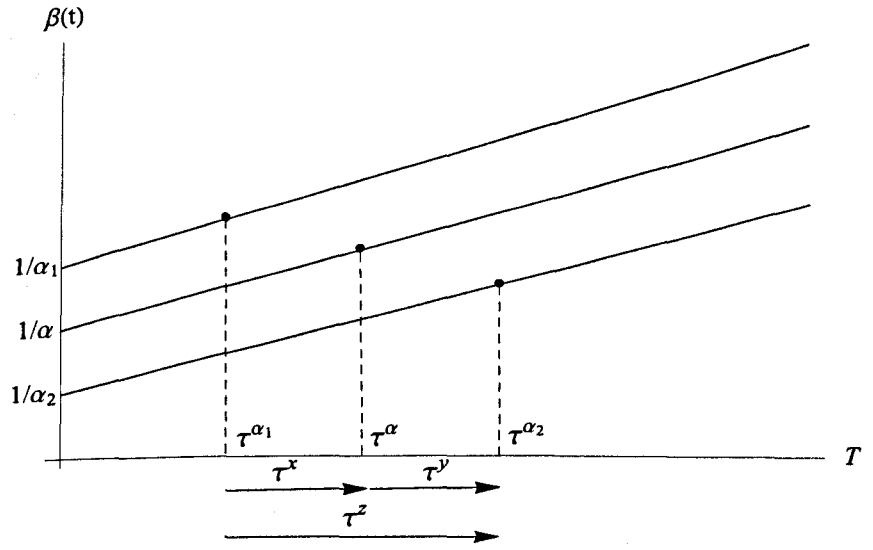


Figure A.1. Illustration of bridge hitting times

of a GBM. The densities of these hitting times are given as follow.

$$f(\tau^x; a_x, \mu) = \frac{a_x}{\sigma\sqrt{2\pi(\tau^x)^3}} \exp\left(-\frac{(a_x - (g-\mu)\tau^x)^2}{2\sigma^2\tau^x}\right) \quad \text{(A.0.4)}$$

$$f(\tau^y; a_y, \mu) = \frac{a_y}{\sigma\sqrt{2\pi(\tau^y)^3}} \exp\left(-\frac{(a_y - (g-\mu)\tau^y)^2}{2\sigma^2\tau^y}\right) \quad \text{(A.0.5)}$$

$$f(\tau^z; a_z, \mu) = \frac{a_z}{\sigma\sqrt{2\pi(\tau^z)^3}} \exp\left(-\frac{(a_x - (g-\mu)\tau^z)^2}{2\sigma^2\tau^z}\right), \quad \text{(A.0.6)}$$

where

$$a_x = \ln\left(\frac{\beta^1(\tau^{\alpha_1})}{\beta^\alpha(\tau^{\alpha_1})}\right) \quad \text{(A.0.7)}$$

$$a_y = \ln\left(\frac{\beta^\alpha(\tau^\alpha)}{\beta^2(\tau^\alpha)}\right) \quad \text{(A.0.8)}$$

$$a_z = \ln\left(\frac{\beta^1(\tau^{\alpha_1})}{\beta^2(\tau^{\alpha_1})}\right). \quad \text{(A.0.9)}$$

Note from (A.0.7) that a_x does not depend of τ^α.

The conditional density of $\tau^x \mid \tau^z$ is

$$f(\tau^x|\tau^z) = \frac{f(\tau^x, \tau^z - \tau^x)}{f(\tau^z)} \tag{A.0.10}$$

$$= \frac{f(\tau^x) f(\tau^y)}{f(\tau^z)}, \tag{A.0.11}$$

where (A.0.11) follows by the independency between τ^x and τ^y. Let $\zeta = g - \mu$. Substituting (A.0.4), (A.0.5) and (A.0.6) into (A.0.11), one has

$$\begin{aligned} f(\tau^x|\tau^z) &= \frac{a_x}{\sigma\sqrt{2\pi(\tau^x)^3}} \exp\left(-\frac{(a_x - \zeta\tau^x)^2}{2\sigma^2\tau^x}\right) \\ &\times \frac{a_y}{\sigma\sqrt{2\pi(\tau^y)^3}} \exp\left(-\frac{(a_y - \zeta\tau^y)^2}{2\sigma^2\tau^y}\right) \\ &\times \left(\frac{a_z}{\sigma\sqrt{2\pi(\tau^z)^3}} \exp\left(-\frac{(a_2 - \mu_2\tau^z)^2}{2\sigma^2\tau^z}\right)\right)^{-1} \\ &= \frac{1}{\sigma\sqrt{2\pi}} \frac{a_x a_y}{a_z} \left(\frac{\tau^x\tau^y}{\tau^z}\right)^{-\frac{3}{2}} \\ &\times \exp\left(-\frac{1}{2}\left(\frac{(a_x - \zeta\tau^x)^2}{2\sigma^2\tau^x} + \frac{(a_y - \zeta\tau^y)^2}{2\sigma^2\tau^y} - \frac{(a_z - \zeta\tau^z)^2}{2\sigma^2\tau^z}\right)\right) \end{aligned} \tag{A.0.12}$$

After some algebra, an exponent term in (A.0.12) can be written as

$$\begin{aligned} &\exp\left(-\frac{1}{2}\left(\frac{(a_1 - \zeta\tau^x)^2}{2\sigma^2\tau^x} + \frac{(a_y - \zeta\tau^y)^2}{2\sigma^2\tau^y} - \frac{(a_z - \zeta\tau^z)^2}{2\sigma^2\tau^z}\right)\right) = \\ &\exp\left(-\frac{1}{2\sigma^2}\left(\zeta^2(\tau^x + \tau^y - \tau^z) - 2\zeta(a_x + a_y - a_z) + \left(\frac{a^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)\right)\right). \end{aligned} \tag{A.0.13}$$

By using $\tau_z = \tau_x + \tau_y$ and (A.0.7) - (A.0.9), the first two terms in an exponent vanish. Hence, the exponent term in (A.0.12) is reduced to

$$\exp\left(-\frac{1}{2\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)\right). \tag{A.0.14}$$

By combining (A.0.12) and (A.0.14), one obtains

$$f(\tau^x \mid \tau^z) = \frac{1}{\sigma\sqrt{2\pi}} \frac{a_x a_y}{a_z} \left(\frac{\tau^x\tau^y}{\tau^z}\right)^{-\frac{3}{2}} \exp\left(-\frac{1}{2\sigma^2}\left(\frac{a_x^2}{\tau^x} + \frac{a_y^2}{\tau^y} - \frac{a_z^2}{\tau^z}\right)\right), \tag{A.0.15}$$

where, with a simplification, $a_x = \ln\left(\frac{\alpha}{\alpha_1}\right)$, $a_y = \ln\left(\frac{\alpha_2}{\alpha}\right)$, and $a_z = \ln\left(\frac{\alpha_2}{\alpha_1}\right)$ as required.

