



Norwegian University of
Science and Technology

Pricing Exotic Options with the Normal Inverse Gaussian Market Model using Numerical Path Integration

Karsten Krog Sæbø

Master of Science in Physics and Mathematics

Submission date: June 2009

Supervisor: Arvid Næss, MATH

Problem Description

Compare the Normal Inverse Gaussian market model against empirical financial market data, and price exotic options using the numerical path integration approach.

Assignment given: 09. February 2009
Supervisor: Arvid Næss, MATH

Preface

This master thesis is submitted as a conclusion of the Master of Physics and Mathematics degree at the Norwegian University of Science and Technology (NTNU), with specialization in Industrial Mathematics. The thesis work was performed during the spring semester 2009 under supervision of Professor Arvid Næss at the Institute of Mathematical Sciences. Working with a problem on the borderline between mathematics and finance have been most rewarding for me, and I would like to thank Professor Arvid Næss for excellent supervision.

Haugesund, June 2009

Karsten Krog Sæbø

Abstract

In this thesis we show in detail how the Normal Inverse Gaussian process can be used to model stocks and stock market indices. We explore the properties of the NIG market model in comparison to empirical findings in the financial markets. Then we will demonstrate how to price exotic options under this NIG market model using numerical path integration. Finally, we perform numerous numerical examples to show the efficiency and accuracy of this implementation approach. Since prices on exotic options are not publicly available, we will price the options using some fixed parameters, and compare with Monte Carlo simulation results.

Contents

1	Introduction	1
1.1	Thesis Outline	1
2	Financial Basics	3
2.1	Stocks and Stock Market Indices	3
2.2	The Black-Scholes Model Dynamics	4
2.3	Options	6
2.4	Numerical Option Pricing Methods	9
3	The Normal Inverse Gaussian Process	13
3.1	The Effects of changing the Different Parameters	15
3.2	NIG Market Model	18
4	Market Calibration of Parameters	21
4.1	Maximum Likelihood Estimator	21
4.2	Historical Parameter Estimation	22
4.3	Goodness of Fit	23
4.4	Implied Parameter Estimation	28
4.5	Historical vs. Implied	35
4.6	Sensitivity Analysis	37
5	Option Pricing and Numerical Path Integration	41
5.1	Model	41
5.2	Implementation	44
5.2.1	Lookback Options	45
5.2.2	Asian Options	48
5.3	Monte Carlo Simulation	50
6	Numerical Results	53
6.1	Barrier Options	54
6.2	Lookback Options	55
6.2.1	Fixed Strike Lookback Options	55
6.2.2	Floating Strike Lookback Options	59
6.3	Asian Options	61

6.3.1 Error Analysis	65
A Source Code	71
B Option Prices	85

List of Tables

3.1	Central moments of NIG distribution	17
3.2	Parameter effects on the central moments.	17
4.1	Data set description	23
4.2	Estimated NIG and Black-Scholes (normal) parameters	23
4.3	Quantiles of the Kolmogorov distribution for different values of α^*	27
4.4	Values of the Kolmogorov-Smirnov test statistic	28
4.5	Number of call prices inside bid-ask spread	30
4.6	Implied parameters for OBX option prices	31
4.7	Implied parameters for Dow Jones Industrial Average option prices	33
4.8	Implied parameters for Coca Cola option prices	34
4.9	Average bid-ask spread and average mid price for the market prices	34
4.10	Implied parameters for StatoilHydro option prices	35
5.1	Payoff for fixed and floating strike lookback options	42
5.2	Payoff for fixed and floating strike Asian options	42
6.1	Fixed input parameters	53
6.2	Down-and-out-up-and-out call prices for different barrier levels	54
6.3	Up-and-out call prices for different barrier levels	55
6.4	Price of a fixed strike lookback call option with varying gridsize	56
6.5	Fixed strike lookback call prices for different strike prices	57
6.6	Price of a fixed strike lookback put option with varying gridsize	57
6.7	Fixed strike lookback put prices for different strike prices	58
6.8	Price of a floating strike lookback call option with varying gridsize	59
6.9	Price of a floating strike lookback put option with varying gridsize	60
6.10	Floating lookback call option prices with varying dt and S_0	60
6.11	Floating lookback put option prices with varying dt and S_0	61
6.12	Fixed input parameters for Asian options	62
6.13	Prices for fixed strike Asian call option for different values of dt and X	62
6.14	Asian option prices with PI1 for Set 1	63
6.15	Asian option prices with PI2 for Set 1	63
6.16	Asian option prices with PI1 for Set 2	64

6.17	Asian option prices with PI2 for Set 2	64
6.18	Asian option prices with PI1 for Set 1 with $dt = \frac{1}{50}$	65
6.19	Asian option prices with PI1 for Set 2 with $dt = \frac{1}{50}$	65
B.1	OBX Call Option prices	86
B.2	DJI Call Option prices	90
B.3	KO Call Option prices	92
B.4	STL Call Option prices	93

List of Figures

2.1	Histogram of daily log returns on Dow Jones IA	5
2.2	Hedging example for American Airlines	7
3.1	Inverse Gaussian PDF	14
3.2	A NIG sample path	15
3.3	Effect on the NIG distributed PDF when parameters are changed	16
3.4	Risk-neutral density for Black-Scholes and NIG	19
4.1	Normalized histograms and the corresponding PDFs	24
4.2	QQ-plots for daily log return	25
4.3	Normalized histogram for StatoilHydro with only 30 bins	26
4.4	OBX call option prices	32
4.5	Dow Jones IA call option prices	33
4.6	Coca Cola call option prices with historical parameters	36
4.7	Graphical visualization of $\frac{\partial C}{\partial \alpha}$	38
4.8	Graphical visualization of $\frac{\partial C}{\partial \beta}$	39
4.9	Graphical visualization of $\frac{\partial C}{\partial \delta}$	40
5.1	A realization of the CDF $F_{M_N}(\eta)$ and the corresponding $f_{M_N}(\eta)$	46
6.1	Joint PDF for parameter set 2 with PI1	66

Chapter 1

Introduction

Through decades, the Black-Scholes model (Black & Scholes 1973) has dominated the world of financial derivatives pricing, mainly due to its simplicity. However, because of several explicit simplifying assumptions needed to be made, the model disagrees with reality in a number of ways, some significant. In particular, the Black-Scholes model assumes that the logarithm of the daily stock returns are normally distributed. The impact of this assumption has become especially noticeable through the last year, as it tends to underestimate the probability of extreme daily variations. Due to such shortfalls of the Black-Scholes model, financial modelers are constantly trying to find more realistic models for the dynamics of financial instruments. Stochastic volatility and jump-diffusion models are frequently referred to in the literature, see e.g. (Rebonato 2004). Different variations of Levy process have also been attempted as modeling tool, see e.g. (Schoutens 2003). Madan, Carr & Chang (1998) introduced the three parameter Variance Gamma process, while Barndorff-Nielsen (1994) first introduced the three parameter Normal Inverse Gaussian (NIG) process as a derivative pricing approach. This thesis will in depth cover option pricing under this NIG market model.

The numerical path integration method is a well described approach in the case where the driving noise is Brownian motion (see e.g. (Naess & Moe 2000) or (Skaug & Naess 2007)). Relatively few articles have been written when the driving noise is a more general Levy process, such as the Variance Gamma or the Normal Inverse Gaussian process. The numerical path integration method is used to approximate the probability density of the solution process of a stochastic differential equation. In the case of the NIG process, there is no explicit SDE for the development of a financial asset, but we will in this thesis show how the numerical path integration method can still be applied.

1.1 Thesis Outline

In chapter 2 we make a short introduction to necessary basic financial aspects. If the reader is familiar with option theory and financial modeling, there are no new insights to be made. In chapter 3 we define the Normal Inverse Gaussian process and give all the necessary mathematical preliminaries. We also discuss the impact of changes in the

different parameters. Chapter 4 is dedicated to a thorough statistical analysis of the performance of the NIG market model with empirical market data. We will calibrate our model using both historical market data (section 4.1-4.3) and current option prices (section 4.4). Finally, we explore the NIG market models sensitivity regarding changes in the parameters in section 4.6. In chapter 5 we show the framework for pricing some exotic options with the numerical path integration method. Finally we in chapter 6 produce numerical results to verify our implementation, and discuss the accuracy and computational speed.

Chapter 2

Financial Basics

In this chapter, we will give a brief introduction to the necessary financial basics for this thesis. In section 2.1 we briefly discuss the concept of indices, while we in section 2.2 introduce the Black-Scholes model. Then we review different types of options in section 2.3, before we in section 2.4 look at different numerical option pricing methods.

2.1 Stocks and Stock Market Indices

A stock market index is a method of measuring a certain section of the stock market. Stock market indices may be classed in many ways. A broad-base index represents the performance of a whole stock market - and by proxy, reflects investor sentiment on the state of the economy. The most regularly quoted market indices are broad-base indices comprised of the stocks of large companies listed on a nation's largest stock exchanges, such as the British FTSE 100, the French CAC 40, the German DAX, the Japanese Nikkei 225, the American Dow Jones Industrial Average and S&P 500 Index and the Norwegian Oslo Stock Exchange Benchmark Index (OSEBX). The concept of indices may be extended well beyond an exchange. The Dow Jones Wilshire 5000 Total Stock Market Index, as its name applies, represents the stocks of nearly every publicly traded company in the United States, including all U.S. stocks traded on the New York Stock Exchange and most traded on the NASDAQ and the American Stock Exchange. More specialized indices exist tracking the performance of specific sectors in the market. The Morgan Stanley Biotech Index, for instance, consists of 36 American firms in the biotechnology industry.

An index may also be classified according to the method used to determine its price. In a price-weighted index such as the Dow Jones Industrial Average and the NYSE ARCA Tech 100 Index, the price of each component stock is the only consideration when determining the value of the index. Thus, price movement of even a single security will heavily influence the value of the index even though the dollar shift is less significant in a relatively highly valued issue, and moreover ignoring the relative size of the company as a whole. In contrast, a market-value weighted or capitalization-weighted index such as the Hang Seng Index and the OSEBX factors in the size of the company. Thus, a

relatively small shift in the price of a large company like StatoilHydro ASA, will heavily influence the value of the OSEBX compared to an equal shift in a smaller company like Norwegian Air Shuttle.

When we are making mathematical calculations and simulations based on a certain theoretical market dynamics, it is interesting to make a brief discussion on whether the model dynamics is suitable for the markets we are looking at. In the next section, we briefly look at how well the Black-Scholes model fits the historical data from the Dow Jones Industrial Average. In chapter 4, we will see how the Normal Inverse Gaussian (NIG) distribution proposed by Barndorff-Nielsen (1994) fits the historical data from OSEBX, as well as the Dow Jones Industrial Average and some single stocks. We will also investigate its compatibility to market option prices using implied parameters.

When looking at relatively small indices and prices of single stocks, it is worth noticing how some properties of these may have an impact on the compatibility to the model dynamics. For instance, the relatively small size of OSEBX makes the index more dependent on certain factors (see e.g. (Bjornland 2008)). These factors are for instance the oil price and the development on bigger stock markets, such as those in New York and Asia. This dependence makes it harder to model the OSEBX independently without looking at what happens to the other markets. Secondly, the models usually assumes that the market has sufficient liquidity. The essential characteristic of a liquid market is that there exists buyers and sellers that are willing to sell and buy at all times and for all desired prices. If the market do not have sufficient liquidity, there would be sudden extreme movements caused by a lack of buyers or sellers, and these sudden, unexpected movements are hard to model. For instance, due to the current financial crisis, the market makers in fall 2008 experience difficulties pricing credit swaps and other credit derivatives.

2.2 The Black-Scholes Model Dynamics

There are several mathematical models of the underlying stock price dynamics used for pricing derivatives. The most popular is the Black-Scholes model proposed by Black & Scholes (1973), which assumes that the stock price develops according to a geometric Brownian motion given by

$$dS_t = \mu S_t dt + \sigma S_t dW_t \quad (2.1)$$

where dW_t is a Brownian motion with mean 0 and variance dt . Rewriting (2.1),

$$\frac{dS_t}{S_t} = \mu dt + \sigma dW_t$$

we notice that the stock price follows a lognormal distribution with mean μdt and variance $\sigma^2 dt$. Under this assumption, the probability for the stock price at time t , S_t , given that the stock price at time $t' < t$, $S_{t'} = s'$, is explicitly given as

$$p_{t|t'}(s|s') = \frac{1}{\sqrt{2\pi\Delta t}\sigma s} \exp\left(-\frac{(\log(s) - \log(s') - (r - \sigma^2/2)\Delta t)^2}{2\sigma^2\Delta t}\right)$$

where $\Delta t = t - t'$.

This normality assumption for the log return of the stock prices is the most crucial assumption of the Black-Scholes framework. The advantage with this model and the reason for why it has become so popular in the financial industry, is that we only have two parameters which we need to estimate; the mean μ and the variance σ^2 . These two parameters are completely observable in the historic data, which makes the estimation easy and straightforward. The mean μ is just the average log return, while the variance σ^2 is the volatility in the historic market data.

The downside with the Black-Scholes model, is also the fact that we only have two parameters to be fitted to the historical data. This gives limited possibilities to take into account heavy tails due to extreme variations, and also additional kurtosis^{2,1} for assets that have most log returns around the mean with some infrequent extreme variations. In Figure 2.1 we see these properties as the normal PDF fail to include the peak around zero and especially the extreme log returns between -0.1 and -0.05 . And at the same time it includes too much in the steep areas around the center.

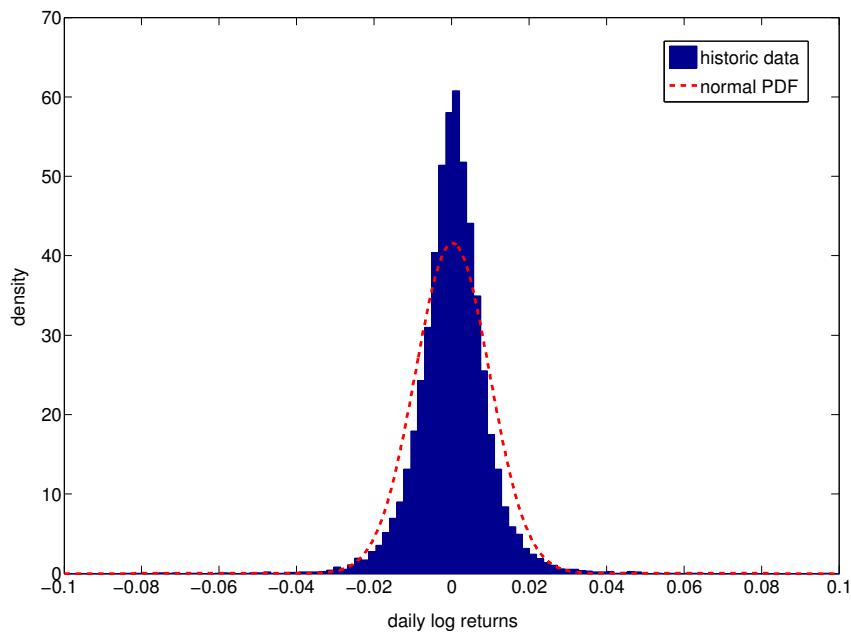


Figure 2.1: Histogram showing the daily log returns on Dow Jones Industrial Average in the period from 3rd of January 1950 to 27th of February 2009 with the normal probability density function (with parameters μ and σ^2 estimated from the same data set) as a red dashed line.

^{2,1}is a measure of the 'peakedness' of the probability distribution and higher kurtosis means more of the variance is due to infrequent extreme deviations

In addition to the normality assumption, Black & Scholes (1973) have made the following assumptions

- it is possible to lend or borrow cash at a constant risk-free rate r
- there are no transaction costs
- the stocks pay no dividends
- all securities are perfectly divisible (i.e. it is possible to buy fractions of a share)
- short selling is allowed
- trading of the underlying asset can take place continuously

For more details on the Black-Scholes model, and these assumptions, see e.g. Wilmott, Howison & Dewynne (1995).

2.3 Options

An option is a contract between a buyer and a seller that offers the buyer the right, but not the obligation, to buy (or sell) a security or another financial asset at an agreed-upon price (the strike price) during a certain period of time or on a specific date (exercise/maturity date). In return for granting the option, the seller collects a payment (the premium) from the buyer. To buy an option is often referred to as having a long position, while selling an option is referred to as a short position. There are two main types of options - American and European. An European option can only be exercised at the expiration date, whereas American options can be exercised at any time before and including the maturity date. When we have the option to buy the underlying asset, we have a call option, while a put option is the option to sell the underlying asset. One profits from buying a put if the price of the underlying asset decreases, and from buying a call when the price of the underlying increases. There exist numerous different types of options, from simple European call options to more complicated options. These various options exist because different institutions and investors have different demands concerning hedging^{2.2} their investments. And as long as there are parties willing to make up the opposite side of the contract, the options are traded in the market. Options that differ from the regular options types, are called exotic options, and examples of such exotic options are barrier options, lookback options and Asian options. These exotic options are typically traded over-the-counter (OTC)^{2.3}.

To illustrate the concept of hedging, we will look at a small example. Assume that American Airlines profits are a linear function of the oil price as showed in the first graph in Figure 2.2. Here we see that if the oil price exceeds \$120, then American Airlines will

^{2.2}hedging is a financial term for reducing or eliminating risk related to variations in a certain market

^{2.3}expression used for financial products that are traded directly between two parties, and not through an exchange

have negative profits. But if American Airlines buy call options with strike $X = \$120$ (with profits showed in the second graph in Figure 2.2), then American Airlines profits will be as showed in the last graph in Figure 2.2. Hence, by buying call options on the oil price, American Airlines have hedged away the risk of large negative profits as a result of a very large oil price.

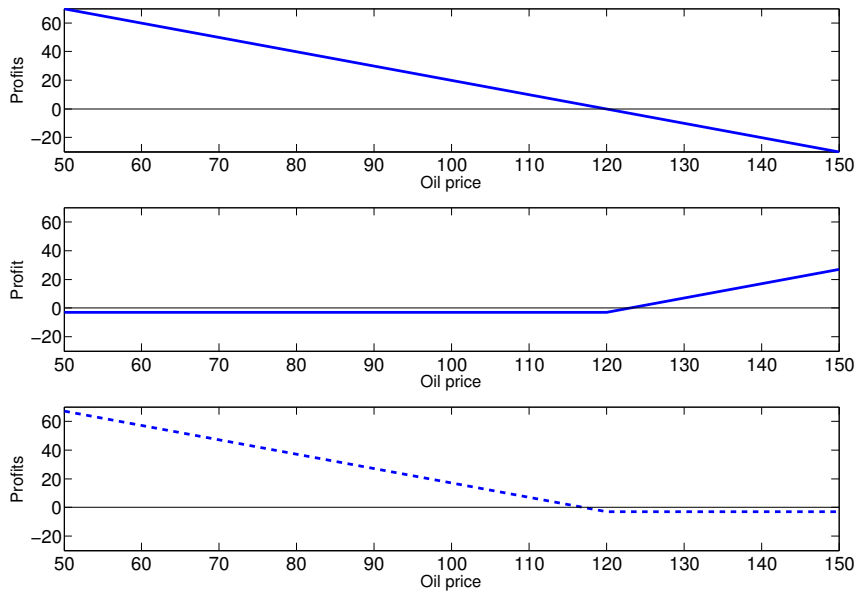


Figure 2.2: The upper graph show the profits of American Airlines as a linear function of the oil price, while the graph in the middle show the profit of a call option on the oil price, and the lower graph show the hedged profits for American Airlines.