Appendix B

Compute Arc Length of an exponential contour

The objective is to compute a distance on an exponential contour

$$\beta(\tau \mid \alpha) = \frac{1}{\alpha}\exp(g\tau), \tag{B.0.1}$$

where α and g are constant. Let $\tau_2 > \tau_1$, $\tau_1, \tau_2 \in (0, \infty)$ and let s be a distance between $\beta(\tau_1)$ and $\beta(\tau_2)$ along the contour. s can be computed as

$$s = \int_{\tau_1}^{\tau_2} \sqrt{1 + \left(\frac{d\beta}{d\tau}\right)^2}\, d\tau. \tag{B.0.2}$$

Substituting $\left(\frac{d\beta}{d\tau}\right)^2 = g^2\left(\frac{1}{\alpha}\right)^2 \exp(2g\tau)$ into (B.0.2), one has

$$s = \int_{\tau_1}^{\tau_2} \sqrt{1 + g^2\left(\frac{1}{\alpha}\right)^2 \exp(2g\tau)}\, d\tau \tag{B.0.3}$$

To compute definite integral in (B.0.3), one needs to compute

$$\int \sqrt{1 + g^2\left(\frac{1}{\alpha}\right)^2 \exp(2g\tau)}\, d\tau. \tag{B.0.4}$$

Define a new variable $z = g\left(\frac{1}{\alpha}\right)\exp(g\tau)$. Write

$$\tau = \frac{1}{g}\ln\left(\frac{z}{g\left(\frac{1}{\alpha}\right)}\right) \tag{B.0.5}$$

$$\frac{d\tau}{dz} = \frac{1}{gz} \tag{B.0.6}$$

$$d\tau = \frac{1}{gz}dz. \tag{B.0.7}$$

Then equation (B.0.4) becomes

$$\int \sqrt{1+z^2}\frac{1}{gz}dz = \frac{1}{g}\int \sqrt{\frac{1+z^2}{z^2}}dz \tag{B.0.8}$$

Then one uses (Gradshteyn and Ryzhik (2007) [115])

$$\int \sqrt{\frac{1+z^2}{z^2}}dz = \sqrt{1+z^2} + \ln(z) - \ln\left(1+\sqrt{1+z^2}\right). \tag{B.0.9}$$

To verify (B.0.9), one differentiates (B.0.9) to get $\frac{\sqrt{1+z^2}}{z}$. Then one has

$$\begin{aligned}&\frac{d}{dz}\sqrt{1+z^2} + \ln(z) - \ln\left(1+\sqrt{1+z^2}\right)\\ &= \frac{z}{\sqrt{1+z^2}} + \frac{1}{z} - \frac{z}{\sqrt{1+z^2}(1+\sqrt{1+z^2})}.\end{aligned} \tag{B.0.10}$$

Write $X = \sqrt{1+z^2}$. One obtains

$$\begin{aligned}&\frac{z}{\sqrt{1+z^2}} + \frac{1}{z} - \frac{z}{\sqrt{1+z^2}(1+\sqrt{1+z^2})}\\ &= \frac{z}{X} + \frac{1}{z} - \frac{z}{X(1+X)}\\ &= \frac{z^2(1+X)+X(1+X)-z^2}{zX(1+X)}\\ &= \frac{z^2X+X+X^2}{zX(1+X)}\\ &= \frac{z^2+1+X}{z(1+X)}\\ &= \frac{z^2+1+X}{(1+X)}\cdot\frac{1}{z}\end{aligned} \tag{B.0.11}$$

One needs to show that $\frac{z^2+1+X}{(1+X)} = X$. Since one sets $X = \sqrt{1+z^2}$, then $z^2 = X^2 - 1$. Substitute z^2 in (B.0.11), one obtains

$$\frac{z^2+1+X}{(1+X)} = \frac{X(1+X)}{1+X} = X. \tag{B.0.12}$$

This verifies (B.0.9) as required.

Now, by substituting (B.0.9) into (B.0.8), one gets

$$\int \sqrt{1+z^2}\frac{1}{gz}dz = \frac{1}{g}\left(\sqrt{1+z^2} + \ln(z) - \ln\left(1+\sqrt{1+z^2}\right)\right). \tag{B.0.13}$$

Since $z = g\left(\frac{1}{\alpha}\right)\exp(g\tau)$, integral (B.0.4) is

$$\begin{aligned}\int \sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau)}d\tau \;=\; & \frac{1}{g}\Bigg(\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau)} + \ln\left(g\left(\frac{1}{\alpha}\right)\exp(g\tau)\right) \\ & - \ln\left(1+\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau)}\right)\Bigg). \end{aligned} \tag{B.0.14}$$

Finally, s can be explicitly computed as

$$\begin{aligned}\int_{\tau_1}^{\tau_2} \sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau)}d\tau \;=\; & \frac{1}{g}\Bigg\{\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau_2)} - \sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau_1)} \\ & + g(\tau_2-\tau_1) - \ln\left(1+\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau_2)}\right) \\ & + \ln\left(1+\sqrt{1+g^2\left(\frac{1}{\alpha}\right)^2\exp(2g\tau_1)}\right)\Bigg\}. \end{aligned} \tag{B.0.15}$$

Bibliography

[1] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions.* New York: Dover, 1965.

[2] D. H. Ahn, S. Figlewski, and B. Gao. Pricing Discrete Barrier Options with an Adaptive Mesh Model. *Journal of Derivatives*, 6:33–43, 1999.

[3] F. Ait-Sahlia and T.L. Lai. Exercise Boundaries and Efficient Approximations to American Option Prices and Hedge Parameters. *Journal of Computational Finance*, 4(4):85–103, 2001.

[4] F. Ait-Sahlia and T.L. Lai. A Canonical Optimal Stopping Problem for American Options and Its Numerical Solution. *Journal of Computational Finance*, 3(2):33–52, 199.

[5] F. Ait-Sahlia, L. Imhof, and T.L. Lai. Fast and Accurate Valuation of American Barrier Options. *Journal of Computational Finance*, 7(1):129–145, 2003.

[6] F. Ait-Sahlia, L. Imhof, and T.L. Lai. Pricing and Hedging of American Knock-In Options. *Journal of Derivative*, 11(3):44–50, 2003.

[7] L. Alili and A.E. Kyprianou. Some Remarks on First Passage of Levy Process, the American Put and Pasting Principles. *The Annals of Applied Probability*, 15(3):2062–80, 2005.

[8] L. Alili and P. Patie. On the First Crossing Times of a Brownian Motion and a Family of Continuous Curves. *C. R. Acad. Sci. Paris*, 340:225–228, 2005.

[9] L. Alili, P. Patie, and Pedersen J.L. Representations of the First Hitting Time Density of an Ornstein-Uhlenbeck Process. *Stochastic Models*, 21(21):967–980, 2005.

[10] A. Almendral. Numerical Valuation of American Options under the CGMY Process. Technical report, In Exotic Option Pricing and Advanced Levy Models, 2004. URL `http://folk.uio.no/ariel/papers/cgmybook.pdf`.

[11] A. Almendral and C.W. Oosterlee. Highly Accurate Evaluation of European and American Options under the Variance Gamma Process. *Journal of Computational Finance*, 10(1):21–42, 2006.

[12] K. Amin and A. Khanna. Convergence of American Option Values from Discrete to Continuous Time Financial models. *Mathematical Finance*, 4(4): 289–304, 1994.

[13] L. Anderson and M. Broadie. Primal-Dual Simulation Algorithm for Pricing Multidimensional American Options. *Management Science*, 50(9):1222–1234, 2004.

[14] N. Areal, A. Rodrigues, and M.R. Armada. Improvements to the Least Squares Monte Carlo Option Valuation Method. *Working Paper*, NEGE, School of Economics and Management, University of Minho, 2008.

[15] A.F. Atiya and S.A.K. Metwally. Efficient Estimation of First Passage Time Density Function for Jump-Diffusion Processes. *SIAM Journal on Scientific Computing*, 26(5):1760 – 1775, 2005.

[16] M. Atlan and B. Leblanc. Time-Changed Bessel Processes and Credit Risk. *Working Paper*, 2006.

[17] A.N. Avramidis and P. L'Ecuyer. Efficient Monte Carlo and Quasi-Monte Carlo Option Pricing under the Variance Gamma Model. *Management Science*, 52(12):1930–1944, 2006.

[18] A.N. Avramidis, P. L'Ecuyer, and P.A. Tremblay. Efficient Simulation of Gamma and Variance-Gamma Processes. *Proceedings of the 2003 Winter Simulation Conference*, pages 319–326, 2003.

[19] P. Baldi. Pricing General Barrier Options: A Numerical Approach using Sharp Large Deviations. *Mathematical Finance*, 9(4):292–322, 1999.

[20] P. Baldi, L. Caramellino, and M.G. Iovino. Pricing General Barrier Options: A Numerical Approach using Sharp Large Deviations. *Mathematical Finance*, 9(4):292–322, 1999.

[21] P. Baldi, L. Caramellino, and M.G. Iovino. *Pricing Complex Barrier Options with General Features Using Sharp Large Deviation Estimates*, pages 149–162. Springer, 1999.

[22] V. Baldi and G. Pagés. A Quantization Algorithm for Solving Multi-Dimensional Optimal Stopping Problem. *Working Paper*, 2000.

[23] V. Baldi and G. Pagés. A Quantization Algorithm for Solving Discrete Time Multi-Dimensional Discrete Time Optimal Stopping Problems. *Stochastic Processes and Their Applications*, 9:1003–1049, 2003.

[24] V. Baldi, G. Pagés, and P. Jacques. A Quantization Tree Method for Pricing and Hedging Mutidimensional American Options. *Mathematical Finance*, 15 (1):119–168, 2005.

[25] L.V. Ballestra and G. Pacelli. Pricing Double-Barrier Options using the Boundary Element Method. *Working Paper*, 2009. URL `http://ssrn.com/abstract=1492653`.

[26] O. Barndorff-Nielsen, P. Blaesild, and C. Halgreen. First Hitting Time Models for the Generalized Inverse Gaussian Distribution. *Stochastic Processes and their Applications*, 7:49–54, 1978.

[27] G. Barone-Adesi and R.E Whaley. Efficient Analytic Approximation of American Option Values. *Journal of Finance*, 42(2):301–319, 1987.

[28] G. Barone-Adesi, N. Fusari, and J. Theal. Barrier Option Pricing Using Adjusted Transition Probability. *Journal of Derivative*, 16:36–52, 2008.

[29] J. Barraquand and D. Martineau. Numerical Valuation of High Dimensional Multivariate American Securities. *Journal of Financial and Quantitative Analysis*, 30(383-405), 1995.

[30] D.R. Beaglehole, P.H. Dybvig, and G. Zhou. Going to Extremes: Correcting Simulation Bias in Exotic Option Valuation. *Financial Analyst Journal*, 53 (1):62, 1997.

[31] M. Becker. Unbiased Monte Carlo Valuation of Lookback, Swing and Barrier Options with Continuous Monitoring under Variance Gamma Models. *Journal of Computational Finance*, 13(4), 2010.

[32] D. Belomestny, G. Milstein, and V. Spokoiny. Regression Method in Pricing American and Bermudan Options using Consumption Process. *Quantitative Finance*, 9(3):315–327, 2009.

[33] C. Bernard, O. L. Courtois, and F. Q. Pinon. Pricing Derivatives with Barriers in a Stochastic Interest Rate Environment. *Journal of Economic Dynamics and Control*, 32(9), 2008.

[34] A. Beskos and G. Roberts. Exact Simulation of Diffusions. *The Annals of Applied Probability*, 15(4):2422–2444, 2005.

[35] C. Beveridge and M. Joshi. Practical Policy Iteration: Generic Methods for Obtaining Rapid and Tight Bounds for Bermudan Exotic Derivatives Using Monte Carlo Simulation. *Working Paper*, 2009. URL `http://ssrn.com/abstract=1331904`.

[36] P. Bjerksund and G. Stensland. Closed-form Valuation of American Options. *Working Paper*, NHH, 2002.

[37] P. Bjerksund and G. Stensland. Closed-Form Approximation of American Options. *Scandinavian Journal of Management*, (9):87–99, 1993.

[38] F. Black and M. Scholes. The Pricing of Options and Corporate Liabilities. *Journal of Political Economy*, 81:637–59, 1973.

[39] N. Bolia and S. Juneja. Function-Approximation-Based Perfect Control Variate for Pricing American Options. Proceedings of the Winter Simulation Conference, Institute of Electrical and Electronics Engineers, Piscataway, NJ, 2005.

[40] S. Boyarchenko and S. Levendorskiĭ. American Option in Regime Switching Model with Stochastic Interest Rates. Technical report, http://ssrn.com/abstract=1015409, 2007.

[41] S. Boyarchenko and S. Levendorskiĭ. Valuation of Continuously Monitored Double Barrier Options and Related Securities. *Mathematical Finance*, pages 1–26, 2010.

[42] P. Boyle and S. Lau. Bumping Up Against the Barrier with the Binomial Method. *Journal of Derivatives*, 1:6–14, 1994.

[43] P. Boyle and Y. Tian. An Explicit Finite Difference Approach to the Pricing of Barrier Options. *Applied Mathematical Finance*, 5:17–43, 1998.

[44] P. Boyle and Y. Tian. Pricing Lookback and Barrier Options under the CEV Processes. *Journal of Financial and Quantitative Analysis*, 34(2):241–264, 1999.

[45] L. Breiman. First Exit Times from the Square Root Boundary. In *Proceedings of the Berkeley Symposium on Mathematical Statistics and Probability*, pages 9–16, 1966.

[46] M.J. Brennan and E.S. Schwartz. Finite Difference Method and Jump Process Arising in the Pricing of Contingent Claims. *Journal of Financial and Quantitative Analysis*, 13:461–474, 1978.

[47] M. Broadie and M. Cao. Improved Lower and Upper Bound Algorithms for Pricing American Options by Simulation. *Quantitative Finance*, 8(8):845–861, 2008.

[48] M. Broadie and J. Detemple. American Option Valuation: New bounds, Approximations, and Comparison of Existing Methods. *Review of Financial Studies*, 9:1211–1250, 1996.

[49] M. Broadie and P. Glasserman. A Stochastic Mesh Method for Pricing High-Dimensional American Options. *Journal of Computational Finance*, 7(4):35–72, 2004.

[50] M. Broadie and P. Glasserman. Pricing American-Style Securities using Simulation. *Journal of Economic Dynamics and Control*, 21:1323–1352, 1997.

[51] M. Broadie, P. Glasserman, and S. Kou. Connecting Discrete and Continuous Path-Dependent Options. *Finance of Stochastics*, 3:55–82, 1999.