There exists a large quantity of different options, from the simplest Vanilla^{2.4} options to more advanced exotic options. In this thesis we will focus on some of the most common exotic options; barrier options, lookback options and Asian options.

Barrier Options

Barrier options are a type of path dependent options, where the option to exercise depends on the underlying crossing or reaching of a given barrier. These barrier options were created to make the insurance value of an ordinary Vanilla option, without charging as much premium. Since they offer much of the same hedging qualities as a regular Vanilla option for a lower price, they have become very popular.

Barrier options are similar in some ways to ordinary options. There are puts and

^{2.4}term used for a regular call or put option

calls, as well as European and American varieties. But they become activated or, on the contrary, null and void only if the underlying asset reaches a predetermined level (barrier). Among these there are four main types: Up-and-out, down-and-out, up-and-in and down-and-in. In addition, there are combinations of these, such as an up-and-out-down-and-out option. An up-and-out options will cease to exist if the underlying asset reaches the barrier. For an up-and-in option, the holder does not have the option to exercise unless the underlying asset reaches the barrier. In mathematical modeling of barrier options, it is also important to discuss how frequently the underlying assets are monitored. It should be clear that an up-and-out option which is monitored continuously^{2.5}, have a higher probability of being observed crossing the barrier than a barrier option which is monitored discretely, i.e. at the end of each trading day. Hence the continuously monitored barrier options should be cheaper since there is a greater probability that it will be worth nothing in the end.

Lookback Options

A lookback option is also a type of path dependent option, which allows the investors to look back at the price of the underlying asset occurring over the life time of the option, and then exercise based on the underlying asset's optimal value. This type of option reduces the uncertainties associated with the timing on market entry for the investor. They have an obvious appeal to the investors, but since they are expensive, they tend to be more of a speculative device.

There are two main types of lookback options: options with a fixed strike price and options with a floating strike price. For a call (or put) with a fixed strike price, the option holder can look back over the life time of the option and choose to exercise at the point when the underlying asset was priced at its highest (or lowest). The option payoff is then the selected past market price minus the fixed strike price. For a call (or put) with a floating strike price, the payoff is the same as for the fixed strike, except that the strike price is fixed at maturity as the lowest (or highest) value of the underlying asset during the life time of the option. As for almost all options, we also have call and put variations of the option and European and American style lookback options.

Asian Options

The last exotic option type we will consider, is Asian options. They are also path dependent options, where the payoff at maturity depends on the average price of the underlying asset during the lifetime of the option. Asian options offers corporations and traders protection against price manipulations and/or large variations in the price of the underlying asset due to low trading volume. Obviously it is easier to manipulate the price at one point in time, compared to manipulating the average over a period of time. Hence the volatility of Asian options is lower than the volatility of the underlying asset. Due

^{2.5}In this setting, continuously must not be thought of in a mathematical sense. This term is in this thesis used to represent situations where no discrete monitoring is defined.

to this low volatility, the Asian options tend to be a cheaper alternative to its European counterparts.

There are commonly two types of averages used; arithmetic average (A^a) and geometric average (A^g)

$$A_N^a = \frac{1}{N+1} \sum_{i=0}^N S_i \quad \text{and} \quad A_N^g = \left(\prod_{i=0}^N S_i \right)^{\frac{1}{N+1}} \quad (2.2)$$

where N is the number of times we are monitoring the price of the underlying asset during the lifetime of the option. In this thesis we will focus only on the arithmetic average case, but it should be relatively straightforward to extend the theory to the geometric case. In addition there are fixed strike (also called average price) and floating strike (also called average strike), where the latter involves a payoff function depending on both the average and the price of the underlying asset at maturity. The option prices also depend on whether the underlying asset is monitored discretely or continuously.

2.4 Numerical Option Pricing Methods

A lot of different methods have been used for pricing and modeling different financial securities and dynamics. The most common ones are perhaps Monte Carlo simulation and binomial option pricing model, which we will discuss below. The development in modern computers have made these methods more applicable with regards to speed, but there are still a lot of problems that are too time-consuming.

Binomial Option Pricing Model

The binomial option pricing model (BOPM) approach is widely used as it is able to handle a variety of conditions for which other models can not easily be applied. This is mainly because the BOPM models the underlying asset over time, and not just at a particular point in time. Opposed to the Black-Scholes formula, BOPM can easily be modified to price American options and some exotic options. Although it is slower than using the Black-Scholes formula, it is considered to be more accurate, especially for longer-dated options. For these reasons, the BOPM is widely used by practitioners in the option markets. The essence of the model can be broken down into three steps:

- initializing a binomial tree, where each node represents a possible price of the underlying asset at a particular point in time
- calculate the option value at each final node (at maturity date)
- calculating the option values backwards from the final nodes to the initial node, which represents the current option price

The BOPM has, as mentioned above, a certain appeal to the practitioners. But since there are just two possible movements at each node (either up or down), you must have

a very small grid size in order to get a binomial tree that represents the reality in a satisfactory manner. This makes the BOPM very time-consuming, which is a obvious weakness of this method.

Monte Carlo Simulation

Monte Carlo simulation (MC) is a method that is more straightforward than many of the other options pricing methods. The MC simulation is a procedure in which random numbers are generated according to probabilities assumed to be associated with a source of uncertainty, in our case the underlying assets development in the future. A simple example of the use of MC simulation in finance, is the pricing of a Vanilla option, with the Black-Scholes model as the underlying asset dynamics. From (2.1) we have an expression for the stock price at time t given the stock price at time t'

$$S_t = S_{t'} (1 + rdt + \sigma dW_t) \quad (2.3)$$

where dW_t is $N(0, dt)$ and $dt = t - t' > 0$. For each step in each simulation, we generate a normal random variable with mean zero and variance dt . The stock price at the next time step is then given by (2.3), and at maturity we know the payoff, $\max[S_T - K, 0]$, for each simulation. The price of the Vanilla option, is then given as the average payoff of the option for each simulation, discounted^{2.6} to the initial point in time.

One of the most attractive properties of MC simulations is that it can be applied on complicated problems. We will return to Monte Carlo simulation in chapter 5.

Numerical Path Integration

The numerical path integration method involves a stepwise algorithm that computes the evolution of a probability distribution from an initial state or an initial distribution. For each discrete time step, we use the conditional transition probability, $p_{t_j|t_{j-1}}(x_j | x_{j-1})$, to propagate forward in time. This approach have the great advantage of handling a stochastic differential equation which may be a function of the underlying variable itself, in a precise way. Some dynamics, such as the Black-Scholes model and our NIG market model, have explicitly given transition probabilities, while in most cases we must find an approximate density. Skaug & Naess (2007) discusses two ways of finding this approximate transition probability, using Euler-Maruyama discretization and simplified weak Taylor scheme of order 2.0.

When we have an explicit expression for the transition probability from state $X_{t'}$ to X_t , $dt = t - t' > 0$, combined with the known initial state or initial distribution, $g_0(x)$, we can propagate forward to find the probability density function at time t

$$g_t(x) = \int_{-\infty}^{\infty} g_{t'}(x^*) p_{t|t'}(x | x^*) dx^*$$

^{2.6}discounting is a way of taking into account the time value of money

This probability density function we can use to find the information we want at a given time in the future, i.e. if $X_{t'}$ is a stock price at time t' , then we can find the expected payoff at expiration T for a European call with strike price K

$$E [(S_T - K)^+] = \int_K^{\infty} (x - K) g_T(x) dx$$

This is essentially what we need in order to price the options, but for some of the exotic options, we need to extend this to a two-dimensional case.

In addition to finding an approximate density function, there are mainly two numerical challenges when implementing path integration. The first one is to find a balance between the grid size and the computational speed to achieve a desired accuracy. For multidimensional problems, it may be critical to use interpolation in order to reduce the number of grid points. Secondly, it is critical to choose an accurate numerical integration method. For each time step we have to calculate m (where m is the number of spacial grid points) integrals, and hence the numerical errors will be considerable if the integration method is not accurate enough. The accuracy regarding the integration will also be influenced by the grid size, so we must also in this case consider interpolation in order to get the necessary accuracy when integrating.

Chapter 3

The Normal Inverse Gaussian Process

The Normal Inverse Gaussian (NIG) distribution is a special case of the generalized hyperbolic distributions, first introduced by Barndorff-Nielsen (1994). Due to the specific characteristics, the NIG distribution is very interesting for applications within finance. With its four parameters, the NIG distribution allows both skewness and higher kurtosis^{3.1} than the Normal (Gaussian) distribution. These properties, especially allowing slower decreasing tails, makes it a suitable tool for modeling financial derivatives with most commonly the stock return as the underlying asset.

The name of the NIG process is due to the fact that we can relate the NIG process to an Inverse Gaussian (IG) time-changed Brownian motion. The Inverse Gaussian distribution describes the distribution of the time a standard Brownian motion with drift $b > 0$ takes to reach a fixed positive level a . The time this takes can of course only be positive, and thus the Inverse Gaussian process is only defined for $x > 0$. The density function for the $IG(a, b)$ distribution is given by

$$f_{IG}(x; a, b) = \left(\frac{a}{2\pi x^3}\right)^{\frac{1}{2}} \exp\left(\frac{-a(x-b)^2}{2b^2x}\right) \quad (3.1)$$

and Figure 3.1 show this PDF for different values of b .

A random variable is said to be NIG distributed with parameters α , β , μ and δ , denoted $X \sim \text{NIG}(\alpha, \beta, \mu, \delta)$, where μ is the location, β is the skewness, α the tail-heaviness and δ the scale. The probability density function for a $\text{NIG}(\alpha, \beta, \mu, \delta)$ -variable is given by

$$f_{NIG}(x; \alpha, \beta, \mu, \delta) = \int_0^\infty f_N(x; \mu + \beta y, y) \cdot f_{IG}(y; \delta\gamma, \gamma^2) dy \quad (3.2)$$

^{3.1}It is important to be aware that there exist two main kurtosis measures; the Pearson kurtosis and the Fisher kurtosis. The Fisher kurtosis is equal to the Pearson kurtosis minus 3. The built-in function *kurtosis()* in Matlab returns the Pearson kurtosis measure, while the built-in function *kurt()* in Excel returns the Fisher kurtosis measure. Since we use Matlab, we have chosen to use the Pearson measure.

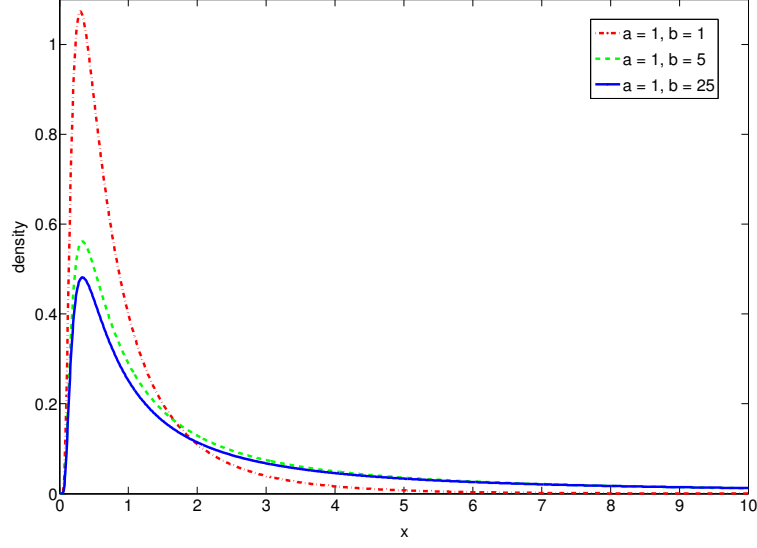


Figure 3.1: Inverse Gaussian probability density functions.

where f_N denotes the density function of the normal Gaussian distribution, $\gamma = \sqrt{\alpha^2 - \beta^2}$ and

$$\alpha > 0 \quad , \quad -\alpha < \beta < \alpha \quad , \quad \delta > 0 \quad , \quad \mu \in \mathbb{R}$$

This density function can also be written as

$$f_{NIG}(x; \alpha, \beta, \mu, \delta) = \frac{\alpha\delta}{\pi} \exp(\delta\gamma + \beta(x - \mu)) \frac{K_1\left(\alpha\sqrt{\delta^2 + (x - \mu)^2}\right)}{\sqrt{\delta^2 + (x - \mu)^2}} \quad (3.3)$$

where $K_1(\cdot)$ denotes the modified Bessel function of the second kind, defined as

$$K_1(u) = \frac{1}{2} \int_0^\infty \exp\left(-\frac{1}{2}u(t + t^{-1})\right) dt$$

For further details on the Bessel function, see e.g. Abramowitz & Stegun (1968). Kalemánova & Werner (2006) show that the latter representation of the density function is computationally much more efficient, and so we will later use this representation for the density function. We can define the NIG process

$$X^{\text{NIG}} = \{X_t^{\text{NIG}}, t \geq 0\} \quad (3.4)$$

with $X_0^{\text{NIG}} = 0$ and stationary and independent NIG distributed increments. $X_{t+dt}^{\text{NIG}} - X_t^{\text{NIG}}$ is then NIG distributed with parameters α , β , μdt and δdt .

Schoutens (2003) stated that a NIG process X^{NIG} with parameters $\alpha > 0$, $|\beta| < \alpha$, $\mu \in \mathbb{R}$ and $\delta > 0$ can be obtained by time-changing a standard Brownian motion $W = \{W_t, t \geq 0\}$ with drift by an IG process $I = \{I_t, t \geq 0\}$ with parameters $a = 1$ and $b = \delta\sqrt{\alpha^2 - \beta^2}$. Hence

$$X_t^{\text{NIG}} = \mu + \beta\delta^2 I_t + \delta W_{I_t} \quad (3.5)$$

is a NIG process with parameters α , β , μ and δ . As mentioned earlier, the NIG distribution with its possibility for slower decreasing tails, is a very suitable tool for modeling financial derivatives. Figure 3.2 show a sample path of a NIG distributed stock price process, and as we can see there are some extreme movements in the process. These events happen more frequently in the NIG process, than in the Black-Scholes framework, which we in chapter 4 will see is more according to historical data collected from various financial indices and stocks.

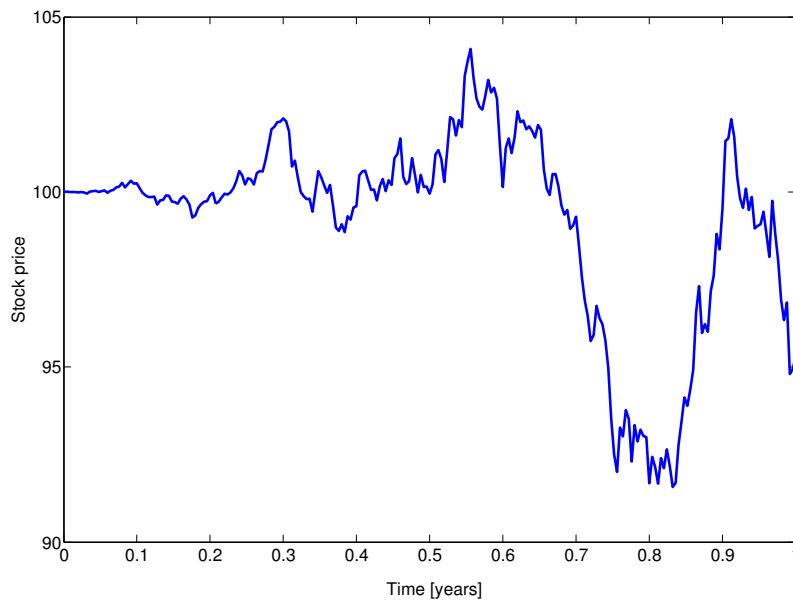


Figure 3.2: A NIG sample path of a stock price development, with $S_0 = 100$, $T = 1$, $\alpha = 70$, $\beta = -15$ and $\delta = 1$.

3.1 The Effects of changing the Different Parameters

Except from the location parameter μ , it might not be obvious how changing the different parameters effect the density function and the central moments of a NIG distributed random variable. In Figure 3.3 we have plotted the probability density function for the NIG distribution, varying one of the parameters at a time, to get an intuitive understanding

of the effects of the changes. As mentioned in the beginning of this chapter, we see that α controls the tail-heaviness, β the skewness of the density function, μ the location and δ controls the scale of the density function.

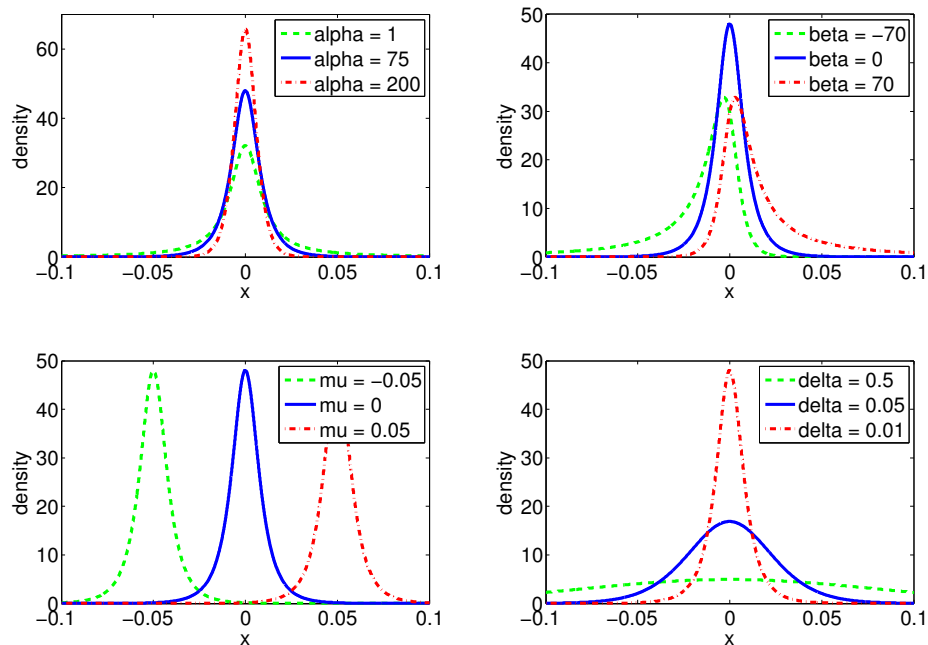


Figure 3.3: The NIG distributed density functions with one parameter value varying (default: $\alpha = 75$, $\beta = 0$, $\mu = 0$ and $\delta = 0.01$)

The central moments of the NIG distribution is summarized in Table 3.1. From this table, it might be a bit hard to intuitively recognize the effect on these moments from a change in one of the parameters. We have therefore summarized these effects in Table 3.2.

Moment	$NIG(\alpha, \beta, \delta)$
Mean(\mathbb{E})	$\mu + \frac{\delta\beta}{\gamma}$
Variance(\mathbb{V})	$\frac{\delta\alpha^2}{\gamma^3}$
Skewness(\mathbb{S})	$\frac{3\beta}{\alpha\sqrt{\delta\gamma}}$
Kurtosis(\mathbb{K})	$3 + 3 \left(1 + 4 \left(\frac{\beta}{\alpha} \right)^2 \right) \frac{1}{\delta\gamma}$

Table 3.1: Expressions for the central moments of the NIG distribution

		α		β			μ	δ		
		\mathbb{V}	\mathbb{K}	\mathbb{E}	\mathbb{V}	\mathbb{S}	\mathbb{K}	\mathbb{E}	\mathbb{V}	\mathbb{K}
Change:	\uparrow	\downarrow	\downarrow	\uparrow	\uparrow	\uparrow	\uparrow	\uparrow	\uparrow	\uparrow
	\downarrow	\uparrow	\uparrow	\downarrow	\uparrow	\downarrow	\uparrow	\downarrow	\downarrow	\downarrow

Table 3.2: Parameter effects on the central moments.

3.2 NIG Market Model

When pricing financial derivatives in a NIG market model framework, we want to model the logarithmic stock returns with the assumption that they are NIG distributed. If we let S_t denote the stock price at time t , then the model is given by

$$S_t = S_0 \cdot \exp\left(\tilde{X}_t^{NIG}\right) \quad (3.6)$$

where \tilde{X}_t^{NIG} denotes the NIG process in (3.5) with $\mu = 0$. Comparing this NIG market model to the more famous geometric Brownian motion model by (Black & Scholes 1973), there are some drawbacks. First, we can not construct the (in theory) perfect delta hedge^{3.2} which in the Black-Scholes model enables us to find a closed form expression for the price of a Vanilla call option. Secondly, in this geometric Brownian motion case, moving from the real world to the risk neutral world^{3.3} is done by setting the drift μ equal to the risk free rate r minus the continuous dividend yield d . However, for our NIG model in equation (3.6) there is no unique transformation from the real world to the risk neutral world. Schoutens (2003) show how we can apply the mean-correcting measure change, where we simply add a drift term m to our model. Our model is then given by

$$S_t = S_0 \cdot \exp\left(mt + \tilde{X}_t^{NIG} + \omega t\right) \quad (3.7)$$

where Schoutens (2003) showed that in the NIG case, ω is given by

$$\omega = \delta \left(\sqrt{\alpha^2 - (\beta + 1)^2} - \sqrt{\alpha^2 - \beta^2} \right) \quad (3.8)$$

In the risk neutral world, m must be given as the risk-free rate subtract the continuous dividend yield, $m = r - d$. In the real world, m must be estimated from historical data or some personal insight to how the future market development will be. In the NIG market model, it is important to notice that all parameters are changed when moving from the real world to the risk neutral world. This differs from the geometric Brownian motion case, where the volatility stays the same under the Girsanov measure transformation. Combining the probability density function from (3.3) with (3.7) and (3.8), the density for the logarithm on the price, $Z_t = \log(S_t/S_0)$, is given by

$$f(z; t) = \frac{\alpha \delta t}{\pi} \exp\left(\delta t \sqrt{\alpha^2 - \beta^2} + \beta(z - \mu t)\right) \frac{K_1\left(\alpha \sqrt{\delta^2 t^2 + (z - \mu t)^2}\right)}{\sqrt{\delta^2 t^2 + (z - \mu t)^2}} \quad (3.9)$$

where

$$\mu = m + \omega = m + \delta \left(\sqrt{\alpha^2 - (\beta + 1)^2} - \sqrt{\alpha^2 - \beta^2} \right)$$

^{3.2}delta hedging is a technique where you remove all risk from a portfolio by continuously selling or buying $\delta = \frac{\partial V}{\partial S}$ of the underlying asset S , where V denotes the fair value of the portfolio or instrument.

^{3.3}term used for the risk-neutral measure, which is a probability measure where one assumes that the current value of all financial assets is equal to the expected value of the future payoff of the asset, discounted at the risk-free rate.

Figure 3.4 show the risk-neutral probability density function for a stock price modeled by a NIG market model. It also show the PDF for the result using the corresponding lognormal distribution from Black & Scholes (1973). As we can see from this plot, the NIG market model allows for both skewness and higher kurtosis, and in the next chapter we will see that these qualities makes the model coincide better with the historical market data.

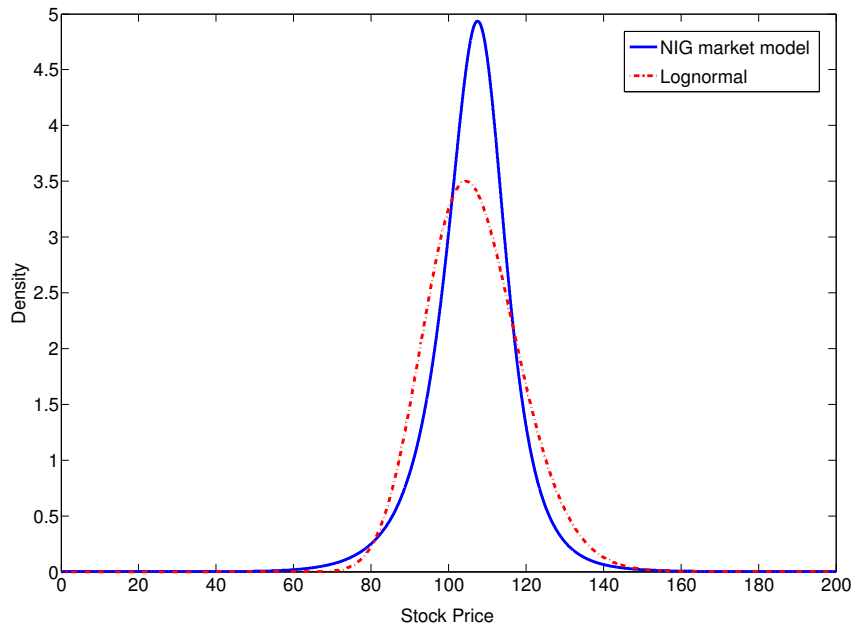


Figure 3.4: The risk-neutral densities obtained with both a lognormal and NIG distribution of the stock price. Parameters used are $S_0 = 100$, $T = 1$ year, $\alpha = 10$, $\beta = -4$, $\delta = 0.1$ and $r = 0.05$. The corresponding σ for the lognormal distribution for these parameters is 0.114.

When we later use this in our path integration approach, we will use daily logarithmic returns

$$\Delta Z_t = Z_t - Z_{t-\Delta t} = \log\left(\frac{S_t}{S_{t-\Delta t}}\right) \quad (3.10)$$

and so the density function will be

$$f(\Delta z; \Delta t) = \frac{\alpha \delta \Delta t}{\pi} \exp\left(\delta \Delta t \sqrt{\alpha^2 - \beta^2} + \beta(\Delta z - \mu \Delta t)\right) \frac{K_1\left(\alpha \sqrt{\delta^2 \Delta t^2 + (\Delta z - \mu \Delta t)^2}\right)}{\sqrt{\delta^2 \Delta t^2 + (\Delta z - \mu \Delta t)^2}} \quad (3.11)$$

Chapter 4

Market Calibration of Parameters

Our motivation for using the more complex NIG market model instead of the Black-Scholes model, is that we are able to include skewness and kurtosis when fitting our model to the market data. It should be clear that when we now have four parameters which need to be calibrated, instead of the normal distributions two variables, we should get a calibration which is better fitted to the market data. However, an increase in the number of parameters also lead to a greater model risk, since there are some model risk associated with each parameter. In this chapter, we will use the two most common approaches for estimating the parameters; historical and implied. In sections 4.1-4.3 we will use the historical data approach. We first find the maximum likelihood estimator (MLE) for the four parameters μ , α , β and δ . Then we will examine if the NIG distribution actually gives a good fit to the stock returns, and this test of goodness of fit we will do both graphically and with numeric measures. In section 4.4 we will use market prices on Vanilla options to calibrate the four parameters. These parameters we call *implied parameters*. In section 4.6 we will perform a short sensitivity analysis to see how sensitive the price of a Vanilla call option is to changes in the NIG-parameters.

4.1 Maximum Likelihood Estimator

The Maximum Likelihood Estimator (MLE) is a popular statistical method used for fitting a mathematical model to historical data. The principal of MLE, which originally was developed by R.A. Fischer in the 1920s, states that the desired probability distribution is the one that makes the historical data 'most likely'. Hence, we seek the parameter values that maximizes the likelihood function

$$L(\alpha, \beta, \delta, \mu | \Delta z) = f(\Delta z_1, \dots, \Delta z_n | \alpha, \beta, \delta, \mu) = \prod_{i=1}^n f(\Delta z_i | \alpha, \beta, \delta, \mu) \quad (4.1)$$

where $f(\Delta z_i | \alpha, \beta, \delta, \mu)$ is the NIG probability density function, α , β , δ and μ are the NIG parameters we want to estimate and $\Delta z = [\Delta z_1, \dots, \Delta z_n]$ is a vector containing the historical daily log-return data. The last equality of (4.1) we get from the assumption that

our daily log returns are iid. This assumption is commonly used in financial modelling. For computational convenience, we obtain the MLE estimate by maximizing the log-likelihood function

$$\log(L(\alpha, \beta, \delta, \mu|\Delta z)) = \sum_{i=1}^n \log(f(\Delta z_i|\alpha, \beta, \delta, \mu)) \quad (4.2)$$

This is because the two functions in Eq. (4.1) and Eq. (4.2) are monotonically related to each other. So maximizing either one would produce the same MLE estimate. For further details on the MLE estimation, see e.g. (Myung 2003).

The maximization of the log-likelihood function is done numerically using an optimization algorithm. We have chosen to use the built-in Matlab function *fminsearch*, which uses the Nelder-Mead method (also called downhill simplex method). Using this nonlinear optimization algorithm, our choice for the initial parameter values is critical. If a multidimensional solution space has several local minima, we need to choose initial parameters near the global minima in order to ensure that our solution is the global minima. These wise initial values, can be found by graphically adjusting the NIG distributed PDF to the histogram of the daily log returns. The plots in Figure 3.3 show the effects of changing the different parameters, and can be helpful when finding the initial parameter values in this way. An alternative way of choosing initial parameter values, is to estimate the central moments in Table 3.1 using time series estimation, and then solve the equations to find the initial parameter values.

4.2 Historical Parameter Estimation

The compatibility of mathematical models to historical data may vary a great deal for different stocks and different indices, due to a varying degree of liquidity and dependence on different external factors etc. In this thesis, we have chosen to look at two stocks and two indices, where one is Norwegian and one is American. From the discussion in section 2.1, one would assume that the NIG model would have a better fit for the large American Dow Jones Industrial Average index with history back to 1950, than for the Norwegian StatoilHydro stock.

Table 4.1 show details on the historical data sets we have used, and Table 4.2 show the MLE estimates for the NIG parameters and the corresponding Black-Scholes parameters. The historic data used is collected from *finance.yahoo.com* and are the closing prices adjusted for dividends etc.

Ticker	Description	Data from:	Data to:
STL.OL	StatoilHydro on the OSE	18.06.2001	27.02.2009
OSEBX.OL	Oslo Stock Exchange Benchmark Index	25.05.2001	27.02.2009
KO	Coca Cola Company on NYSE	03.01.2000	27.02.2009
^DJI	Dow Jones Industrial Average	03.01.1950	27.02.2009

Table 4.1: Short description of the data sets with OSE referring to the Oslo Stock Exchange and NYSE referring to the New York Stock Exchange. *Data source* : *finance.yahoo.com*

	NIG				Black Scholes	
	α	β	δ	μ	μ	σ
StatoilHydro	48.9813	-1.7204	5.4548	0.2902	3.9382×10^{-4}	0.0214
OSEBX	44.2755	-10.2123	2.6999	0.6457	2.2990×10^{-5}	0.0165
Coca Cola	41.0038	-0.3620	2.5567	0.0083	-5.7233×10^{-5}	0.0158
Dow Jones	91.1804	-4.9853	1.9317	0.1657	2.3986×10^{-4}	0.0096

Table 4.2: Estimated NIG and Black-Scholes (normal) parameters

4.3 Goodness of Fit

With the parameters estimated from our stock return data, we now need to check how well our NIG model fit the historical data. There are many ways of doing this, and we will start off with perhaps the most intuitive one. Figure 4.1 show the daily log returns plotted as a normalized histogram, and according to the theory presented so far, these log returns should be NIG distributed with the respective estimated parameters from Table 4.2. The NIG distributed PDF is plotted as the red dashed line, and from the figure we see that empirical distribution and the theoretical distribution coincides quite well, except perhaps for StatoilHydro. The green dotted line show the normal probability distribution with the estimated parameters from Table 4.2. Looking at Figure 4.1, the NIG distribution seems to give a good fit to the empirical distribution, and a relatively much better fit than the normal distribution. For StatoilHydro, we see that there is a small improvement, but it seems hard to find a pattern in the empirical data. As mentioned before, the size and trading volume etc. will have an impact on an index or a stocks compatibility to a theoretical model. For StatoilHydro it may seem as the historical data set and/or the relative size of the company is too small.

Figure 4.1 gives us a fairly good picture of how well the NIG distribution fits the empirical distribution, but it is hard to see whether the NIG distribution manages to include the tail events. These events are very hard to model, and in chapter 2 we discussed how the Black-Scholes model fail to include these. So in order to determine

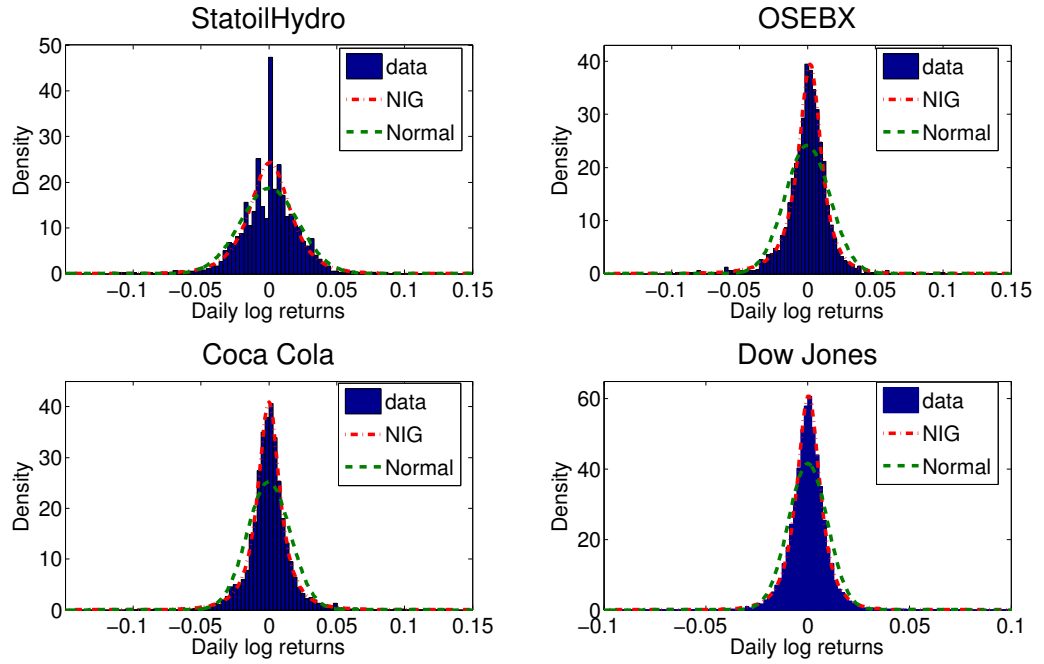


Figure 4.1: Normalized histograms over the daily log return for OSEBX, StatoilHydro, Coca Cola Company and Dow Jones, with the corresponding NIG distributed PDFs (red dashed line) and the normal PDFs (green dashed line).

how well the NIG distribution includes the tail events, we have in the left column of Figure 4.2 plotted the QQ-plots (Quantile-Quantile plots) for the empirical distribution against the theoretical NIG distribution with the estimated parameters. For comparison, we have in the right column of Figure 4.2 plotted the QQ-plots with the normal theoretical distribution.

If the empirical and theoretical distributions coincides perfectly, then they should form a straight line (the red dashed line). From the normal QQ-plots in Figure 4.2 we see how the tails are not included, while the NIG QQ-plots seem to capture the tail behaviour much better. But still, the NIG distribution do not include all extreme daily variations. As mentioned earlier, it is very hard to find a model that is able to capture all these tail events. If we look at the lower left corner of the QQ-plots for the Dow Jones Industrial Average, we see one extreme outlier. This outlier is due to the famous crash on the 19th of October 1987, also known as Black Monday, when the stock markets all over the world crashed and the Dow Jones Industrial Average fell 22.6% in one single day. The NIG model seem to fit the historical data quite well, but as all other models, it underestimates the probability of a stock market crash like the one in October 1987.

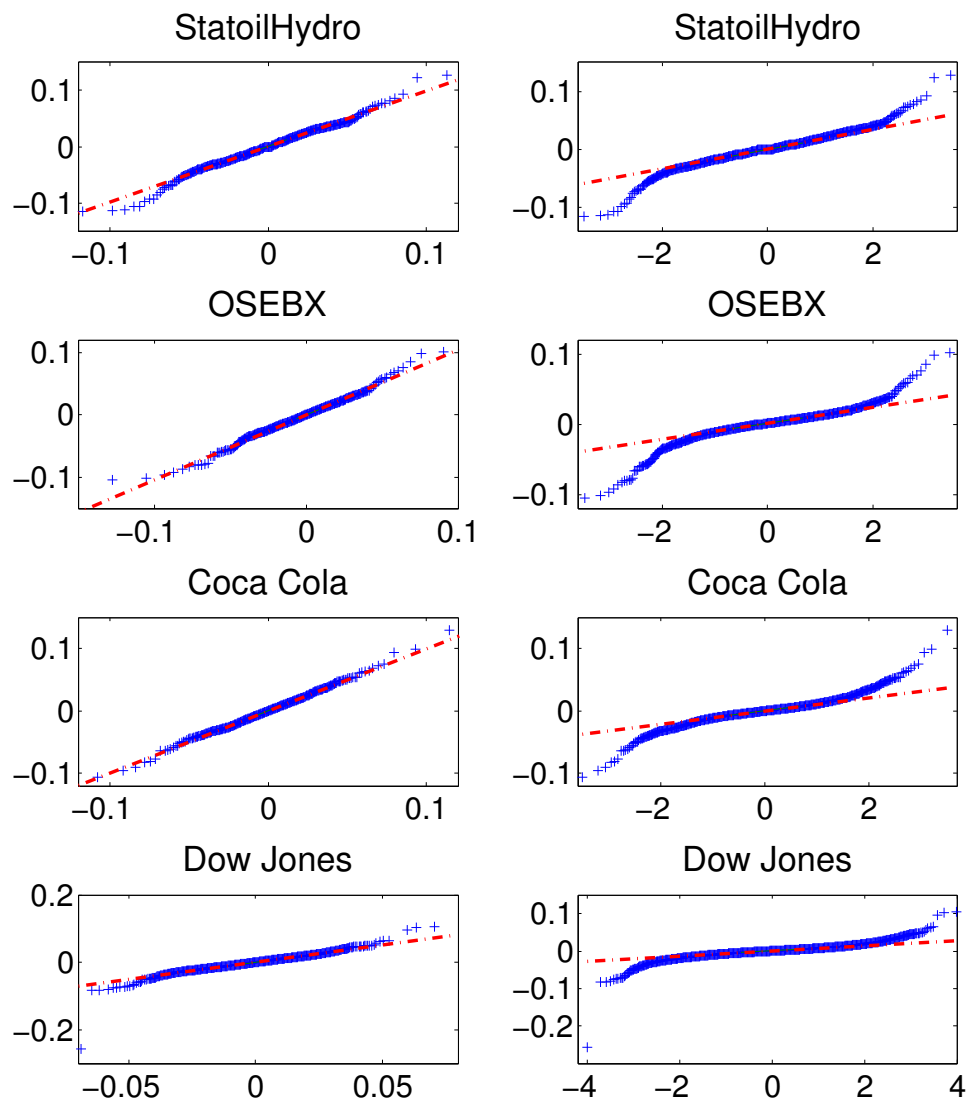


Figure 4.2: QQ-plots for the daily log return for OSEBX, StatoilHydro, Coca Cola Company and Dow Jones. The left column showing the data compared with the NIG distributed quantiles, while the right column is showing the data compared with the normal distributed quantiles. The quantiles of the data are on the y-axis.