[52] P. Buchen and O. Konstanatos. A New Approach to Pricing Double-Barrier Options with Arbitrary Payoffs and Exponential Boundaries. *Applied Mathematical Finance*, 16(6):497–515, 2009.

[53] D.S. Bunch and H. Johnson. A Simple Numerical Efficient Valuation Method for American Put using a Modified Geske-Johnson Approach. *Journal of Finance*, 47, 1992.

[54] N. Cai, N. Chen, and X. Wan. Pricing Double-Barrier Options under a Flexible Jump Diffusion Model. *Operations Research Letters*, 37:163–167, 2009.

[55] L. Campi, S. Polbennikov, and A. Sbuelz. Systematic equity-based credit risk: A cev model with jump to default. *Journal of Economic Dynamics and Control*, 33:93–108, 2009.

[56] P. Carr. Randomization and the American Put. *The Review of Financial Studies*, 11(3):597–626, 1998.

[57] P. Carr and A. Hirsa. Why be Backward? *Risk*, 16(1):103–107, 2003.

[58] P Carr and A. Hirsa. *Forward Evolution Equations for Knock-Out Options*, page 41. Applied and Numerical Harmonic Analysis. Springer, 2007.

[59] P. Carr, R. Jarrow, and R. Myneni. Alternative Characterization of American Put Options. *Mathematical Finance*, 2:87–106, 1992.

[60] P. Carr, H. Geman, D.B. Madan, and M. Yor. The Fine Structure of Asset Returns: an Empirical Investigation. *Journal of Business*, 75(2):305–332, 2002.

[61] J. F. Carriere. Valuation of the Early-Exercise for Options Using Simulations and Nonparametric Regression. *Insurance:Mathematics and Economics*, 19 (19-30), 1996.

[62] C.C. Chang, S.L. Chung, and R.C. Stapleton. Richardson Extrapolation Techniques for the Pricing of American-Style Options. *Journal of Future Markets*, 27(8):791–817, 2007.

[63] S.K. Chaudhary. American Options and the LSM Algorithm: Quasi-Random Sequences and Brownian Bridge. *Journal of Computational Finance*, 8(4): 101–115, 2005.

[64] C. Chiarella and A. Ziogas. Pricing American Options on Jump-Diffusion Processes using Fourier Hermite Series Expansions. *Working Paper*, Quantitative Finance Research Centre, University of Technology Sydney, 2005.

[65] C. Chiarella and A. Ziogas. A Fourier Transform Analysis of the American Call Option on Assets Driven by Jump-Diffusion Processes. *Working Paper*, Quantitative Finance Research Centre, University of Technology, Sydney, 2006.

[66] C. Chiarella, C.B. Kang, G.H. Meyer, and A. Ziogas. The Evaluation of American Option Price under Stochastic Volatility and Jump Diffusion Dynamic using Method of Lines. *Computing in Economics and Finance*, 14, 2006.

[67] C. Chiarella, G.H. Meyer., and A. Ziogas. Pricing American Options under Stochastic Volatility. AMAMEF Workshop, Paris., 2006.

[68] P. Chirayukool and N.J. Webber. A Bermudan Control Variate for American Put Options. *Working Paper*, Warwick Business School, 2010.

[69] S.L. Chung. Pricing American Options on Foreign Assets in a Stochastic Interest Rate Economy. *Journal of Financial and Quantitative Analysis*, 37 (4):667–692, 2002.

[70] S.L. Chung. American Option Valuation under Stochastic Interest Rates. *Review of Derivatives Research*, (3):283–307, 1999.

[71] N. Clarke and K. Parrott. Multigrid for American Option Pricing with Stochastic Volatility. *Applied Mathematical Finance*, 6:177–195, 1999.

[72] E. Clement, D. Lamberton, and P. Protter. An Analysis of a Least Square Regression Method for American Option Pricing. *Finance and Stochastics*, 6: 449–471, 2002.

[73] R. Cont and P. Tankov. *Financial Modelling with Jump Processes*. Chapman and Hall/CRC Financial Mathematics Series, 2004.

[74] M. Costabile. Extending the Cox-Ross-Rubinstein Algorithm for Pricing Options with Exponential Boundaries. *Proceedings of Algoritmy Conference on Scientific Computing*, pages 23–32, 2002.

[75] J.C. Cox and S.A. Ross. The Valuation of Options for Alternative Stochastic Processes. *Journal of Financial Economics*, 3:145–166, 1976.

[76] J.C. Cox, S.A. Ross, and M. Rubinstein. Option Pricing: A Simplified Approach. *Journal of Financial Economics*, 7:229–263, 1979.

[77] C.W. Cryer. The Solution of a Quadratic Programme using Systematic Overrelaxation. *SIAM Journal of Control and Optimization*, 9:385–392, 1971.

[78] G. Dahlquist and A. Björck. *Numerical Methods*. Dover Publication, New Yorl, 1974.

[79] M. Dai and Y.K. Kwok. Knock-In American Options. *The Journal of Future Markets*, 24(2):179–192, 2004.

[80] T. S. Dai and Y. D. Lyuu. *An Efficient, and Fast Convergent Algorithm for Barrier Options*, pages 251–261. Applied and Numerical Harmonic Analysis. Springer, 2007.

[81] H. E. Daniels. The First Crossing-Time Density for Brownian Motion with a Perturbed Linear Boundary. *Bernoulli*, pages 571–580, 2000.

[82] D. Davydov and V. Linetsky. Structuring, Pricing and Hedging Double-Barrier Step Options. *Journal of Computational Finance*, 5(2):55–87, 2002.

[83] M.A.H. Dempster, J.P. Hutton, and D.G. Richards. LP Valuation of Exotic American Option Exploiting Structure. *Journal of Computational Finance*, 2: 61–84, 1998.

[84] E. Derman, I. Kani, and I. Ergener, D.and Bardhan. Enhanced Numerical Methods for Options with Barriers. *Financial Analysts Journal*, 51(6):65–74, 1995.

[85] J. Detemple. *American-Style Derivatives: Valuation and Computation.* Chapman and Hall, 2006.

[86] Y. d'Halluin, P.A. Forsyth, and G. Labahn. A Penalty Method for American Options with Jump Processes. *Working Paper*, 2003.

[87] G. Dorfleitner, P. Schneider, K. Hawlitschek, and A. Buch. Pricing Options with Greens Functions when Volatility, Interest Rate and Barriers Depend on Time. *Quantitative Finance*, (8):119–133, 2008.

[88] J. Durbin and D. Williams. The First-Passage Density of the Brownian Motion Process to a Curved Boundary. *Journal o f Applied Probability*, 29(2):291–304, 1992.

[89] S.M.T. Ehrlichman and S.G. Henderson. Adaptive Control Variate for Pricing Multi-Dimensional American Options. *Journal of Computational Finance*, 11 (1):65–91, 2007.

[90] A. Esser. Derivatives Written on a Power of the Stock Price: General Valuation Principles and Application to Stochastic Volatility. *Working Paper*, School of Business and Economics, Goethe University, 2003.

[91] S. Figlewski and B. Gao. The Adaptive Mesh Model: a New Approach to Efficient Option Pricing. *Journal of Financial Economics*, 53:313–351, 1999.