Both the distribution plots and the QQ-plots we have presented so far indicates that the goodness of fit is relatively good for the NIG distribution, with perhaps StatoilHydro as a small exception. Visual examination of statistical plots is the best way to get an impression of how well the data fits a theoretical distribution. However, they might sometimes trick us into making a less correct conclusion. An example of this we see if we compare the plot for StatoilHydro in Figure 4.1 with Figure 4.3. In Figure 4.3 we have only divided the histogram into 30 bins, and from the plot, it might seem as the NIG distribution gives a reasonably good fit. But if we look at the plot in Figure 4.1, where we have divided the data into 80 bins, we might suspect that the conclusion is different. For reasons like this, it is also useful to do statistical hypothesis testing. In our case, the null hypothesis is that the historical daily log return data are generated from a NIG distribution with parameters given in Table 4.2. There are several tests one can use for testing goodness of fit, and we will here use the Kolmogorov-Smirnov test.

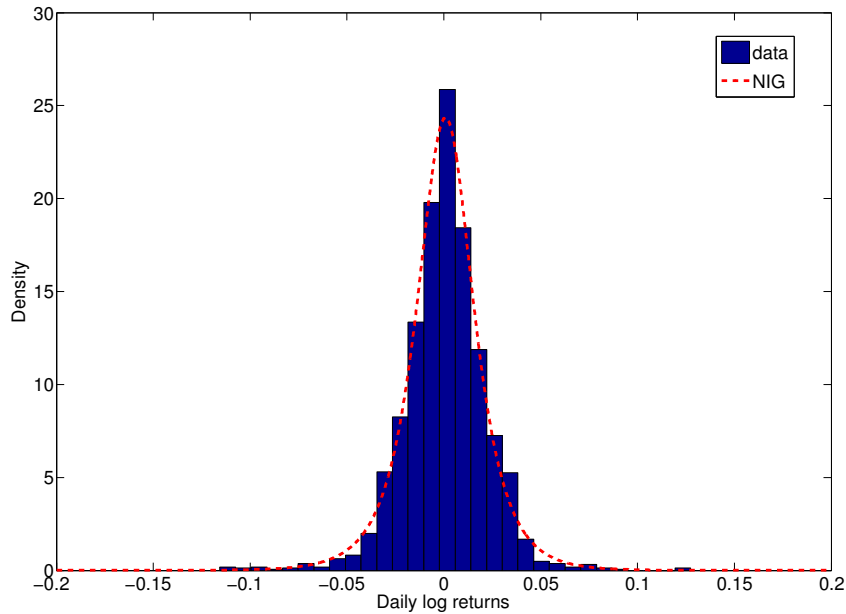


Figure 4.3: Normalized histogram of historical log returns for StatoilHydro with only 30 bins, and the corresponding NIG distributed PDF.

The Kolmogorov-Smirnov test is a form of minimum distance estimation used as a parametric test of equality of one-dimensional probability distributions used to compare a sample with a reference probability, in our case the NIG distribution with the parameters from Table 4.2. The Kolmogorov-Smirnov statistic, D_n , quantifies a distance between the empirical distribution function of the sample, $F_n(\Delta z)$, and the cumulative distribution function the reference distribution, $F(\Delta z)$. The empirical distribution function $F_n(\Delta z)$

for n iid observations Δz_i is defined as

$$F_n(\Delta z) = \sum_{i=1}^n I(\Delta z_i \leq \Delta z)$$

where $I(\Delta z_i \leq \Delta z)$ is the indicator function, equal to 1 if $\Delta z_i \leq \Delta z$ and 0 otherwise. The Kolmogorov-Smirnov statistic for a given cumulative distribution function $F(\Delta z)$ is

$$D_n = \sup_z |F_n(\Delta z) - F(\Delta z)|$$

where \sup_z denotes the supremum with respect to z . According to the Kolmogorov-Smirnov test, our null hypothesis is rejected at level α^* if

$$\sqrt{n}D_n > K_{\alpha^*} \tag{4.3}$$

where K_{α^*} is found from

$$P(K \leq K_{\alpha^*}) = 1 - \alpha^*.$$

K is a random variable following the Kolmogorov distribution with cumulative distribution function given by

$$P(K \leq x) = \frac{2\pi}{x} \sum_{i=1}^{\infty} \exp\left(-\frac{(2i-1)^2\pi^2}{8x^2}\right)$$

For more details on the Kolmogorov-Smirnov test, see e.g. (Massey 1951).

α^*	K_{α^*}
0.5	0.8276
0.3	0.9731
0.2	1.0728
0.1	1.2238
0.05	1.3580
0.025	1.4800
0.01	1.6280
0.005	1.7300
0.001	1.9470

Table 4.3: Quantiles of the Kolmogorov distribution for different values of α^* .

In Table 4.3 we have calculated the value of K_{α^*} for different values of α^* , and in Table 4.4 we have calculated the Kolmogorov-Smirnov test statistic $\sqrt{n}D_n$ using both the NIG distribution and the normal distribution as the reference distribution. Using Eq. (4.3) to compare the test statistic values in Table 4.4 to the quantiles in Table 4.3, we

get the same conclusions as we have seen before for all α^* -values between 0.001 and 0.5. Our null hypothesis that the empirical daily log return data follows a NIG distribution with parameters from Table 4.2 is rejected for all data sets when evaluating against the normal distribution. And for the NIG distribution, we accept the null hypothesis for all data sets, except for StatoilHydro.

Name	NIG	Normal
StatoilHydro	2.6089	2.9492
OSEBX	0.6460	3.9050
Coca Cola	0.4350	4.1474
Dow Jones	0.5809	8.1176

Table 4.4: Values of the Kolmogorov-Smirnov test statistic ($\sqrt{n}D_n$) for both NIG and Normal

To conclude, the NIG distribution seems to fit the two stock indices and the relatively large and liquid Coca Cola stock reasonably well, as we have seen in several ways. Even though StatoilHydro is a large corporation in a Norwegian scale, it is still a bit small and illiquid compared to stock indices and large international corporations.

4.4 Implied Parameter Estimation

The historical parameters discussed in the previous section, seem to give good fit to the historical data, except perhaps for StatoilHydro. The drawback with this approach is that the history of the underlying asset may not represent the investors future expectation regarding the development of the underlying asset in a satisfactory manner. And these future expectations is in fact what drives the price of the underlying asset, so one might say that this is the most crucial factor in modelling the future. The second approach, using the implied parameters in the model, better absorbs these future expectations by considering the actual prices in the market for derivatives. This approach minimizes some error measure between the model prices and comparable prices in the market for all different strikes and maturities.

There are several different pricing error measures for studying the performance of our model compared to the market. The average relative percentage error^{4.1} (ARPE), the root mean square error (RMSE) and the average absolute error (AAE) are defined as:

^{4.1}ARPE will be very high in cases where the market price of options are close to zero, and we will therefore exclude market option prices less than 0.01.

$$\begin{aligned}
ARPE &= \frac{1}{N} \sum_{i=1}^N \frac{|P_i^{mod} - P_i^{obs}|}{P_i^{obs}} \\
RMSE &= \sqrt{\frac{1}{N} \sum_{i=1}^N (P_i^{mod} - P_i^{obs})^2} \\
AAE &= \frac{1}{N} \sum_{i=1}^N |P_i^{mod} - P_i^{obs}|
\end{aligned} \tag{4.4}$$

where P^{mod} are the modelled prices and P^{obs} are the corresponding prices found in the market. For a given set of option parameters, there is however not one distinct *price* in the market. The three most commonly listed prices for an financial instrument are the bid price, the ask price and the last traded price. The bid and ask price are the prices that the financial institutions (market makers) offer to buy and sell, respectively, the instrument. The last traded price tells us at what price the last transaction in this instrument were bought/sold for. However, this transaction might have been one second ago or one day, depending on the liquidity of the instrument. This uncertainty makes the last trade price a less attractive choice compared to the bid/ask prices, where the market makers constantly update their opinion on what is the fair price.

The difference between the bid and ask price, is called the bid-ask spread. The market makers make their money on this spread, and in many markets this spread can be of significant size. The most obvious way to imply a market price from these two, is to take the average

$$P_i^{mid} = \frac{P_i^{bid} + P_i^{ask}}{2}$$

However, the bid-ask spread may vary significantly across different strikes and maturities for the same instrument, and so this obvious choice may not be the best choice. The bid-ask spread is in general a measure of the options liquidity, and higher liquidity gives a tighter bid-ask spread. When calibrating our model to the market data, it might be a good idea to give the more liquid options a higher weight, since these option prices contains more information. If we look at the error measures in (4.4), we see that the AAE and RMSE assign the options with large spreads more weight, which is exactly the opposite of what we want. Therefore we will in addition perform the calibration with a fourth error measure, where we adjust the ARPE with the relative spread $\frac{P_i^{ask} - P_i^{bid}}{P_i^{mid}}$. This error statistic is given by:

$$ARPE_{spread} = \frac{1}{N} \sum_{i=1}^N \frac{|P_i^{mod} - P_i^{mid}|}{P_i^{mid} \cdot \frac{P_i^{ask} - P_i^{bid}}{P_i^{mid}}} = \frac{1}{N} \sum_{i=1}^N \frac{|P_i^{mod} - P_i^{mid}|}{P_i^{ask} - P_i^{bid}} \tag{4.5}$$

There is naturally no correct answer to what error measure is best, but we will try out all four and see if we can come to a conclusion later.

The calibration is performed by using a numerical optimization algorithm to find the parameters that minimize the error statistics above. As we did earlier, we use the built-in function *fminsearch* in Matlab to perform the minimization. In some cases, this function violates the constraints given for NIG distributed PDF in chapter 2, but this is solved by adding a penalty term to the objective (see e.g. (Nocedal & Wright 2000)). When performing the calibration, it is important to be aware of the calibration risk. Calibration risk is the risk of performing a calibration where the resulting parameters gives an option price that deviates considerably from the market data. By performing the calibration with different error measures, as we will do later, we can investigate the calibration risk by comparing the resulting parameter sets. For further details on calibration risk, see (Detlefsen & Härdle 2006).

Performing the calibration, we have used option chains^{4.2} from OBX, StatoilHydro, Coca Cola and Dow Jones Industrial Average collected from Bloomberg on the 27th of April 2009. As the risk-free interest rate r we have used the Norwegian and US Treasury bill rates, because they are in practice the most risk-free rate available. All the modelled option prices and market prices are listed in Appendix B. We will focus on pricing call options, and for this purpose it is common to use call option market prices for the calibration. Table 4.5 show how many of the total modelled call prices that end up inside the bid-ask spread. Although the optimization is done with concern to the error measures in (4.4) and (4.5), one might say that this also is a desired property.

	$ARPE_{spread}$	$ARPE$	$RMSE$	AAE	Total
OBX	41	41	40	41	51
DJI	168	168	168	168	168
KO	49	38	36	41	76
STL	31	25	22	31	46

Table 4.5: Number of call prices inside bid-ask spread. The last column show the total number of options prices in the option chains.

If we look at the numbers in Table 4.5, we see that for the OBX index, approximately 80% of the modelled prices end up inside the spread and the number does not depend significantly on the error measure. By considering the calibration results for the OBX in Table 4.6, we see that the $ARPE_{spread}$ and $ARPE$ errors are relatively small, especially for $ARPE$. Considering that we are trying to fit a theoretical model to actual market data, these results are relatively good. One should not put too much thought into the numbers in Table 4.5, but they may be a good indication to how well our model fits the market data.

Continuing to analyse the numbers in Table 4.6, we see that the parameters for the four different error measures are reasonably similar. One might argue that a difference

^{4.2}data chains containing option prices for all maturities and strike prices available in the market for the given asset

	NIG				Black Scholes	
	α	β	δ	Error	σ	Error
ARPE _{spread}	8.9932	-4.5176	1.1528	14.19%	0.3627	47.65%
ARPE	9.2214	-4.5964	1.1783	1.71%	0.3559	4.44%
RMSE	8.9070	-4.8770	1.0764	0.2835	0.3873	0.8717
AAE	8.1057	-4.2033	1.0517	0.2097	0.3823	0.6834

Table 4.6: Implied parameters for OBX option prices. The market data is collected from Bloomberg on 27th of April 2009, and the index was in 217.61 NOK. The prices are for three different maturities; $T_1 = \frac{17}{250}$, $T_2 = \frac{37}{250}$ and $T_3 = \frac{57}{250}$. For these three maturities, we have used the 3 month Norwegian Treasury bill rate (1.92%) as the risk-free interest rate, r .

of 13.8% between the α value from ARPE and AAE is a relatively large deviation, but if we look at the output prices in Appendix B, we see that the prices are almost identical. In other words, the calibration risk for the OBX index is very small, at least for this option chain. In Table 4.6 we have also included implied volatility for the Black-Scholes model, and we see that the errors under this model are approximately three times as large. Earlier in this chapter, we discussed how the Black-Scholes model fail to include crucial aspects of the historical data, and here we again observe that the Black-Scholes model seem to fit the market data considerably less than our NIG model. The fact that this error is that large, is perhaps a bit surprising, considering that the Black-Scholes model is the most common model in financial institutions. Therefore one should expect the Black-Scholes model to have a good fit to the market option prices, since the market makers most likely have used this model to find their bid/ask prices. One explanation for this, might be that the market is very irregular due to the current financial situation, which may affect the market makers pricing. Another reason might be that the market makers include additional factors to their models.

Figure 4.4 show a plot of the OBX call option prices for our data chain, and as we can see they fit inside the bid-ask spread quite well, especially the NIG prices. For the largest strike prices and maturity dates, we see that both modelled prices falls outside the spread, but elsewhere the NIG prices stay inside this spread. It might be hard to get much more information from this plot, but all prices are listed in Table B.1 in Appendix B.

We have seen that the OBX prices are quite consistent, and Table 4.7 show that the errors for the Dow Jones Industrial Average are even smaller. The ARPE is less than 1%, and the parameters are quite similar. The fact that both the parameters and the modelled prices are almost identical, again indicate that the calibration risk is relatively small in this case as well. In Table 4.5 we see that all 168 modelled call prices are inside the bid-ask spread, and in Figure 4.5 we have plotted the call prices for four maturity dates.

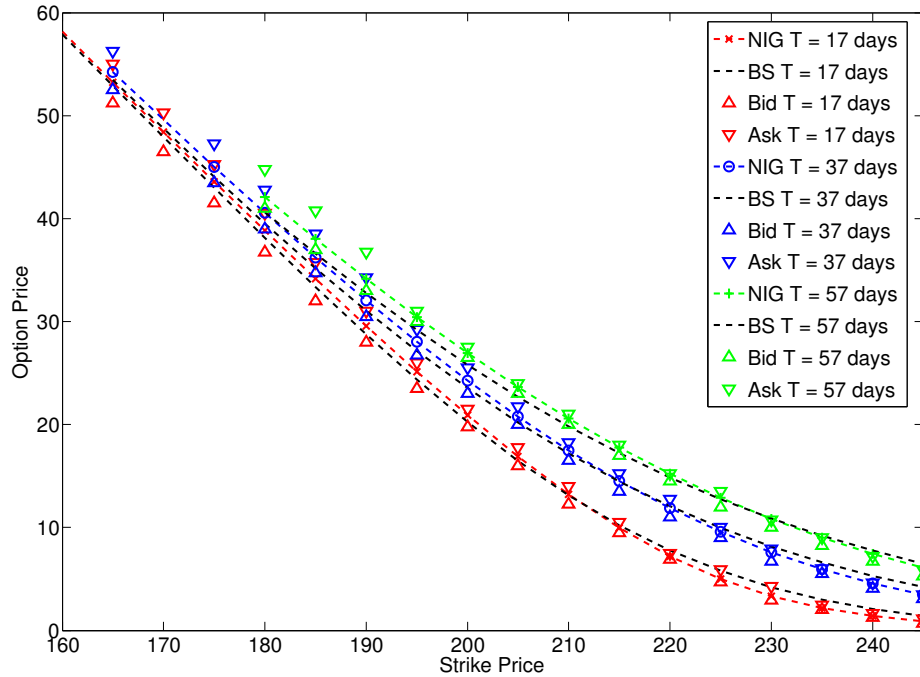


Figure 4.4: OBX call option prices for our data chain. The NIG prices are calculated with the ARPE-parameters and are shown as colored dashed lines, while the Black and Scholes (BS) prices are shown as black dashed lines.

The data chain we collected for the Dow Jones Industrial Average, contained for some unknown reason mainly deep-in-the-money^{4.3} call option prices, and as we can see from Figure 4.5, the option prices are then almost linear. This is perhaps one of the reasons for the very good calibration results. We also see that the bid-ask spread is relatively wide for these options, which again may explain why all 168 modelled option prices fall inside the spread.

One last thing worth noticing about the parameters in Table 4.7, is that all four error measure optimization results lie on the boundary $|\beta| \leq \alpha$. This may indicate that our solutions most likely are not the global solutions of the unconstrained optimization problem. However, we see that our calibration results are very good for this option chain.