[92] T.J. Finuance and M.J. Tomas. American Stochastic Volatility Call Option Pricing: a Lattice Based Approach. *Review of Derivative Research*, 1:183–201, 1996.

[93] G. Foufas and M.G. Larson. Valuing European, Barrier, and Lookback Options using the Finite Element Method and Duality Techniques. *Working Paper*, 2004. URL `http://citeseerx.ist.psu.edu`.

[94] J.P. Fouque and C.H. Han. Variance Reduction for Monte Carlo Methods to Evaluate Option Prices under Multi-Factor Stocastic Volatility Models. *Quantitative Finance*, 4(5):597–606, 2004.

[95] J.P. Fouque and C.H. Han. A Control Variate Method to Evaluate Option Prices under Multi-Factor Stochastic Volatility Model. *Working Paper*, 2004.

[96] J.P. Fouque and C.H. Han. A Martingale Control Variate Method for Option Pricing with Stochastic Volatility. ESAIM: Probability and Statistics, 2006.

[97] J.H. Friedman. Multivariate Adaptive Regression Splines. *Annals of Statistics*, 19(1):1–141, 1991.

[98] G. Fusai and M.C. Recchioni. Analysis of Quadrature Methods for Pricing Discrete Barrier Options. *Journal of Economic Dynamics Control*, 5:826–860, 2007.

[99] A. Gamma. *Real Option Valuation: a Monte Carlo Approach in Real Option.* Elsevier, 2003.

[100] B. Gao, J.Z. Haung, and M. Subrahmanyam. The Valuation of American Barrier Options using the Decomposition Technique. *Journal of Economic Dynamics Control*, 24:1783–1827, 2000.

[101] H. Gerber and E. Shiu. Martingale Approach to Pricing Perpetual American Options. *ASTIN BULLETIN*, 24:195–220, 1994.

[102] H.U Gerber and E.S.W. Shiu. Martingale Approach to Pricing Perpetual American Options on Two Stocks. *Mathematical Finance*, 6:303–322, 1996.

[103] R. Geske. The Valuation of Compound Options. *Journal of Financial Economics*, 7:63–81, 1979.

[104] R. Geske and H. Johnson. The American Put Option Valued Analytically. *Journal of Finance*, 30:1511–1524, 1984.

[105] M.B. Giles. *Improved Multilevel Monte Carlo Convergence using the Milstein Scheme*, pages 343–358. Springer, 2007.

[106] M.B. Giles. Multi-Level Monte Carlo Path Simulation. *Operations Research*, 56(3):607–617, 2008.

[107] M.B. Giles. Multilevel Monte Carlo for Basket Options. Proceedings of the 2009 Winter Simulation Conference, 2009.

[108] M.B. Giles and B.J. Waterhouse. *Multilevel quasi-Monte Carlo path simulation*, pages 165–181. Springer, 2009.

[109] M.B. Giles, D.J. Higham, and X. Mao. Analysing Multi-level Monte Carlo for Options with Non-Globally Lipschitz Payoff. *Finance and Stochastics*, 13(3): 403–413, 2009.

[110] P. Glasserman. *Monte Carlo Method in Financial Engineering*. Springer, 2004.

[111] P. Glasserman and J. Staum. Conditioning on One-Step Survival for Barrier Option Simulations. *Operations Research*, 49(6):923–937, 2001.

[112] P. Glasserman and B. Yu. *Simulation for American Options: Regression now or Regression Later?*, pages 213–226. Springer, Berlin, 2004.

[113] P. Glasserman and B. Yu. Number of Path Versus Number of Basis Functions in American Option Pricing. *The Annals of Applied Probability*, 14(4):2090–2119, 2004.

[114] A. Göing-Jaeschke and M. Yor. A Clarification Note about Hitting Times Densities for Ornstein-Uhlenbeck Processes. *Finance and Stochastics*, 7:413–415, 2003.

[115] I.S. Gradshteyn and I.M. Ryzhik. *Table of Integrals, Series, and Products*. Elsevier, seventh edition, 2007.

[116] G.D. Graziano and L.C.G. Rogers. Barrier Option Pricing for Assets with Markov-Modulated dividends. *Journal of Computational Finance*, 9(4):75–87, 2006.

[117] S.A. Griebsch and U. Wystup. On the Valuation of Fader and Discrete Barrier Options in Heston's Stochastic Volatility Model. *Quantiative Finance*, 11(5): 693–709, 2011.

[118] L.K. Guan and G. Xiaoqiang. Pricing American Options with Stochastic Volatility: Evidence from SP 500 Future Options. *The Journal of Future Markets*, 20(7):625–659, 2000.

[119] E.G. Haug. Closed Form Valuation of American Barrier Options. *International Journal of Theoretical and Applied Finance*, 4:355–359, 2001.

[120] M. Haugh and L. Kogan. Pricing American Options: a Duality Approach. *Operational Research*, 52(2):258–270, 2004.

[121] S.G. Henderson and P.W. Glynn. Approximating Martingales for Variance Reduction in Markov Process Simulation. *Mathematics of Operations Research*, 27(2):253–271, 2002.

[122] S.L. Heston. A Closed-Form Solution for Options with Stochastic Volatility with Application in Bond and Currency Options. *The Review of Financial Studies*, 6(2):327–343, 1993.

[123] A. Hirsa and D.B. Madan. Pricing American Options under Variance Gamma. *Journal of Computational Finance*, 7(2):63–80, 2004.

[124] T.C. Ho, R.C. Stapleton, and M.G. Subrahmanyam. The Valuation of American Options with Stochastic Interest Rates: A Generalization of the Geske-Johnson Technique. *The Journal of Finance*, 52(2):827–840, 1997.

[125] D. G. Hobson, D. Williams, and A. T. A. Wood. Taylor Expansions of Curve-Crossing Probabilities. *Bernoulli*, 5(5):779–795, 1999.

[126] P. Hörfelt. Extension of the Corrected Barrier Approximation by Broadie, Glasserman, and Kou. *Finance and Stochastics*, 7:231–243, 2003.

[127] C.H. Hui. Time-Dependent Barrier Option Values. *Journal of Futures Markets*, 17(6):667–688, 1997.

[128] J.C. Hull and A. White. The Pricing of Option on Assets with Stochastic Volatilities. *Journal of Finance*, 42(2):281–300, 1987.

[129] H. Hulley and E. Platen. Laplace Transform Identities for Diffusions, with Applications to Rebates and Barrier Options. *Working Paper*, Quantitative Finance Research Centre, University of Technology Sydney, 2007.

[130] A. Ibánez and C. Velasco. The Optimal Method for Pricing Bermudan Options by Simulation. *Working Paper*, Universidad Carlos III de Madrid-Department of Economics, 2010. URL `http://ssrn.com/abstract=1507994`.

[131] S. Ikonen and J. Toivanen. Operator Splitting Methods for American Options with Stochastic Volatility. *European Congress on Computational Methods in Applied Sciences and Engineering*, 2004.

[132] S. Ikonen and J. Toivanen. Componentwise Splitting Methods for Pricing American Options under Stochastic Volatility. *Working Paper*, Department of Mathematical Information Technology, University of Jyvaskyla, Finland, 2006. URL `http://www.finactu.fi/backGround/compsplit.pdf`.