So far, we have seen satisfying results for OBX and Dow Jones. However, for Dow Jones the option chain collected limited our analysis to deep-in-the-money call options, which are almost linear. In Table 4.8 we see that the calibration results for Coca Cola are not nearly as good as OBX and Dow Jones. First of all, we see that the $ARPE_{spread}$

^{4.3}term used when the intrinsic value, which is the value of exercising the option now, is relatively large

	NIG				Black Scholes	
	α	β	δ	Error	σ	Error
$ARPE_{spread}$	0.5836	-0.5836	0.0948	5.03%	0.2762	12.72%
ARPE	0.5737	-0.5737	0.0875	0.75%	0.2761	1.66%
RMSE	0.5809	-0.5809	0.0932	0.2572	0.2859	0.7203
AAE	0.5653	-0.5653	0.0838	0.2209	0.2788	0.5799

Table 4.7: Implied parameters for Dow Jones Industrial Average option prices. The market data is collected from Bloomberg on 27th of April 2009, and the index was in 80.25\$. The prices are for 8 different maturities ranging from $T_1 = \frac{14}{250}$ to $T_8 = \frac{424}{250}$. For these maturity dates, we have used the US Treasury bill rates (ranging from 0.07% to 0.90%) as the risk-free interest rate, r .

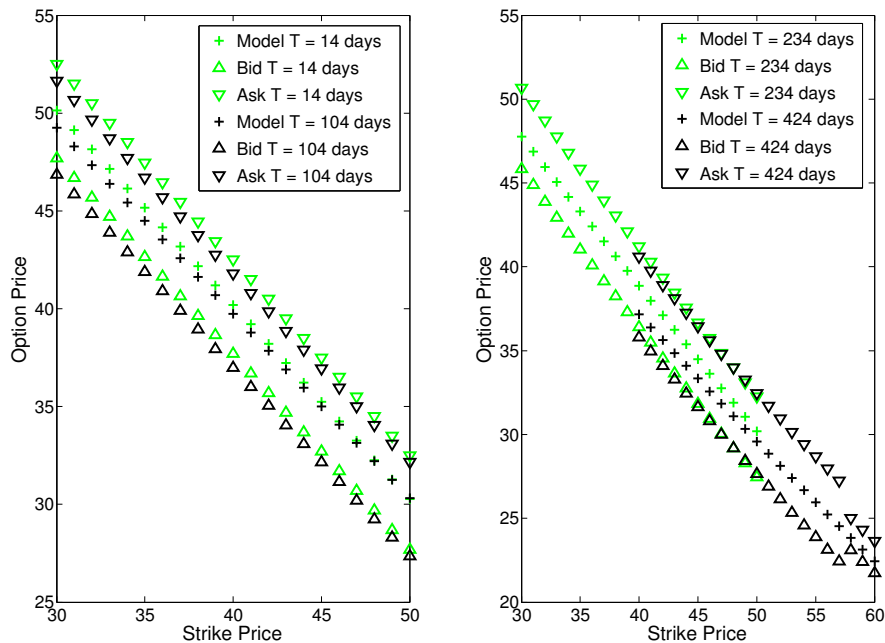


Figure 4.5: Call option prices on the Dow Jones Industrial Average for four of the maturity dates in our data chain. The modelled NIG prices are calculated with the ARPE-parameters.

results have an error of 73%, while the error for the $ARPE$ calibration is only 7.8%. To find the explanation for this, we have in Table 4.9 calculated the average bid-ask

spread for the four option chains. From this table we see that for the Coca Cola case, the average is far less than 1, which together with Eq. (4.5) explain the high error for the $ARPE_{spread}$ calibration. Secondly, we see that the RMSE parameters deviate much from the other parameters, which may indicate that the calibration risk for this data chain is considerable. However, looking at the modelled call option prices in Appendix B, this large deviation in the parameters is not reflected in the output prices. Hence the calibration risk might not be as large as the parameter deviation would indicate.

So far we have not commented on the calibration errors of RMSE and AAE since they may be a bit hard to analyse without knowing the option prices. Is an error of 0.25 significant? In an attempt to get a quick intuition on this, we have in Table 4.9 also calculated the average mid prices. From this table, we see that the average mid price for Dow Jones and OBX are three to four times as large as Coca Cola. This indicates that an RMSE error of 0.25 for Coca Cola is considerably worse than for Dow Jones or OBX. The average mid prices in Table 4.9 might also explain why the ARPE errors for OBX and Dow Jones are smaller than for Coca Cola and StatoilHydro.

	NIG				Black Scholes	
	α	β	δ	Error	σ	Error
$ARPE_{spread}$	26.4502	-17.3990	0.8872	73.47%	0.2536	109.22%
ARPE	18.7369	-7.3969	0.9497	7.81%	0.2378	8.56%
RMSE	89.0449	-55.8721	2.6938	0.2501	0.2503	0.2532
AAE	17.2344	-8.4112	0.8265	0.1777	0.2485	0.1975

Table 4.8: Implied parameters for Coca Cola option prices. The market data is collected from Bloomberg on 27th of April 2009, and the spot price was 42.24\$. The prices are for 6 different maturities ranging from $T_1 = \frac{13}{250}$ to $T_6 = \frac{435}{250}$. For these maturity dates, we have used the US Treasury bill rates (varying from 0.07% to 0.90%) as the risk-free interest rate, r .

	OBX	DJI	KO	STL
Spread	2.0637	4.7307	0.2743	1.4576
Mid Price	24.7858	33.5228	7.6878	12.6451

Table 4.9: Average bid-ask spread and average mid price for the market call option prices.

At last, we have made the same calculations for StatoilHydro, with the resulting parameters and errors shown in Table 4.10. Here we again see relatively large deviations in the parameters, especially for the RMSE. From Table 4.5 we also see that for RMSE, less than half of the modelled prices fit inside the bid-ask spread. All errors are significantly higher than for the three other option chains, and one might be tempted to conclude

that StatoilHydro is hard to model because of its relatively small size. Looking at the output prices in Appendix B, we also see that this is the first case where the option prices seem to deviate significantly for some strikes and maturities. One last thing worth noticing, is that the normality assumption from Black-Scholes seem to be almost as good as the NIG model. For the AAE calibration results, the error is actually smaller for the Black-Scholes calibration results.

	NIG				Black Scholes	
	α	β	δ	Error	σ	Error
ARPE _{spread}	9.7278	-3.2261	1.1524	41.48%	0.3551	47.18%
ARPE	3.8591	-2.4536	0.5722	17.84%	0.3569	20.24%
RMSE	326.5750	-318.7785	0.4351	0.7103	0.3415	1.1781
AAE	13.7671	-6.0458	1.3167	0.8450	0.3511	0.7545

Table 4.10: Implied parameters for StatoilHydro option prices. The market data is collected from Bloomberg on 27th of April 2009, and the spot price was 120.30 NOK. The prices are for three different maturities; $T_1 = \frac{37}{250}$, $T_2 = \frac{102}{250}$ and $T_3 = \frac{167}{250}$. For these three maturity dates, we have used the 3 month and 6 month Norwegian Treasury bill rates (1.92% and 1.81%) as the risk-free interest rate, r .

To conclude this section, it seems as OBX, Dow Jones and Coca Cola are reasonably well fitted to the market data with the NIG market model, while StatoilHydro again seem to be harder to model accurately. It might be hard to make this conclusion definitively, since the data we base our analysis on, are somewhat incomplete. This is perhaps the biggest drawback with the implied parameter calibration method, since the available market data may often not be completely adequate. One other issue working in the implied parameter calibration methods disfavour, is that the parameters may often vary a lot depending on which options we want to price. For instance, we are in this thesis mainly focusing on pricing exotic options, and then we should use exotic option market data to find our parameters. However, as mentioned earlier, these exotic options are traded OTC, which means that the option prices are not publicly listed and therefore are hard to collect unless you are actually buying.

4.5 Historical vs. Implied

We have now tried two different approaches to fit our NIG market model to data in the market. The conclusion so far have been that our model seem to fit market data better than the widely used Black-Scholes model proposed by Black & Scholes (1973) and that we get reasonable prices as long as we look at large, liquid indices or stocks. But which one should we choose? The historical approach has the advantage that the data are easy to collect and that we often have a sufficient amount of data. The drawback is that the history seldom contain the investors future expectation regarding skewness, return etc.

The implied approach capture this property, but has the drawback that the amount of market data are limited, and if we want to price exotic options there may not be any market data available at all. For instance, for the Dow Jones call option data above, it may not be satisfying to have deep-in-the-money market data to calculate implied parameters for the price of an out-of-the-money call option.

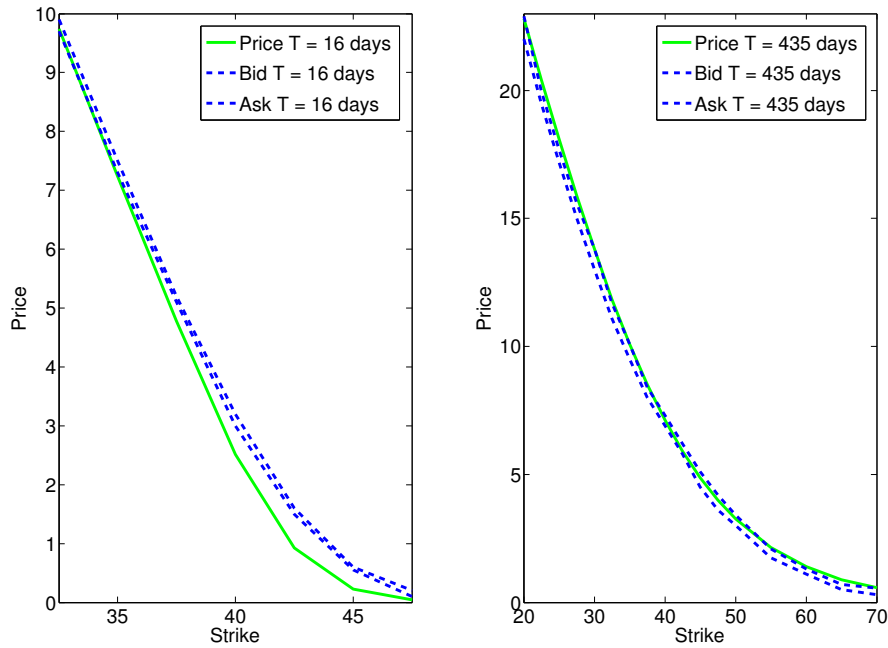


Figure 4.6: Coca Cola call option prices using the market bid/ask prices and the parameters estimated from historical data (Table 4.2)

In many cases, we are then forced to use the historical parameters due to the lack of market option data. Figure 4.6 show the result from pricing Coca Cola call options using the parameters obtained by historical calibration. If we look at the figure, we see that for the long maturity, $T = \frac{435}{250}$ years, the results are relatively satisfying. For several strike prices, the price end up inside the bid-ask spread. For the short maturity, which often has more liquidity and hence smaller bid-ask spread, the results are considerably worse. A possible intermediate approach between the historical and implied parameter estimation, which we will not consider in this thesis, might be to assign the newer history more weight than the old history.

4.6 Sensitivity Analysis

So far we have tried our best to fit the NIG market model to market data, both with historical data and with implied parameters. The results of this calibration is important for market makers when they are modelling financial derivatives. But there is one aspect of the modelling that is as important, if not more important, for practitioners; How does the price of the option change with respect to changes in the different parameters? These change measures are in finance referred to as the Option Greeks, or just the Greeks. The reason why the option Greeks are of such practical importance, is that the financial institutions that sell the derivatives want to remove as much as possible of the risk associated with the options. They do not want to carry the risk themselves; they want to hedge their positions.

When using a theoretical model, in our case the NIG market model, there are in practice a certain risk associated with each parameter in the model. In the Black-Scholes model, the option's sensitivity to small changes in the implied volatility is called the Vega or the Zeta. In the NIG market model we have three parameters that play the role of the volatility in the Black-Scholes model. It is therefore of great interest to see how the option prices change when these parameters change. We have not seen Option Greeks in the NIG market model context before, and hence we define them with the following names:

$$\begin{aligned} \text{AlphaZeta} &= \frac{\partial V}{\partial \alpha} \\ \text{BetaZeta} &= \frac{\partial V}{\partial \beta} \\ \text{DeltaZeta} &= \frac{\partial V}{\partial \delta} \end{aligned}$$

where V is the option value. Naturally, there are no analytic expression for these Greeks, and we will calculate them numerically using the following finite difference scheme:

$$\begin{aligned} \frac{\partial V}{\partial \alpha} &\approx \frac{V(\alpha + \Delta\alpha) - V(\alpha - \Delta\alpha)}{2\Delta\alpha} \\ \frac{\partial V}{\partial \beta} &\approx \frac{V(\beta + \Delta\beta) - V(\beta - \Delta\beta)}{2\Delta\beta} \\ \frac{\partial V}{\partial \delta} &\approx \frac{V(\delta + \Delta\delta) - V(\delta - \Delta\delta)}{2\Delta\delta} \end{aligned}$$

Figures 4.7-4.9 show the visualization of these three Greeks for a Vanilla call option with $X = 100$, $r = 0.05$, $\alpha = 10$, $\beta = -4$ and $\delta = 1$. To understand what these plots tell us, it can be helpful to take a look at Figure 3.3 to see how a positive change in one of the parameters influences the PDF, and hence also the probability distribution for the daily log returns.

Before looking at Figure 4.7, we recall from earlier that if we increase the α -value, then the PDF will be more peaked with thinner tails. In other words; the probability of small daily variations increases, while the probability of larger daily variations decreases. This is more or less analogous to decreasing the volatility in the Black-Scholes model.

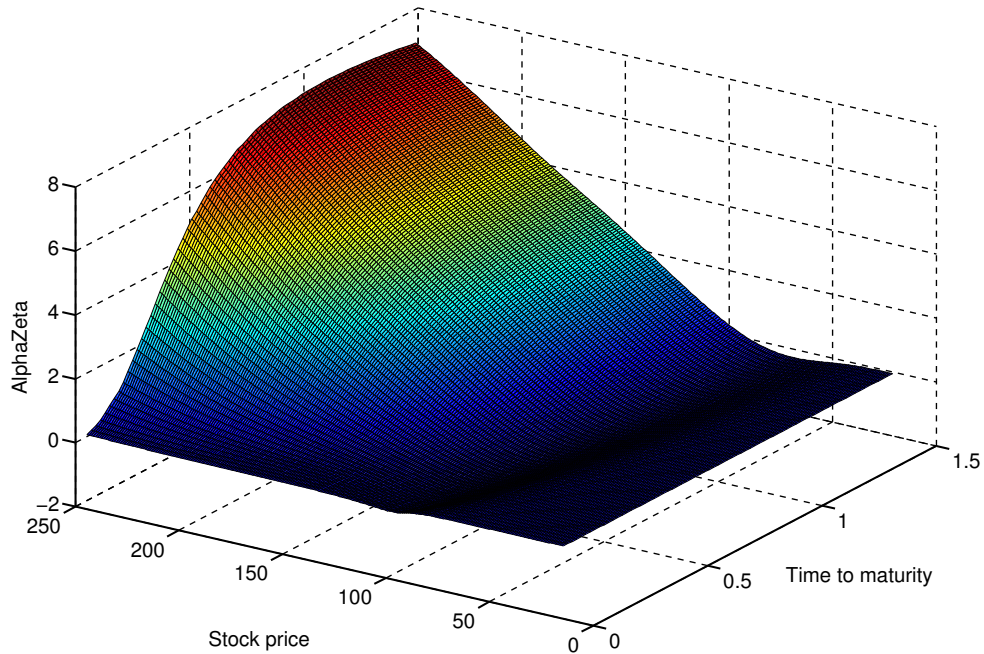


Figure 4.7: Graphical visualization of $\frac{\partial V}{\partial \alpha}$ as a function of time to maturity $T - t$ and stock price S_t .

Since $\beta = -4$, the PDF is a little skew towards negative daily variations, and so smaller tails will in this case be a desired change for the holder of the call option. Looking at Figure 4.7, we see that around $S_t = 250$ and for long maturity dates we have the largest value of $\frac{\partial V}{\partial \alpha}$. This is because these stocks have the most to gain from loosing some of the probability of negative daily returns, since they have a high intrinsic value^{4.4} and a long time until expiry.

In the other end of the scale, where S_t is around the strike price X , it is not a good thing for the call option holder that the tails decrease. If for example $S_t = 99$ and there are 5 days until maturity, then you do not loose any money on the call option if the stock price falls. Hence, you will have a greater probability of earning money on the call option if the probability of larger daily returns increases. This is why $\frac{\partial V}{\partial \alpha}$ is negative for stock prices around $X = 100$.

Figure 4.8 looks quite similar to Figure 4.7, which seems reasonable since small changes in α and β have a relatively similar effect on the PDF (see Figure 3.3). And so the explanation behind $\frac{\partial V}{\partial \beta}$ follows the same reasoning as for $\frac{\partial V}{\partial \alpha}$.

^{4.4}the value of an option if it were cashed in now

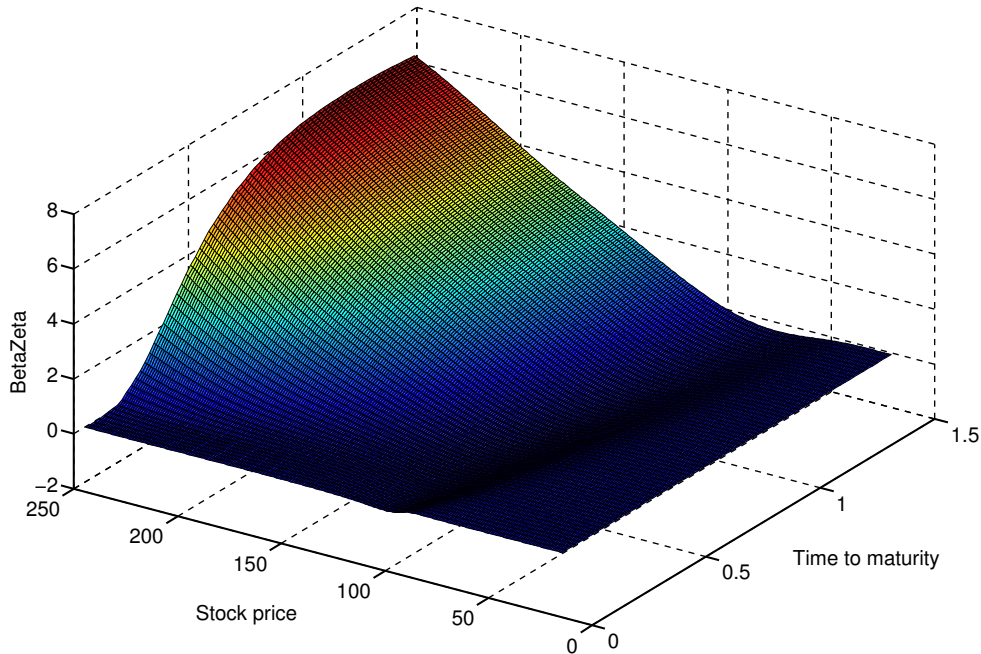


Figure 4.8: Graphical visualization of $\frac{\partial V}{\partial \beta}$ as a function of time to maturity $T - t$ and stock price S_t .

Considering Figure 4.9, we see that the picture is very different from the two other plots. This is because a small change in δ , can make a large impact on the PDF (ref. Figure 3.3). So a small positive change in the δ -value, may result in a large increase in the probability of large daily variations. This leads to an opposite reasoning of the one above. Around the strike price $X = 100$ we see a positive value of $\frac{\partial V}{\partial \delta}$ since the probability that the stock price will end up above $X = 100$, is greater. For large values of S_t and $T - t$ we have a relatively large negative value of $\frac{\partial V}{\partial \delta}$ since the probability that the stock price will fall below $X = 100$ has increased relatively much.

Looking at the magnitude of these Greeks, we see that for *AlphaZeta* and *BetaZeta* with $S_t = 250$ and $T - t = 1.5$, we get values of around 7. Compared to the intrinsic value of 150, this is a relatively large change, almost 5%. However, if we look at the *DeltaZeta*, this ratio is almost 40%. These values are very high, but it should be noticed that call options with an intrinsic value of 150% of the strike price are very rare.

The purpose of this section was not to make a conclusion about the sensitivity properties of the NIG market model. To make this conclusion is hard without having similar results to compare with. However, it is important to be aware of this aspect of financial modelling when comparing different models.

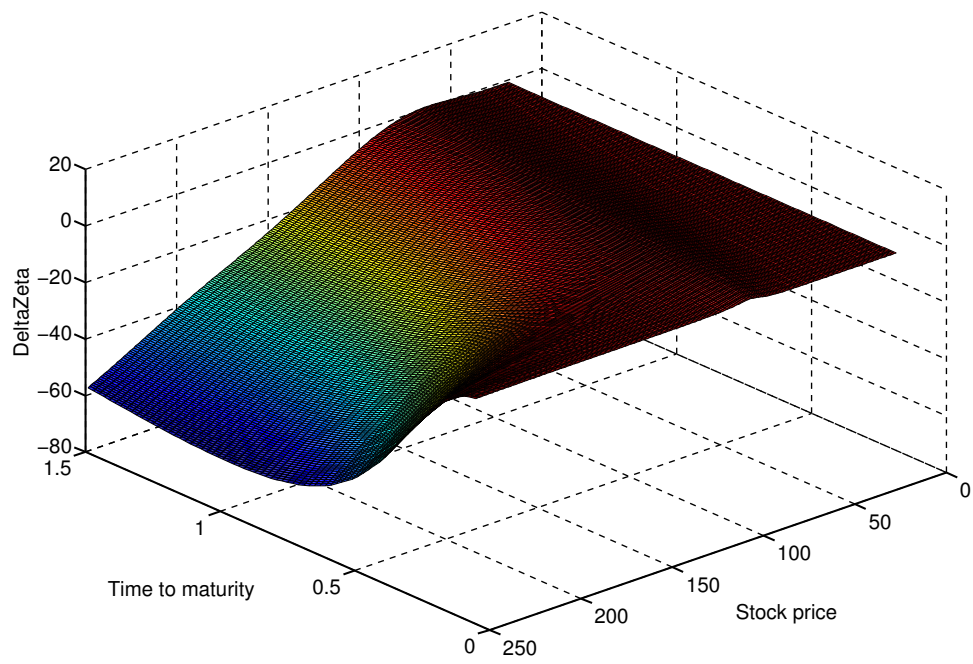


Figure 4.9: Graphical visualization of $\frac{\partial V}{\partial \delta}$ as a function of time to maturity $T - t$ and stock price S_t .

Chapter 5

Option Pricing and Numerical Path Integration

We have now shown how the NIG distributed density function can be fitted nicely to empirical market data, and we will in this section focus on pricing several options under the NIG market model using numerical path integration with given parameters. Unlike option pricing with the Black-Scholes framework, we do not have any analytic formula for option pricing under the NIG market model. And since the options we are pricing in this thesis are typically traded OTC, we have no market prices to compare with either. Instead we will use Monte Carlo simulation to verify our answers. Pricing European options with Monte Carlo simulation is reasonably straight forward, and hence if the results from the MC and path integration are equal, it should be a good indication of correctness.

To explain the path integration approach, we will mainly focus on the barrier option situation, since this easily can be extended to both the Vanilla case and the lookback case. An important practical issue when pricing these exotic options, is how often the barrier, the maximum/minimum or the average is monitored. These monitoring times can either be discrete or continuous. In equities and commodities, it is said to be most common with discrete monitoring, while in FX markets continuous monitoring is the standard. The discrete monitoring value is often set to the official daily closing price, and unless otherwise specified, we will use daily monitoring in this chapter.

5.1 Model

We will in this thesis mainly focus on pricing discretely monitored lookback options and Asian options, which we briefly explained in chapter 2. However, since the underlying probability calculations for the barrier option case is the foundation for pricing lookback option, we will use some time to verify that our barrier option calculations are correct. The payoff for the lookback and Asian options at maturity T are summarized in Table 5.1 and Table 5.2.

From the tables, we see that in addition to call and put options, there are two types

		Payoff at maturity T
Fixed strike	Call	$\max(M_T - X_{LB}, 0)$
	Put	$\max(X_{LB} - m_T, 0)$
Floating strike	Call	$\max(S_T - m_T, 0)$
	Put	$\max(M_T - S_T, 0)$

Table 5.1: Payoff for fixed and floating strike lookback options. M_T and m_T denotes the maximum and minimum, respectively, of the underlying asset, S_T , at maturity T . X_{LB} denotes the fixed strike price.

of lookback/Asian options; fixed strike and floating strike. For the fixed strike options, the payoff only depend on the maximum/minimum or the average, while for the floating strike the payoff in addition depend on the price of the underlying asset at maturity, S_T .

		Payoff at maturity T
Fixed strike	Call	$\max(A_T - X_A, 0)$
	Put	$\max(X_A - A_T, 0)$
Floating strike	Call	$\max(S_T - A_T, 0)$
	Put	$\max(A_T - S_T, 0)$

Table 5.2: Payoff for fixed and floating strike Asian options. A_T denotes the arithmetic average of the underlying asset, S_T , at maturity, T . X_A denotes the fixed strike price.

The strike price for the barrier options is denoted by X_B , the strike price for the lookback options X_{LB} and the strike price for the Asian option X_A . Further, the upper barrier is denoted by U , lower barrier by L , and a constant interest rate r is assumed. Initial stock price is denoted S_0 . The maximum of the stock price S_t up to and including time t is denoted M_t and the corresponding minimum is denoted m_t . The arithmetic average of the stock price S_t up to time t and including S_0 is denoted A_t . For the barrier options, the up-and-out-down-and-out option becomes worthless if the stock price S_t is larger than or equal to the upper barrier U or less than or equal to the lower barrier L at any of the monitoring times. If the stock price stays within the barrier interval at every monitoring time until maturity, the barrier call option value is $\max(0, S_T - X_B)$ at the maturity time T . The underlying asset is monitored at N different times τ_j , $j = 1, \dots, N$ such that $0 < \tau_1 < \tau_2 < \dots < \tau_{N-1} < \tau_N = T$. For the initial conditions to make sense, we must have $X < U$ and $L < S_0 < U$.

To find the price of the barrier option, or the underlying cumulative distribution function for the lookback option (see section 5.2.1), we need the conditional probability distribution function for the stock price at maturity, T , conditional on the property that

the stock price, S_t , has stayed within the barrier interval the entire time $t \in [0, T]$

$$\tilde{S}_{\tau_i} = S_{\tau_i} \cdot I\{L < S_{\tau_j} < U \quad \forall \quad 0 < \tau_j \leq \tau_i\} \quad (5.1)$$

Throughout this chapter, we will work with the process $Z_t = \log(S_t/S_0)$ instead of S_t . Denoting this conditional probability density function for \tilde{S}_T expressed in terms of Z_T by $q_N(z)$, it is clear that the double barrier call option value is given by:

$$c = e^{-rT} \int_{\log(X/S_0)}^{\log(U/S_0)} (S_0 e^z - X) q_N(z) dz \quad (5.2)$$

From this it follows that the only difference between pricing Vanilla options and barrier options, is that we need to calculate the function $q_N(z)$ on a limited interval in space. It should however be noted that since we for a Vanilla option only are interested in what happens at the end, it is more inefficient to price a Vanilla option this way. From the definition of $q_N(z)$ above, we get

$$q_N(z) = \int_{\log(L/S_0)}^{\log(U/S_0)} \cdots \int_{\log(L/S_0)}^{\log(U/S_0)} p_{N|N-1}(z|z_{N-1}) \cdots p_{2|1}(z_2|z_1) p_{1|0}(z_1|0) dz_1 \cdots dz_{N-1} \quad (5.3)$$

where $p_{i|i-1}(z|z_{i-1})$ denotes the transition probability density function of Z_{τ_i} from τ_{i-1} to τ_i given $Z_{\tau_{i-1}} = z_{i-1}$. When this multiple integral is solved, we obtain the option price we need from Eq. (5.2). In the case of an up-and-out option, we just set $L = 0$, and for down-and-out we have $U = \infty$. If we want to price a Vanilla option, we set $L = 0$ and $U = \infty$. Setting $L = 0$ will produce a singularity, since we have chosen to work with Z_t instead of S_t . However, this problem is nearly a theoretical one, and we lose no accuracy by instead considering $\lim_{L \rightarrow 0} \log(L/S_0) = -\infty$ as the lower limit.

Similarly, the price of fixed strike lookback call and put options, given that the probability density functions $f_{M_N}(\eta)$ and $f_{m_N}(\xi)$ exists, are given by

$$\begin{aligned} C_{fixed}(X, T) &= e^{-rT} \int_{X_{LB}}^{\infty} (\eta - X_{LB}) f_{M_N}(\eta) d\eta \\ P_{fixed}(X, T) &= e^{-rT} \int_0^{X_{LB}} (X_{LB} - \xi) f_{m_N}(\xi) d\xi \end{aligned} \quad (5.4)$$

In the case of a floating strike lookback option, $X_{LB} = S_N$. Hence, if the joint probability density functions $f_{M_N, Z_N}(\eta, z)$ and $f_{m_N, Z_N}(\xi, z)$ are known, the price of the floating call and put options can be calculated by

$$\begin{aligned} C_{floating}(T) &= e^{-rT} \int_{-\infty}^{\infty} \int_0^{S_0 e^z} (S_0 e^z - \xi) f_{m_N, Z_N}(\xi, z) d\xi dz \\ P_{floating}(T) &= e^{-rT} \int_{-\infty}^{\infty} \int_{S_0 e^z}^{\infty} (\eta - S_0 e^z) f_{M_N, Z_N}(\eta, z) d\eta dz \end{aligned} \quad (5.5)$$

For the case of the floating strike Asian option, given that the joint probability density function $f_{Z_N, A_N}(z, y)$ is known, the price is given by

$$\begin{aligned} C_{floating}(T) &= e^{-rT} \int_{-\infty}^{\infty} \int_0^{S_0 e^z} (S_0 e^z - y) f_{Z_N, A_N}(z, y) dy dz \\ P_{floating}(T) &= e^{-rT} \int_{-\infty}^{\infty} \int_{S_0 e^z}^{\infty} (y - S_0 e^z) f_{Z_N, A_N}(z, y) dy dz \end{aligned} \quad (5.6)$$

where $\rho = 1$ if call option and $\rho = -1$ if put option. For the fixed strike Asian option, we only need the marginal density function, f_{A_N} , which we get by integrating out of the joint density function $f_{Z_N, A_N}(z, y)$. The price of a fixed strike Asian call or put option is then given by

$$\begin{aligned} C_{fixed}(X, T) &= e^{-rT} \int_{X_A}^{\infty} (y - X_A) f_{A_N}(y) dy \\ P_{fixed}(X, T) &= e^{-rT} \int_0^{X_A} (X_A - y) f_{A_N}(y) dy \end{aligned} \quad (5.7)$$

5.2 Implementation

Skaug & Naess (2007) showed that whenever the transition probability density $p_{i|i-1}(z | z')$ for $i = 1, \dots, N$ is known, we can obtain $q_N(z)$ from Eq. (5.3). When the NIG market model is used, the transition probability density can be derived from the PDF given in (3.11). Assuming equidistant monitoring times, with $\Delta\tau = \tau_i - \tau_{i-1}$, it is found that

$$p_{1|0}(z|0) = q_1(z) = f(z; \Delta\tau) \quad (5.8)$$

and

$$p_{i|i-1}(z|z') = f(z - z'; \Delta\tau) \quad , \quad i = 2, \dots, N \quad (5.9)$$

Hence it follows that, for $i = 2, \dots, N$,

$$q_i(z) = \int_{\log(L/S_0)}^{\log(U/S_0)} f(z - z'; \Delta\tau) q_{i-1}(z') dz' \quad (5.10)$$

All the integrals are computed using numerical techniques. Since the NIG density function contains the modified Bessel function of the second kind, it contains a hidden integral, but both Matlab and Excel have this as a built-in function. Still, calculating the Bessel function is a significant part of the total computation time, when we apply the numerical path integration method on the NIG market model.

When the function $q_N(z)$ has been calculated for a given set of barriers, finding the option price for any given strike X and maturity $T = \tau_i$ for $i \leq N$, or any given binary option, is done just as fast as finding the value of a Vanilla call option by Eq. (5.2).

5.2.1 Lookback Options

So far we have looked at how we can price different barrier options using the numerical path integration, and this approach will make up the foundation when pricing the lookback options. For the lookback call (put) option with fixed strike, the only variable element is the maximum (minimum) of the underlying asset S_t during the lifetime of the option. For the floating strike lookback options there are two variable elements, the price of the underlying asset at maturity T , S_T , and the minimum (maximum) of the underlying asset. We will first focus on the fixed strike lookback option.

Fixed strike lookback options

In order to price a fixed strike lookback option, we see from Eq. (5.4) that we need to find the marginal density function f_{M_N} (f_{m_N} for put option). If we have already priced a floating strike lookback option, we have obtained the joint probability density function f_{M_N, Z_N} (f_{m_N, Z_N} for a call option), and then it should be straightforward to find the marginal density function by just integrating out. However, this is an unnecessary time-consuming approach, since we have to use two dimensional interpolation etc.

The function $q_N(z)$ from Eq. (5.3) is the probability density function for the logarithm of the stock at time $T = \tau_N$ given that the stock has not crossed the upper or lower barrier level during its lifetime. Focusing on an up-and-out call option (letting $L = 0$), we see that the probability that the stock, and hence the maximum of the stock, is less than $U = \eta$, is given by

$$\begin{aligned} F_{M_N}(\eta) &= P(M_N \leq \eta) = P(S_{\tau_1} \leq \eta \cap S_{\tau_2} \leq \eta \cap \dots \cap S_{\tau_N} \leq \eta) \\ &= P(\tilde{S}_T \leq \eta) = \int_{-\infty}^{\log(\eta/S_0)} q_N(z) dz \end{aligned} \quad (5.11)$$

where \tilde{S}_{τ_i} is defined in (5.1). This calculation we repeat for a range of different barriers, $\eta \in \{\eta_{min}, \eta_{min} + d\eta, \dots, \eta_{max} - d\eta, \eta_{max}\}$, where η_{min} and η_{max} are chosen such that $F_{M_N}(\eta_{min}) \approx 0$ and $F_{M_N}(\eta_{max}) \approx 1$. As an example of what is meant by this, in the graph in Figure 5.1 these limits would typically be 90 and 150, respectively.

Calculating $F_{M_N}(\eta)$ for grid with small values of $d\eta$, is very time-consuming, and hence we should choose our grid carefully. If we look at the CDF/PDF in Figure 5.1, we see that the most critical part is around S_0 and so it is crucial that the gridsize $d\eta$ is small in the range around S_0 . We have therefore chosen a relatively much smaller gridsize in this range, and then applied piecewise cubic interpolation in order to get an accurate representation of the cumulative distribution function. The probability density function is then obtained by taking the derivative of $F_{M_N}(\eta)$

$$f_{M_N}(\eta) = \frac{dF_{M_N}(\eta)}{d\eta}$$

When we have calculated this probability density function, the price of a fixed strike lookback call option is given by Eq. (5.4). In the same manner, we can find the cumulative

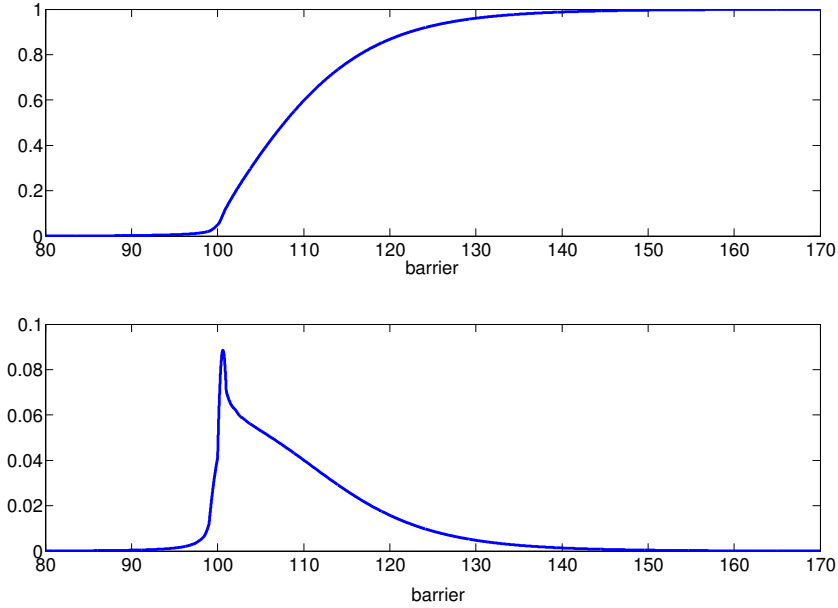


Figure 5.1: A realization of the cumulative distribution function $F_{M_N}(\eta)$ (upper graph) and the corresponding $f_{M_N}(\eta)$ (lower graph) as functions of the barrier η . In this example, $S_0 = 100$.

distribution function for the minimum, m_N , by

$$\begin{aligned}
 F_{m_N}(\eta) &= P(m_N \leq \eta) = 1 - P(m_N \geq \eta) \\
 &= 1 - P(S_{\tau_1} \geq \eta \cap S_{\tau_2} \geq \eta \cap \dots \cap S_{\tau_N} \geq \eta) \\
 &= 1 - P(\tilde{S}_T \geq \eta) = 1 - \int_{\log(\eta/S_0)}^{\infty} q_N(z) dz
 \end{aligned} \tag{5.12}$$

The probability density function $f_{m_N}(\eta)$ is obtained by taking the derivative as above, and the fixed strike lookback put option price is then given by Eq. (5.4).

Floating Strike Lookback Option

For the floating strike lookback option we also need to calculate the density function for the maximum (minimum), but in addition we need the probability density function for the stock price at maturity. These two variables are not independent, and so we can not perform these two calculations independently. Because of this, we need to calculate the joint density function $f_{M_n, Z_n}(\eta, z)$. As we did for the fixed strike case, we do this by first calculating the joint cumulative distribution function, $F_{M_n, Z_n}(\eta, z)$, and then take the partial derivatives. As mentioned in the introduction, the maximum at time τ_n is defined as $M_n = \max(S_1, S_2, \dots, S_n)$ where the stock price at time τ_n is included.