[133] P. Imkeller and I. Pavlyukevich. First Exit Times of SDEs Driven by Stable Lévy Processes. *Stochastic Processes and their Applications*, 116:611–642, 2006.

[134] J. E. Ingersoll. Approximating American Options and Other Financial Contracts using Barrier Derivatives. *Journal of Computational Finance*, 2(1): 85–112, 1998.

[135] S. Iyengar. Hitting Lines with Two-Dimensional Brownian Motion. *SIAM Journal on Applied Mathematics*, 45(6):983–989, 1985.

[136] S. Jacka. Optimal Stopping and American Put. *Mathematical Finance*, 1: 1–14, 1991.

[137] J. James and N.J. Webber. *Interest Rate Modelling.* Wiley Series in Finance, first edition, 2009.

[138] F. Jamshidian. An Analysis of American Options. *Review of Futures Markets*, 11(72-80), 1992.

[139] M. Jeannin and M. Pistorius. A Transform Approach to Compute Prices and Greeks of Barrier Options Driven by a Class of lévy Process. *Quantitative Finance*, 18:629–644, 2010.

[140] C. Jennen and H. R. Lerche. First Exit Densities of Brownian Motion Through One-Sided Moving Boundaries. *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, 55:133–148, 1981.

[141] L. Jiang and M. Dai. Convergence of Binomial Tree Method for American Options. *Partial Differential Equations and their Application*, pages 106–118, 1999.

[142] X. Jin, H.H. Tan, and J. Sun. A State-Space Partitioning Method for Pricing High-Dimensional American-Style Options. *Mathematical Finance*, 17(3):399–426, 2007.

[143] H. Johnson. An Analytical Approximation for the American Put Price. *Journal of Financial and Quantitative Analysis*, 18:141–148, 1983.

[144] C. Jonen. An Efficient Implementation of a Least Squares Monte Carlo Method for Valuing American-Style Options. *International Journal of Computer Mathematics*, 86(6):1024–1039, 2009.

[145] M. Joshi and T. S. Leung. Using Monte Carlo simulation and Importance Sampling to Rapidly Obtain Jump-Diffusion Prices of Continuous Barrier Options. *Journal of Computational Finance*, 10(4):137–150, 2007.

[146] M. Joshi and R. Tang. Pricing and Deltas of Discretely-Monitored Barrier Options Using Stratified Sampling on the Hitting-Times to the Barrier. *International Journal of Theoretical and Applied Finance*, 2010.

[147] N. Ju. Pricing an American Option by Approximating Its Early Exercise Boundary as a Multipiece Exponential Function. *Review of Financial Studies*, 11:627–646, 1998.

[148] H. Juneja and S. Kalra. Variance Reduction Techniques for Pricing American Options using Function Approximations. *Journal of Computational Finance*, 12(3):79–102, 2009.

[149] N. Kahale. Analytical Crossing Probabilities for Certain Barriers by Brownian Motion. *The Annals of Applied Probability*, 18(4):1424–1440, 2008.

[150] S. Kallast and A. Kivinukk. Pricing and Hedging American Options using Approximations by Kim Integral Equations. *European Finance Review*, 7: 361–383, 2003.

[151] I. Karatzas and S.E. Shreve. *Brownian Motion and Stochastic Calculus.* Springer, 1991.

[152] A. Kebaier. Statistical Romberg Extrapolation: A New Variance Reduction Method and Applications to Options Pricing. *Annals of Applied Probability*, 3:2681–2705, 1990.

[153] E. Kellezi and N.J. Webber. Valuing Bermudan Options when Asset Returns are Lévy Processes. *Quantitative Finance*, 4(1):87–100, 2004.

[154] J. Kent. Eigenvalue Expansions for Diffusion Hitting Times. *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, 52:309–319, 1980.

[155] J. Kent. The Spectral Decomposition of a Diffusion Hitting Time. *The Annals of Probability*, 10(1):207–219, 1982.

[156] M. Kijima and T. Suzuki. The Pricing of Options with Stochastic Boundaries in a Gaussian Economy. *Journal of Operational Research Society of Japan*, 50 (2):137–150, 2007.

[157] I. Kim. The Aalytic Valuation of American Options. *Review of Financial Studies*, 3:547–572, 1990.

[158] S. Kim and S.G. Henderson. Adaptive Control Variates for Finite-Horizon Simulation. *Mathematics of Operations Research*, 32(3):508–527, 2007.

[159] S. G. Kou and H. Wang. First Passage Times of a Jump Diffusion Process. *Advances in Applied Probability*, 35(2):504–531, 2003.

[160] G. Kuan and N. Webber. Valuing Continuous Barrier Options on a Lattice Solution for a Stochastic Dirichlet Problem. *Working Paper*, Cass Business School, City University, 2003.

[161] N. Kunitomo and M. Ikeda. Correction: Pricing Options with Curve Boundaries. *Mathematical Finance*, 10(4):459, 2000.

[162] N. Kunitomo and M. Ikeda. Pricing Options with Curve Boundaries. *Mathematical Finance*, 2(4):275–298, 1992.

[163] B. Leblanc, O. Renault, and O. Scaillet. A Correction Note on the First Passage Time of an Ornstein-Uhlenbeck Process to a B3oundary. *Finance and Stochastics*, 4:109–111, 2000.

[164] H.R. Lerche. *Boundary Crossing of Brownian Motion.* Springer-Verlag, 1986.

[165] S. Levendorskiĭ, O. Kudryavtsev, and V. Zherder. The Relative Efficiency of Numerical Methods for Pricing American Options under Lévy Processes. *Journal of Computational Finance*, 9(2):69–97, 2005.

[166] X.S. Lin. Double Barrier Hitting Time Distributions with Applications to Exotic Options. *Insurance: Mathematics and Economics*, 23:45–58, 1998.

[167] S. Lindset and A.C. Lund. A Monte Carlo Approach for the American Put under Stochastic Interest Rates. *Journal of Economic Dynamics and Control*, (31):1081–1105, 2007.

[168] V. Linetsky. Computing Hitting Time Densities for CIR and OU Diffusions: Applications to Mean Reverting Models. *Journal of Computational Finance*, 7:1–22, 2004.

[169] V. Linetsky. Lookback Options and Diffusion Hitting Times: A Spectral Expansion Approach. *Finance and Stochastics*, 8:373–398, 2004.

[170] C. F. Lo and C. H. Hui. Lie Algebraic Approach for Pricing Moving Barrier Options with Time-Dependent Parameters. *Journal of Mathematical Analysis and Applications*, 323(2):1455–1464, 2006.

[171] C. F. Lo and C. H. Hui. Computing the First Passage Time Density of a Time-Dependent OrnsteinUhlenbeck Process to a Moving Boundary. *Applied Mathematics Letters*, 19:1399–1405, 2006.

[172] C. F. Lo, P. H. Yuen, and C. H. Hui. Pricing Barrier Options with Square Root Process. *International Journal of Theoretical and Applied Finance*, 4(5): 805–818, 2001.

[173] C. F. Lo, H. C. Lee, and C. H. Hui. A Simple Approach for Pricing Barrier Options with Time-Dependent Parameters. *Quantitative Finance*, 3(2):98–107, 2003.