The joint CDF for the maximum of the stock, M_N , and the stock price S_N , is given by

$$\begin{aligned} F_{M_N, Z_N}(\eta, z) &= P(M_N \leq \eta, Z_N \leq z) \\ &= \begin{cases} P(M_N \leq \eta) & \text{if } S_0 e^z \geq \eta \\ P(M_N \leq \eta, Z_N \leq z) & \text{if } S_0 e^z < \eta \end{cases} \end{aligned} \quad (5.13)$$

where $P(M_N \leq \eta)$ is given in Eq. (5.11). Considering the first case, when $S_0 e^z \geq \eta$, we see that the joint CDF is independent of z , which should be intuitive, since S_n can not be greater than M_n when the maximum at τ_n includes S_n . The joint PDF is found by taking the partial derivatives (see (5.16)), and hence we get

$$f_{M_n, Z_n}(\eta, z) = 0 \quad \text{for } S_0 e^z > \eta \quad (5.14)$$

For $S_0 e^z < \eta$ we should expect a dependence of both z and η . We get

$$F_{M_N, Z_N}(\eta, z) = P(M_N \leq \eta, Z_N \leq z) = \int_{-\infty}^z q_N^\eta(z') dz' \quad (5.15)$$

where $q_N^\eta(z)$ depends on η through the fact that it is the PDF for $\tilde{Z}_N = Z_N \cdot I\{Z_j \leq \log(\frac{\eta}{S_0}) \quad \forall \quad 0 < \tau_j \leq \tau_N\}$, and we have denoted this with the additional superscript.

So the area of interest, is the triangular spanned by $\eta_{min} \leq \eta \leq \eta_{max}$ and $S_0 e^z \leq \eta$, where the joint probability density function is nonzero. As we did for the fixed strike case, we then calculate $F_{M_N, Z_N}(\eta, z)$ from Eq. (5.15) for $\eta \in \{\eta_{min}, \eta_{min} + d\eta, \dots, \eta_{max} - d\eta, \eta_{max}\}$ and $0 \leq S_0 e^z \leq \eta$, where η_{min} and η_{max} are chosen such that $F_{M_N}(\eta_{min}) = \int_{-\infty}^{\infty} F_{M_N, Z_N}(\eta_{min}, z) dz \approx 0$ and $F_{M_N}(\eta_{max}) = \int_{-\infty}^{\infty} F_{M_N, Z_N}(\eta_{max}, z) dz \approx 1$. The two dimensional probability density function $f_{M_n, Z_n}(\eta, z)$ is then given by

$$f_{M_n, Z_n}(\eta, z) = \frac{\partial}{\partial z} \left(\frac{\partial F_{M_n, Z_n}(\eta, z)}{\partial \eta} \right) \quad (5.16)$$

When we have calculated this joint PDF, the price of a floating strike lookback put option is given by Eq. (5.5).

When implementing this in Matlab, our Barrier function return $q_N(z)$ for a given barrier η . This is the probability density function for $z = \log(\frac{s}{S_0})$ given that the stock (and hence the maximum) has stayed beneath η at all points in time. To find $F_{M_N, Z_N}(\eta, z)$ from this, we have to first calculate the integral in Eq. (5.15) for all $z \leq \log(\frac{\eta}{S_0})$ and then later take the partial derivative with respect to z again. This is an inefficient way to do it, and we may get some unnecessary extra numerical errors. We will instead calculate

$$\frac{\partial}{\partial z} F_{M_N, Z_N}(\eta, z) = q_N^\eta(z)$$

for all $\eta_{min} \leq \eta \leq \eta_{max}$ and $0 \leq S_0 e^z \leq \eta$, and then find the joint probability density function by taking the partial derivative with respect to η .

In the same manner, we find the joint probability density function for the minimum by calculating

$$\frac{\partial}{\partial z} F_{m_N, Z_N}(\xi, z) = 1 - q_N^\xi(z)$$

where the superscript ξ now denotes the lower barrier for the stock price during its lifetime. The joint PDF is then found by taking the partial derivative with respect to ξ . With the resulting joint PDF, we can find the price of a floating strike lookback call option from Eq. (5.5).

5.2.2 Asian Options

When pricing Asian options, we are interested in the density function at maturity of two variables, the price of the underlying asset S_N and the average price A_N . We now propose two different approaches for finding $f_{Z_N, A_N}(z, y)$ from Eq. (5.6).

Approach 1

Lets define the running average process as

$$A_n = \frac{1}{n+1} \sum_{i=0}^n S_i = \frac{S_0}{n+1} \sum_{i=0}^n e^{Z_i} \quad \text{for } n = 1, \dots, N \quad (5.17)$$

which also can be written as

$$A_n = \frac{n}{n+1} A_{n-1} + \frac{S_0}{n+1} e^{Z_n} \quad (5.18)$$

Using (5.18) and the transition probability for z , $p_{i|i-1}(z|z')$, from Eq. (5.9), we can define a transition probability distribution for $\{(Z_i, A_i)\}_{i=0}^N$ as follows

$$f_{Z_i, A_i | Z_{i-1}, A_{i-1}}(z, y | z', y') = \delta \left(y - \frac{1}{i+1} y' - \frac{S_0}{i+1} e^z \right) \cdot p_{i|i-1}(z | z') \quad (5.19)$$

where δ denotes the Dirac measure, that is, $\delta(B) = 1$ if $0 \in B$ for any set B , and $\delta(B) = 0$ if $0 \notin B$.

Assuming that the random variables Z_{i-1} and A_{i-1} are specified by a joint PDF $f_{Z_{i-1}, A_{i-1}}$, the law of total probability then implies that the joint probability density function of Z_i and A_i , f_{Z_i, A_i} , is given by

$$f_{Z_i, A_i}(z, y) = \int_{-\infty}^{\infty} \int_0^{\infty} f_{Z_i, A_i | Z_{i-1}, A_{i-1}}(z, y | z', y') f_{Z_{i-1}, A_{i-1}}(z', y') dy' dz' \quad (5.20)$$

From this relation and Eq. (5.19), it can now be shown that the joint PDF, f_{Z_i, A_i} , is given by

$$f_{Z_i, A_i}(z, y) = \frac{i+1}{i} \int_{-\infty}^{\infty} f(z - z'; \Delta\tau) f_{Z_{i-1}, A_{i-1}} \left(z', \frac{i+1}{i} y - \frac{S_0}{i} e^z \right) dz' \quad (5.21)$$

where $f(z, t)$ is the probability density function given in (3.11).

With given numerical initial conditions $Z_0 = 0$ and $A_0 = S_0$, the initial joint PDF $f_{Z_0, A_0}(z, y)$ is not a proper PDF, but rather a product of delta distributions of singular type. Formally it can be written as

$$f_{Z_0, A_0}(z, y) = \delta(z)\delta(y - S_0) \quad (5.22)$$

However, this kind of initial distribution is not convenient numerically, and so we will derive a closed form expression for the nonsingular PDF, $f_{Z_2, A_2}(z, y)$. Using Bayes rule, we can write f_{Z_2, A_2} as

$$f_{Z_2, A_2}(z, y) = f_{A_2|Z_2}(y|z)f_{Z_2}(z) \quad (5.23)$$

The conditional probability density function $f_{A_2|Z_2}$ can be derived by transformation. If we denote $A_2|Z_2 = u(Z_1)$ and then $Z_1 = u^{-1}(A_2|Z_2) = w(A_2|Z_2)$, we get

$$f_{A_2|Z_2}(y|z) = f_{Z_1}(w(y|z)) \cdot |w'(y|z)| \quad (5.24)$$

Combining (5.23) and (5.24), and rearranging, we get the following initial PDF:

$$f_{Z_2, A_2}(z, y) = f_{Z_1} \left(\log \left(\frac{3y}{S_0} - 1 - e^z \right) \right) \cdot \left| \frac{1}{\frac{3y}{S_0} - 1 - e^z} \frac{3}{S_0} \right| \cdot f_{Z_2}(z) \quad (5.25)$$

Starting with this initial joint PDF, and using Eq. (5.21) recursively for $i = 3, \dots, N$, an approximation is obtained for the time-dependent joint PDF associated with the vector stochastic process (Z_t, A_t) . We can then calculate the option prices from Eq. (5.6) and Eq. (5.7).

Approach 2

An alternative approach, is to define Y_n as

$$Y_n = \sum_{i=0}^n S_i = S_0 \sum_{i=0}^n e^{Z_i} = Y_{n-1} + S_0 e^{Z_n} \quad (5.26)$$

for $n = 1, \dots, N$.

Using the same reasoning as for approach 1, the joint PDF is given by:

$$f_{Z_i, Y_i}(z, y) = \int_{-\infty}^{\infty} f(z - z'; \Delta\tau) f_{Z_{i-1}, Y_{i-1}}(z', y - S_0 e^z) dz' \quad (5.27)$$

and the initial condition from Eq. (5.25) is now

$$f_{Z_2, Y_2}(z, y) = f_{Z_1} \left(\log \left(\frac{y}{S_0} - 1 - e^z \right) \right) \cdot \left| \frac{1}{\frac{y}{S_0} - 1 - e^z} \frac{1}{S_0} \right| \cdot f_{Z_2}(z) \quad (5.28)$$

In the same manner as above, Eq. (5.27) is used recursively for $i = 3, \dots, N$ to obtain $f_{Z_N, Y_N}(z, y)$. However, before calculating the option prices, we need to make a small modification to Eq. (5.6) and (5.7):

$$V_{floating}(T) = e^{-rT} \int_{-\infty}^{\infty} \int_0^{\infty} \rho \cdot \left(S_0 e^z - \frac{1}{N+1} y \right) f_{Z_N, Y_N}(z, y) dy dz \quad (5.29)$$

and

$$V_{fixed}(X, T) = e^{-rT} \int_0^\infty \rho \cdot \left(\frac{1}{N+1}y - X \right) f_{Y_N}(y) dy \quad (5.30)$$

The disadvantage of this approach, is that we need to adjust the grid at each time step τ_i , $i = 3, \dots, N$, since the value of Y_i is growing for each monitoring. Hence, this approach is less practical, especially for larger values of N .

5.3 Monte Carlo Simulation

To verify our answers in the next chapter, we will use Monte Carlo (MC) simulation. The main advantage with MC simulation, is that the implementation is reasonably straightforward, while the drawback is that it is often very time-consuming. However, for the purposes of this thesis, we are interested in verifying our answers, and the time this takes is not of importance.

Monte Carlo simulation uses the Law of Large Numbers. The Law of Large Numbers states that given a random variable with a finite expected value (in our case the price of the stock, and hence the price of the option), if its values are repeatedly sampled, and as the number of these observations increases, their mean will tend to approach and stay close to the expected value. So if we denote the option price for each sample path in the Monte Carlo simulation as X_i , $i = 1, \dots, N_{sim}$, then the sample mean

$$\bar{X}_{N_{sim}} = \frac{1}{N_{sim}} \sum_{i=1}^{N_{sim}} X_i$$

converges to the expected value, μ ,

$$\bar{X}_{N_{sim}} \longrightarrow \mu \quad \text{as} \quad N_{sim} \rightarrow \infty$$

In our case, the expected value μ is the unknown price of the option. Running the MC simulation with a very high number of simulations, we will in theory get this price. But increasing the number of simulations enough to get this answer, is very costly and we can not be sure when we have reached this price. So instead, we will in this thesis set $N_{sim} = 6 \times 10^6$ and then use the Central Limit Theorem to give a 95% confidence interval for μ . The Central Limit Theorem states that

$$\sqrt{N_{sim}} (\bar{X}_{N_{sim}} - \mu) \xrightarrow{D} N(0, \sigma^2)$$

where $\mu = E[X_i]$ and $\sigma^2 = \text{Var}[X_i]$. From this we get the confidence interval

$$P \left(\left| \frac{\bar{X}_{N_{sim}} - \mu}{\sigma / \sqrt{N_{sim}}} \right| \leq z_{\alpha/2} \right) = 1 - \alpha$$

where $z_{\alpha/2}$ is the quantile for the standard normal distribution. In our case, we do not know σ , and so we have to estimate it by using the empirical standard deviation from

all the simulations, s ,

$$\widehat{\text{Var}}[X_i] = s^2 = \frac{1}{N_{\text{sim}} - 1} \sum_{i=1}^{N_{\text{sim}}} (X_i - \bar{X}_{N_{\text{sim}}})^2 \quad (5.31)$$

Since we had to estimate the variance, our statistic is in theory no longer normally distributed, but student t-distributed with $N_{\text{sim}} - 1$ degrees of freedom. However, the student t-distribution will converge to a standard normal distribution as the number of degrees of freedom gets sufficiently large, which is the case here. Using $\alpha = 0.05$, our confidence interval is given as

$$\text{P} \left(\bar{X}_{N_{\text{sim}}} - \frac{s}{\sqrt{N_{\text{sim}}}} \cdot z_{0.025} \leq \mu \leq \bar{X}_{N_{\text{sim}}} + \frac{s}{\sqrt{N_{\text{sim}}}} \cdot z_{0.025} \right) = 0.95$$

where $z_{0.025} = 1.96$.

To simulate the development of the stock price, we need to generate NIG distributed random variables. These NIG random variables are easy to generate, because the NIG process can be viewed as an Inverse Gaussian time-changed Brownian motion (see chapter 2). We can simulate the NIG process (and hence the stock price) by drawing a random Inverse Gaussian number and a random standard normal number. The corresponding random NIG number is given by Eq. (3.5). Drawing random numbers from the normal distribution can be done by using a built-in function (*randn()* in Matlab), but for the Inverse Gaussian distributed random variable, this is not the case. Several people have proposed algorithms for generating an $IG(a, b)$ random number X , among them are (J. Michael & Haas 1976) which proposed the following algorithm:

1. generate a realization of Y , where $Y \sim \chi_1^2$
2. calculate

$$Y_1 = \frac{1}{a} + \frac{b^2}{2a^4} \left(Y + \sqrt{Y + 4\frac{a^3}{b^2}Y} \right)$$

$$Y_2 = \frac{1}{a} + \frac{b^2}{2a^4} \left(Y - \sqrt{Y + 4\frac{a^3}{b^2}Y} \right)$$

3. generate a realization of U , where $U \sim \text{uniform}(0, 1)$
4. set

$$X = \begin{cases} Y_1 & \text{if } U \leq \frac{a}{a+bY_1} \\ Y_2 & \text{if } U > \frac{a}{a+bY_1} \end{cases}$$

Combining this $IG(a, b)$ -generator with Eq. (3.5), we can then generate a $NIG(\alpha, \beta, \mu, \delta)$ random variable Z in the following manner:

1. generate $Y_1 \sim IG(\delta\sqrt{\alpha^2 - \beta^2}, \alpha^2 - \beta^2)$ according to the algorithm above

2. generate Y_2 , where $Y_2 \sim N(0, 1)$
3. then $Z = \mu + \beta Y_1 + \sqrt{Y_1} \cdot Y_2$

When we have this NIG random number generator, we can simulate the development of the stock price forward in time. For the path dependent options we are working on in this thesis, we need to do some modifications to the simulation:

- Barrier option: if the stock price at any monitoring time crosses the barrier level, we set the stock price to zero
- Lookback option: in addition to simulating the stock price, we also keep track of the maximum/minimum of the stock price at all monitoring times
- Asian option: in addition to simulating the stock price, we also sum up the stock prices at each monitoring time, dividing by the number of monitoring times at the end

With these modifications, we can use the payoff functions in Table 5.1 and Table 5.2 to find the simulated price X_i . Repeating this N_{sim} times, we can calculate a 95% confidence interval for the price of the option as described above.

The Monte Carlo technique that we have described here, is the brute-force way of using Monte Carlo simulation. There exist several methods to increase the efficiency of the MC simulation, e.g. variance reduction by control variates, simulation with anti-thetic variables and simulation with quasi-random sampling. For further details on these methods, see e.g. (Jäckel 2002). However, since we in this thesis are only concerned with the MC simulation for validation purposes, we will use the brute-force method.

Chapter 6

Numerical Results

Until now, we have discussed the compatibility of the NIG market model to the market, and also showed how to use numerical path integration together with the NIG market model to price barrier options, lookback options and Asian options. In this chapter we will present the results from our path integration implementation, and as mentioned earlier, we will use Monte Carlo simulation to verify our results. The goal with this thesis was originally to price these exotic options using implied parameters and then compare the results to market prices, as we did for Vanilla options in section 4.4. However, these exotic options are, as mentioned before, traded OTC and we have not succeeded in collecting prices for these exotics from any financial institution. We will therefore in this section compare our results to MC simulation, and leave the discussion on compatibility to market to chapter 4.

Pricing the barrier and lookback options, we will use the fixed parameters from Table 6.1 unless stated otherwise. In some of the tables below, we have used the notation 'PI' for the results using path integration, while 'MC' is the Monte Carlo simulation results.

S_0	100
X_B	100
X_{LB}	105
r	0.05
d	0
α	10
β	-4
δ	1
T	0.2 (50 trading days)

Table 6.1: Fixed input parameters

6.1 Barrier Options

As mentioned earlier, the implementation for pricing barrier options is later used when pricing lookback options, and it is therefore important that $q_N(z)$ from Eq. (5.3) is accurate. As a first check of the correctness of the implementation, we will price some barrier call and put options, and compare with the results from (Aukrust & Naess 2008) and our Monte Carlo results. Table 6.2 show the prices for several different up-and-out-down-and-out call options. The first thing one might notice, is that we achieve the same results as Aukrust & Naess (2008) with one third of the gridsize M . The reason for this, is that we have used piecewise cubic interpolation to reduce the number of calculation of the integral in Eq. (5.10) at each time step. This reduction also show in the computing time which have been reduced with almost 50%. For the purpose of pricing single barrier options, one might argue that this reduction is not very important. But when we later price lookback options, we will perform these calculations many times, and this reduction will then show to be very significant.

L	U	Aukrust		Our results		
		M=600	MC	M=200	MC	95% CI for MC
50	150	6.169	6.168	6.168	6.166	[6.159 , 6.173]
60	140	5.852	5.853	5.852	5.850	[5.844 , 5.857]
70	130	5.054	5.054	5.054	5.053	[5.048 , 5.059]
80	120	3.273	3.273	3.273	3.273	[3.269 , 3.278]
90	110	0.735	0.735	0.735	0.735	[0.733 , 0.736]
Computing time:		2.2 s	> 5000 s	1.15 s	< 220 s	

Table 6.2: Down-and-out-up-and-out call prices for different barrier levels. The Monte Carlo calculations are calculated with 6×10^6 simulations, and the PI calculations are made $M = 200$ gridpoints.