[174] C. F. Lo, H. M. Tang, K. C. Ku, and C. H. Hui. Valuing Time-Dependent CEV Barrier Options. *Journal of Applied Mathematics and Decision Sciences*, pages 1–17, 2009.

[175] C.F. Lo, P.H. Yuen, and C.H. Hui. Constant Elasticity of Variance Option Pricing Model with Time-Dependent Parameters. *International Journal of Theoretical and Applied Finance*, 4:661–674, 2000.

[176] F. Longstaff and E. Schwartz. Valuing American Options by Simulation: A Simple Least-Square Approach. *The Review of Financial Studies*, 2001.

[177] F. Longstaff and E.S. Schwartz. Valuing Risky Debt: A New Approach. *Journal of Finance*, 50(3):789–820, 1995.

[178] L. MacMillan. An Analytical Approximation for the American Put Price. *Advance in Futures and Options Research*, 1:119–139, 1986.

[179] D.B. Madan, P. Carr, and E.C. Chang. The Variance Gamma Process and Option Pricing. *European Finance Review*, 2:79–105, 1998.

[180] A.M. Matache, P.A. Matache, and C. Schwab. Wavelet Galerkin Pricing of American Options on Lévy Driven Asset. *Quantitative Finance*, 5(4):403–424, 2005.

[181] H. McKean. Appendix: Free Boundary Problem for the Heat Equation Arising from a Problem in Mathematical Economics. *Industrial Management Review*, 6:32–39, 1965.

[182] R. Merton. The Theory of Rational Option Pricing. *Bell Journal of Economics and Management Science*, 4:141–183, 1973.

[183] R.C. Merton. Option Pricing when Underlying Stock Returns are Discontinuous. *Journal of Financial Economics*, 3:125–144, 1976.

[184] S.A.K. Metwally and A.F. Atiya. Using Brownian Bridge for Fast Simulation of Jump-Diffusion Processes and Barrier Options. *Journal of Derivative*, pages 43–54, 2002.

[185] G.H. Meyer and J. van der Hoek. The Evaluation of American Options with the Method of Lines. *Advance in Futures and Options Research*, 9:265–285, 1997.

[186] J.R. Michale, W.R. Schucany, and R.W. Haas. Generating Random Variates using Transformations witih Multiple Roots. *The American Statistician*, 30 (2):88–90, 1976.

[187] A. Mijatović. Local Time and the Pricing of Time-Dependent Barrier Options. *Finance and Stochastics*, 14:13–48, 2010.

[188] E. Mordecki. Optimal Stopping and Perpetual Options for lévy Processes. *Finance and Stochastics*, 6:473–493, 2002.

[189] M. Moreno and J. F. Navas. On the Robustness of Least-Squares Monte Carlo (LSM) for Pricing American Derivatives. Technical report, Universitat Pompeu Fabra, 2003.

[190] M. Morimoto and H. Takahashi. On Pricing Exponential Square Root Barrier Knockout European Options. *Asia - Pacific Financial Markets*, 9(1):1–21, 2002.

[191] J. Ndogmo and D. B. Ntwiga. High-Order Accurate Implicit Methods for the Pricing of Barrier Options. *Working Paper*, 2007. URL `http://arxiv.org/pdf/0710.0069`.

[192] A. Novikov. Martingale and First-Passage Times for Ornstein-Uhlenbeck Processes with a Jump Component. *Theory of Probability and its Applications*, 48(2):288–303, 2004.

[193] A. Novikov, V. Frishling, and N. Kordzakhia. Approximations of Boundary Crossing Probabilities for a Brownian Motion. *Journal of Applied Probability*, 36(4):1019–1030, 1999.

[194] A. Novikov, V. Frishling, and N. Kordzakhia. Time-Dependent Barrier Options and Boundary Crossing Probabilities. *Georgian Mathematical Journal*, 10:325–334, 2003.

[195] E. Omberg. A Note on the Convergence of the Binomial Pricing and Compound Option Models. *Journal of Finance*, 42:1019–1030, 1987.

[196] C. Park and S. R. Paranjape. Evaluations of Barrier-Crossing Probabilities of Wiener Paths. *Journal of Applied Probability*, 13(2):267–275, 1976.

[197] A. Pelsser. Pricing Double Barrier Options using Laplace Transforms. *Finance and Stochastics*, 4:95–104, 2000.

[198] J. Pitman and M. Yor. Hitting, Occupation and Inverse Local Times of one-dimensional Diffusions: Martingale and Excursion Approaches. *Bernoulli*, 9 (1):1–24, 2003.

[199] J. Pitman and M. Yor. A Decomposition of Bessel Bridge. *Z. Wahrscheinlichkeitstheorie verw. Gebiete*, 59:425–457, 1982.

[200] A. PréKopa and T. Sźantai. On the Analytical-Numerical Valuation of the Bermudan and American Options. *Quantitative Finance*, 10(1):152–164, 2010.

[201] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes: The Art of Scientific Computing*. Cambridge University Press, 2007.

[202] F. Rapisarda. Pricing Barriers on Underlyings with Time-Dependent Parameters. *Working Paper*, 2005. URL `http://papers.ssrn.com/sol3/papers.cfm?abstract_id=682184`.

[203] N.S. Rasmussen. Control Variate for Monte Carlo Valuation of American Options. *Journal of Computational Finance*, 9(1):83–118, 2005.

[204] S.B. Raymar and M.J. Zwecher. Monte Carlo Estimation of American Call Options on Maximum of Several Stocks. *Journal of Derivative*, 5(1):7–23, 1997.

[205] E. Reiner and M. Rubinstein. Breaking Down the Barriers. *Risk*, 4(8):28–35, 1991.

[206] C. Ribeiro and N.J. Webber. Valuing Path-Dependent Options in the Variance-Gamma Model by Monte Carlo with a Gamma Bridge. *Journal of Computational Finance*, 7(2):81–100, 2003.

[207] C. Ribeiro and N.J. Webber. A Monte Carlo Method for the Normal Inverse Gaussian Option Valuation Model using an Inverse Gaussian Bridge. *Working Paper*, Cass Busness School, City University, 2004.

[208] C. Ribeiro and N.J. Webber. Correcting for Simulation Bias in Monte Carlo Methods to Value Exotic Options in Models Driven by Lévy Processes. *Applied Mathematical Finance*, 13(4):333–352, 2006.

[209] L.M. Ricciardi and S. Sato. First Passage Time Density and Moments of the Ornstein-Uhlenbeck Process. *Journal of Applied Probability*, 25(1):43–57, 1988.

[210] L.M. Ricciardi, L. Sacerdote, and S. Sato. On an Integral Equation for First Passage Time Probability Densities. *Journal of Applied Probability*, 21(2): 302–314, 1984.

[211] D.R. Rich. The Mathematical Foundations of Barrier Option Pricing Theory. *Advances in Futures and Options Research*, 7:267–311, 1994.

[212] P. Ritchken. On Pricing Barrier Options. *Journal of Derivatives*, (3):19–28, 1995.

[213] G.O. Roberts and C.F. Shortland. The Hazard Rate Tangent Approximation for Boundary Hitting Times. *The Annals of Applied Probability*, 5(2):446–460, 1995.

[214] G.O. Roberts and C.F. Shortland. Pricing Barrier Options with Time-Dependent coefficients. *Mathematical Finance*, 7(1):83–93, 1997.

[215] A. Rodriques and M.J.R. Armada. The Valuation of Real Options with the Least Squares Monte Carlo Simulation Method. *Working Paper*, 2006.

[216] L.C.G. Roger. Monte Carlo Valuation of American Options. *Mathematical Finance*, 12(3):271–286, 2002.

[217] L. Rogers and E. Stapleton. Fast Accurate Binomial Pricing. *Finance and Stochastics*, 2:3–17, 1998.

[218] L.C.G. Rogers and O. Zane. Valuing Moving Barrier Options. *Journal of Computational Finance*, 1(1):5–11, 1997.

[219] B. Roynette, P. Vallois, and A. Volpi. Asymptotic Behavior of the Hitting Time, Overshoot and Undershoot for Some Lévy Processes. *ESAIM Probability and Statistics*, 12:58–97, 2008.

[220] P. Salminen. On the First Hitting Time and the Last Exit Time for a Brownian Motion to/from a Moving Boundary. *Advances in Applied Probabilty*, 20(2): 411–426, 1988.

[221] P. Salminen, P. Vallois, and M. Yor. On the Excursion Theory for Linear Diffusions. *Japan Journal of Mathematics*, 2:97–127, 2007.

[222] S. Sato. Evaluation of the First-Passage Time Probability to a Square Root Boundary for the Wiener Process. *Journal of Applied Probability*, 14(4):850–856, 1977.

[223] E.S. Schwartz. The Valuation of Warrants: Implementing a New Approach. *Journal of Financial Economic*, 5:79–93, 1977.

[224] V. Seshadri. *The Inverse Gaussian Distribution: A Case Study in Exponential Families*. Oxford Science Publication, 1993.

[225] P.V. Shevchenko. Addressing the Bias in Monte Carlo Pricing of Multi-Asset options with Multiple Barriers Through Discrete Sampling. *Journal of Computational Finance*, 6(3):1–20, 2003.

[226] J. Sidenius. Double Barrier Options: Valuation by Path Counting. *Journal of Computational Finance*, 1(3):63–79, 1998.

[227] D. Siegmund and Y. S. Yuh. Brownian Approximations to First Passage Probabilities. *Z. Wahrscheinlichkeitstheorieverw. Gebiete*, 59:239–248, 1982.

[228] A. Speight. A Multilevel Approach to Control Variates. *Journal of Computational Finance*, 12(4):3–27, 2009.

[229] M. Steiner, M. Wallmeier, and R. Hafner. Pricing Near the Barrier: the Case of Discrete Knock-Out Options. *Journal of Computational Finance*, 3(1):69–90, 1999.

[230] L. Stentoft. Convergence of the Least Square Monte Carlo Approach to American Option Valuation. *Management Science*, 50(9):1193, 1203 2004.

[231] L. Stentoft. Assessing the Least Square Monte-Carlo Approach to American Option Valuation. *Review of Derivative Research*, 7(129-168), 2004.

[232] L. Stentoft. Value Function Approximation: A Comparison of Two Recent Numerical Methods for American Option Pricing using Simulation and Regression. *Working Paper*, 2008. URL `http://ssrn.com/abstract=1315306`.

[233] M.A. Sullivan. Pricing Discretely Monitored Barrier Options. *Journal of Computational Finance*, 3(4):35–52, 2000.

[234] D. Tavella and C. Randall. *Pricing Financial Instruments: The Finite Difference Method.* John Wiley Sons, 2000.

[235] G.W.P. Thompson. Bounds on the Value of Barrier Options with Curved Boundaries. *Working Paper*, Judge Institute of Management, University of Cambridge, 2002.

[236] J. Tilley. Valuing American Options in a Path Simulation Model. *Transactions of the Society of Actuaries*, 45:84–104, 1993.

[237] A. Trolle and E.S. Schwartz. A General Stochastic Volatility Model for the Pricing of Interest Rate Derivatives. *Review of Financial Studies*, 22:2007–2057, 2009.

[238] J.N. Tsitsiklis and B. Van Roy. Regression Methods for Pricing Complex American-Style Options. *IEEE Transactions on Neural Networks*, 12:694–703, 2000.

[239] J.N. Tsitsiklis and B. Van Roy. Optimal Stopping of Markov Processes: Hilbert Space Theory, Approximation Algorithms, and an Application to Pricing High-Dimensional Financial Derivatives. *IEEE Transactions on Automatic Control*, 44(10):1840–1851, 1999.

[240] O.A. Vasicek. An Equilibrium Characterization of the Term Structure. *Journal of Financial Economics*, (5):177–188, 1977.

[241] M. Vellekoop and H. Kieuwenhuis. A Tree-Based Method to Price American Options in the Heston Model. *Journal of Computational Finance*, 13(1):1–21, 2009.

[242] B.A. Wade, A.Q.M. Khaliq, M. Yousuf, J. Vigo-Aquiar, and R. Deininger. On Smoothing of the CrankNicolson Scheme and Higher Order Schemes for Pricing Barrier Options. *Journal of Computational and Applied Mathematics*, 204:144–158, 2007.

[243] I.R. Wang, J.W.L. Wan, and P.A. Forsyth. Robust Numerical Valuation of European and American Options under the CGMY Process. *Journal of Computational Finance*, 10(4):31–69, 2007.

[244] L. Wang and K. Pötzelberger. Boundary Crossing Probability for Brownian Motion and General Boundaries. *Journal of Applied Probability*, 34(1):341–352, 1997.

[245] M.L. Wang, Y. Liu, and Y.L. Hsiao. Barrier Options Pricing: A Hybrid Method Approach. *Quantitative Finance*, 9(3):341–352, 2009.

[246] Y. Wang and R. Caflisch. Pricing and Hedging American-Style Options: A Simple Simulation-Based Approach. *Journal of Computational Finance*, 13 (4), 2010.

[247] N.J. Webber. *Implementing Models of Financial Derivatives: Object Oriented Applications with VBA*. Wiley Finance, 1 edition, 2011.

[248] C. Wöster. An Efficient Algorithm for Pricing Barrier Options in Arbitrage-Free Binomial Models with Calibrated Drift Terms. *Quantitative Finance*, 10 (5):555–564, 2010.

[249] C. Yi. On the First Passage Time Distribution of an OrnsteinUhlenbeck Process. *Quantitative Finance*, 10(9):957–960, 2010.

[250] D. Zhang and R. V. N. Melnik. First Passage Time for Multivariate Jump-Diffusion Stochastic Models with Application in Finance. *Working Paper*, 2007.

[251] X.L. Zhang. Numerical Analysis of American Option Pricing in a Jump-Diffusion Model. *Mathematics of Operations Research*, 22(3):668–690, 1997.

[252] R. Zvan, P.A. Forsyth, and K.R. Vetzal. Penalty Methods for American Options with Stochastic Volatility. *Journal of Computational and Applied Mathematics*, 91:199–218, 1998.

[253] R. Zvan, P. A. Forsyth, and K. R. Vetzal. PDE Methods for Pricing Barrier Options. *Journal of Economic Dynamics and Control*, 24(11):1563–1590, 2000.