The prices in Table 6.2 seem to coincide very well with the results found by Aukrust & Naess (2008), and also with our Monte Carlo results. We see that all prices are inside the 95% confidence interval for the MC calculations, which is a strong indication that our implementation of the barrier options is accurate. Further, in Table 6.3 we have calculated the prices for several up-and-out call option prices, with varying barrier values. Here we have used a smaller gridsize, $M = 350$, because the stock price range is wider. For the up-and-out-in-and-out option, the range was at most 100 (50 to 150), while here the maximum stock price range is 155. Hence, the computing time for our path integration implementation also depend on the stock price range we need in order to get the desired accuracy. Looking at the prices in Table 6.3, we see that our prices again coincides very well with both the results from (Aukrust & Naess 2008) and the MC results, and all prices stay inside the 95% confidence interval.

Barrier	Aukrust		Our results		
	PI	MC	PI	MC	95% CI for MC
105	0.114	0.114	0.114	0.114	[0.114 , 0.115]
115	2.000	2.001	2.000	1.998	[1.995 , 2.001]
125	4.317	4.319	4.317	4.315	[4.310 , 4.320]
135	5.541	5.541	5.541	5.539	[5.533 , 5.545]
145	6.046	6.042	6.047	6.049	[6.043 , 6.056]
155	6.245	6.246	6.245	6.245	[6.238 , 6.252]
Computing time:	-	> 5000 s	2.9 s	< 220 s	

Table 6.3: Up-and-out call prices for different barrier levels. The Monte Carlo calculations are calculated with 6×10^6 simulations, and my PI calculations are made with $M = 350$ gridpoints.

The results presented in Table 6.2 and Table 6.3 seem to be very accurate and fast, which is a good start when we now move on to price lookback options.

6.2 Lookback Options

As we showed in section 5.2.1, the underlying probability density function for $\tilde{Z}_T = \log(\frac{\tilde{S}_T}{S_0})$, $q_N(z)$, make up the foundation when pricing both fixed and floating strike lookback options. The accuracy we have seen above, is a good indication that our implementation of $q_N(z)$ is accurate, and so we will now use this to price lookback options, starting with the fixed strike case. Before we start, it is important to recognize that there are now two main numerical aspects which influences the speed and accuracy; the stock price range gridsize M as before and also the size of the interval between each evaluation, $d\eta$. In addition the range $\eta_{max} - \eta_{min}$ will influence the speed, and should be chosen wisely. If we make the range to narrow, we might not capture all the probability. At the same time one should not choose a too wide range, because we might end up doing many relatively time-consuming calculations where the probability density is zero. So if we need a small $d\eta$ to get an accurate answer, the 50% reduction in computing time from above may be very significant. These aspects we will discuss further below.

6.2.1 Fixed Strike Lookback Options

When we price fixed strike lookback call and put options, we follow the reasoning from section 5.2.1, and use the fixed parameters from Table 6.1. As we mentioned above, the choice of M and $d\eta$ influences the results with regards to both speed and accuracy, and we have therefore in Table 6.4 experimented with different values of M and $d\eta$ to see how the accuracy and speed vary.

	M = 200		M = 300		M = 400		M = 500	
	Price	Time	Price	Time	Price	Time	Price	Time
$d\eta = 10$	6.195	22s	6.137	48s	6.141	96s	6.141	147s
$d\eta = 5$	6.106	31s	6.102	74s	6.102	140s	6.102	227s
$d\eta = 3$	6.075	41s	6.099	106s	6.099	205s	6.099	332s
$d\eta = 2$	6.077	56s	6.099	144s	6.098	280s	6.098	460s
$d\eta = 1$	6.072	94s	6.099	265s	6.098	521s	6.098	855s

Table 6.4: Price of a fixed strike lookback call option with strike price $X_{LB} = 105$, with varying gridsize M and $d\eta$. The price for this call option according to the Monte Carlo simulation is 6.098 (with the corresponding 95% confidence interval [6.091, 6.105]). The computing time for the MC calculations is approximately 270 seconds.

In section 5.2.1, we briefly discussed the critical part around S_0 , and for all the results below we have used $d\eta = 1$ in the range $[S_0 - 5, S_0 + 5]$. For the particular case in Table 6.4, where we have $X_{LB} = 105$ and a call option, this modification has little impact since the payoff function for the option is zero for $\eta \leq 105$. However, for lower values of X_{LB} or a put option with $X_{LB} \geq 100$, this modification will have a significant impact. This smaller value of $d\eta$ around S_0 is also the reason that the computing time is not linear with respect to $d\eta$, as we normally might expect.

From the results in Table 6.4, we see that the prices using $d\eta = 3$ gives us a very satisfying result, at least for $M \geq 300$. We have used this $d\eta$ -value when pricing fixed strike lookback call options with different strike prices in Table 6.5. For our validation purposes in this thesis, we will use small gridsizes since we are interested in verifying the accuracy of our implementation. However, in practice it does not make much sense to calculate lookback option prices with the accuracy of three decimals, since the model error almost certainly will be much higher, and also because a typical bid-ask spread can be as high as several percentage points.

Recall from Table 6.4 that $d\eta = 3$ gave satisfying results, and in Table 6.5 we have calculated the prices for fixed call options with different strike prices. For the purpose of verifying the convergence behavior with respect to M , we have done this for three values of M . As we can see from Table 6.5, there is almost nothing to gain in accuracy by increasing M above 300, considering the corresponding increase in computing time. The results seem very accurate compared to the MC results, and we see that all prices are inside the 95% confidence interval for the MC simulations. The computing time is also considerably smaller than for the MC simulations. As we discussed above, for practical purposes this high accuracy might not be necessary, which again would lead to an even smaller computing time.

In Table 6.6 we have calculated the prices for a fixed strike put, for varying gridsize values M and $d\eta$. The computing times in Table 6.6 differs a bit from the corresponding call prices in Table 6.4, and the reason for this is that the range $[\eta_{min}, \eta_{max}]$ is different.

Strike (X)	M = 300	M = 400	M = 500	MC	95% CI for MC
65	44.513	44.513	44.513	44.516	[44.508 , 44.524]
70	39.564	39.564	39.564	39.565	[39.557 , 39.573]
75	34.615	34.615	34.615	34.622	[34.615 , 34.630]
80	29.667	29.667	29.667	29.662	[29.654 , 29.669]
85	24.721	24.721	24.721	24.724	[24.717 , 24.732]
90	19.780	19.780	19.780	19.782	[19.774 , 19.790]
95	14.850	14.850	14.850	14.848	[14.841 , 14.856]
100	9.977	9.977	9.977	9.979	[9.972 , 9.987]
105	6.099	6.099	6.099	6.098	[6.091 , 6.105]
110	3.549	3.549	3.550	3.549	[3.544 , 3.555]
115	1.992	1.992	1.993	1.996	[1.992 , 2.001]
120	1.100	1.100	1.100	1.100	[1.097 , 1.104]
125	0.606	0.606	0.607	0.606	[0.603 , 0.608]
130	0.337	0.337	0.338	0.339	[0.337 , 0.341]
135	0.190	0.191	0.191	0.191	[0.189 , 0.192]
140	0.109	0.109	0.110	0.111	[0.110 , 0.112]
Computing time	106s	205s	332s	<270 s	

Table 6.5: Fixed strike lookback call prices for different strike prices. The Monte Carlo calculations are calculated with 6×10^6 simulations, and the PI calculations are made with varying number of gridpoints M and $d\eta = 3$.

	M = 200		M = 300		M = 400		M = 500	
	Price	Time	Price	Time	Price	Time	Price	Time
$d\eta = 10$	13.573	19s	13.289	41s	13.272	71s	13.281	110s
$d\eta = 5$	13.414	31s	13.267	65s	13.249	113s	13.257	176s
$d\eta = 3$	13.451	43s	13.248	92s	13.249	160s	13.254	248s
$d\eta = 2$	13.434	61s	13.249	130s	13.248	227s	13.254	349s
$d\eta = 1$	13.420	111s	13.250	239s	13.247	411s	13.254	639s

Table 6.6: Price of a fixed strike lookback put option with strike price $X_{LB} = 105$, with varying gridsize M and $d\eta$. The price for this put option according to the Monte Carlo simulation is 13.252 (with the corresponding 95% confidence interval [13.244 , 13.260]). The computing time for the MC calculations is approximately 275 seconds.

While we for the call prices had a consistency in the price already at $M = 300$ and $d\eta = 3$, we here see some small variations for smaller gridsizes. But these variations are relatively small, especially compared to factors like model error and bid-ask spreads. As

for the call price case above, we are satisfied with the accuracy given by $d\eta = 3$, and will use this $d\eta$ -value when pricing fixed put options for different strike prices in Table 6.7.

Strike (X)	M = 300	M = 400	M = 500	MC	95% CI for MC
65	0.160	0.160	0.161	0.163	[0.162 , 0.164]
70	0.310	0.308	0.310	0.311	[0.310 , 0.313]
75	0.573	0.573	0.575	0.576	[0.573 , 0.578]
80	1.032	1.031	1.034	1.034	[1.030 , 1.037]
85	1.805	1.804	1.808	1.805	[1.801 , 1.809]
90	3.078	3.078	3.082	3.085	[3.080 , 3.090]
95	5.124	5.124	5.129	5.125	[5.119 , 5.132]
100	8.409	8.410	8.415	8.410	[8.402 , 8.417]
105	13.248	13.249	13.254	13.252	[13.244 , 13.260]
110	18.189	18.190	18.195	18.193	[18.185 , 18.200]
115	23.137	23.138	23.144	23.147	[23.139 , 23.154]
120	28.087	28.088	28.094	28.091	[28.084 , 28.099]
125	33.038	33.039	33.044	33.048	[33.040 , 33.055]
130	37.988	37.989	37.994	37.996	[37.988 , 38.003]
135	42.938	42.939	42.944	42.948	[42.940 , 42.955]
140	47.888	47.889	47.895	47.894	[47.886 , 47.901]
Computing time	92s	160s	248s	< 275 s	

Table 6.7: Fixed strike lookback put prices for different strike prices. The Monte Carlo calculations are calculated with 6×10^6 simulations, and the PI calculations are made with varying number of gridpoints M and $d\eta = 3$.

In Table 6.7 we see that the put prices seem to be relatively accurate. If we compare the prices for the different M -values, we see that the prices with $M = 500$ are a bit higher than for $M = 300$ and $M = 400$. From Table 6.6 we saw the same thing for $d\eta = 3$ to $d\eta = 5$. The reason for this, we have no good explanation for, other than a small numerical inaccuracy. Looking at the prices with $M = 300$ and $M = 400$ which are relatively alike, we see that they mostly lie around the lower limit of the 95% confidence interval. Some fall outside, while some are inside the interval. Again it is worth mentioning that we are dealing with a very narrow confidence interval, and for practical use these results are more than enough regarding accuracy. And as we can see, they are much faster than the MC simulation results. Considering the results with $M = 500$, we see that all prices, except for $X = 65$, end up inside the confidence interval which again shows that our numerical path integration implementation is very accurate. And even for $M = 500$ the path integration approach is faster than the Monte Carlo simulation.

6.2.2 Floating Strike Lookback Options

When we now price floating strike lookback call and put options, we follow the reasoning in section 5.2.1, and use the fixed parameters from Table 6.1. As explained before, we now have to keep track of the probability density for the stock price at maturity T , S_T , in addition to the maximum/minimum. In order to reduce the computing time considerably, it is necessary to use cubic interpolation, both one and two dimensional. This interpolation may not preserve the property from Eq. (5.14) in a satisfactory manner, and it is therefore important that we adjust our implementation accordingly.

	M = 200		M = 300		M = 400		M = 500	
	Price	Time	Price	Time	Price	Time	Price	Time
$d\eta = 10$	10.351	33s	9.036	64s	9.054	102s	8.981	153s
$d\eta = 5$	10.473	45s	9.182	95s	9.290	151s	9.285	231s
$d\eta = 3$	10.382	60s	9.180	127s	9.291	208s	9.283	320s
$d\eta = 2$	10.558	80s	9.184	166s	9.291	286s	9.282	435s
$d\eta = 1$	10.552	148s	9.182	301s	9.291	511s	9.281	811s

Table 6.8: Price of a floating strike lookback call option with varying gridsize M and $d\eta$. The price for this call option according to the Monte Carlo simulation is 9.282 (with the corresponding 95% confidence interval [9.275 , 9.289]). The computing time for the MC calculations is approximately 215 seconds.

Table 6.8 show the results from calculating the price of a floating strike lookback call option for different gridsizes M and $d\eta$. From the table, we first notice that the computing times are not significantly different from the corresponding calculations with fixed strikes. As for the fixed strike put prices in Table 6.6, we see some small variations in the prices for smaller gridsizes than $M = 300$ and $d\eta = 3$. The fact that the fixed strike put prices and the floating strike call prices are similar in this manner, is not surprising, since they both use the minimum in their payoff function. We see that for $M \geq 400$ and $d\eta \leq 5$ we get relatively satisfying results, but we must have $M = 500$ in order to get inside the 95% confidence interval.

Table 6.9 show the price of a floating strike lookback put option for different gridsizes M and $d\eta$. The results show much of the same consistency as we had for the fixed call options in Table 6.4, which are also based on the maximum of the stock. The results seem to converge quite well, and we see that for $M \geq 400$ and $d\eta \leq 3$ our resulting prices end up inside the 95% confidence interval.

Considering the computational speed, we see that in order to get a theoretically respectable accuracy, we do not necessarily end up with the answer much faster than using Monte Carlo simulations for these two cases. However, if we take a closer look into the implementation theory in section 5.2.1, we may notice that for each value of η , we run calculations that are independent of each other. Hence, we can in practice run the

	M = 200		M = 300		M = 400		M = 500	
	Price	Time	Price	Time	Price	Time	Price	Time
$d\eta = 10$	15.482	28s	17.986	57s	19.528	88s	20.3357	132s
$d\eta = 5$	9.170	40s	8.948	83s	8.975	127s	8.991	196s
$d\eta = 3$	9.141	52s	8.867	109s	8.860	171s	8.858	265s
$d\eta = 2$	9.372	72s	8.878	144s	8.862	232s	8.862	358s
$d\eta = 1$	9.783	127s	8.885	254s	8.866	415s	8.863	644s

Table 6.9: Price of a floating strike lookback put option with varying gridsize M and $d\eta$. The price for this put option according to the Monte Carlo simulation is 8.863 (with the corresponding 95% confidence interval [8.855 , 8.871]).

calculations for each value of η simultaneously if we take parallel computing into use. This would lead to a drastic decrease with respect to computing time.

We conclude this section by considering call and put prices on floating strike lookback options for different values of S_0 . In addition we also test our implementation for options with weekly monitoring, in addition to the daily monitoring options we have discussed so far. Changing the value of $\Delta\tau$ should, unlike many other numerical problems, not cause any change in the accuracy of our results. This because the only element depending on $\Delta\tau$ in Eq. (5.10), is $f(z - z'; \Delta\tau)$, which is explicitly given by Eq. (3.11) for all values of $\Delta\tau$. From Table 6.10 and Table 6.11 we see that changing from daily to weekly monitoring changes the prices, but not the accuracy.

		PI	MC	95% CI for MC
$dt = \frac{1}{50}$	$S_0 = 60$	4.928	4.929	[4.925 , 4.933]
	$S_0 = 80$	6.564	6.566	[6.561 , 6.572]
	$S_0 = 100$	8.208	8.211	[8.205 , 8.218]
	$S_0 = 120$	9.852	9.852	[9.844 , 9.860]
	$S_0 = 140$	11.490	11.491	[11.481 , 11.501]
$dt = \frac{1}{250}$	$S_0 = 60$	5.586	5.577	[5.573 , 5.581]
	$S_0 = 80$	7.438	7.436	[7.430 , 7.442]
	$S_0 = 100$	9.281	9.282	[9.275 , 9.289]
	$S_0 = 120$	11.144	11.148	[11.139 , 11.157]
	$S_0 = 140$	13.002	12.996	[12.986 , 13.006]

Table 6.10: Floating lookback call option prices with varying dt and S_0 . For the PI calculation, $M = 500$ and $d\eta = 1$ is used.

First considering Table 6.10, we see that the results are quite good. All the prices are inside the 95% confidence interval for the Monte Carlo simulations, except for the

case with daily monitoring and $S_0 = 60$. Here the price for some reason is less than 0.1% below the lower limit. In Table 6.11 we see that all put prices end up inside the 95% confidence interval. In general these results confirm that the our path integration implementation works very well for different values of dt and S_0 , which is reassuring.

		PI	MC	95% CI for MC
$dt = \frac{1}{50}$	$S_0 = 60$	4.611	4.612	[4.607 , 4.616]
	$S_0 = 80$	6.151	6.148	[6.142 , 6.154]
	$S_0 = 100$	7.685	7.687	[7.679 , 7.695]
	$S_0 = 120$	9.225	9.226	[9.217 , 9.235]
	$S_0 = 140$	10.756	10.762	[10.752 , 10.773]
$dt = \frac{1}{250}$	$S_0 = 60$	5.323	5.321	[5.317 , 5.326]
	$S_0 = 80$	7.096	7.090	[7.083 , 7.096]
	$S_0 = 100$	8.863	8.863	[8.855 , 8.871]
	$S_0 = 120$	10.627	10.630	[10.621 , 10.640]
	$S_0 = 140$	12.401	12.410	[12.399 , 12.421]

Table 6.11: Floating lookback put option prices with varying dt and S_0 . For the PI calculation, $M = 500$ and $d\eta = 1$ is used.

When calculating these results, we have used $d\eta = 1$ and $M = 500$, since we have seen from the tables above that these gridsizes consistently gives us accurate results. However, making the calculations with such a narrow grid is time-consuming. And as we have discussed before, we will get adequate results with relatively much wider grid, and hence save significantly with respect to computing time.

6.3 Asian Options

We have implemented the two different approaches described in section 5.2.2. When calculating the results, we have used the two fixed parameter sets in Table 6.12 unless stated otherwise. The results below, are made mainly with monthly monitoring because the computing time and the need of a very accurate grid is otherwise too large. The parameter values in set 1 are chosen as the same that Fusai & Meucci (2008) used when pricing their Asian options. Although they only presented prices for one call option, we still get a second verification that our Monte Carlo simulation is correct.

Below we will use the following notation for the different approaches:

- PI1: path integration using approach 1 from section 5.2.2
- PI2: path integration using approach 2 from section 5.2.2

We start off by using the results obtained by Fusai & Meucci (2008) to verify that our Monte Carlo simulations are correct. Although the standard errors presented by Fusai

	Set 1	Set 2
S_0	100	100
X	90	90
r	0.0367	0.05
α	6.1882	10
β	-3.8941	-4
δ	0.1622	1
T	1	1
dt	$\frac{1}{12}$	$\frac{1}{12}$
d	0	0

Table 6.12: Fixed input parameters for Asian options

& Meucci (2008) are relatively high, they still have a compliance between their results using Monte Carlo with control variates and their results using numerical quadrature. From Table 6.13 we see that our Monte Carlo results coincides very well with the results from (Fusai & Meucci 2008), and this gives us a good indication that our Monte Carlo simulation results are reasonably correct.

		MC_{FM}	SE_{FM}	$Quad_{FM}$	MC	95% CI for MC
$dt = \frac{1}{12}$	$X = 90$	12.623	0.341	12.622	12.624	[12.617 , 12.631]
	$X = 100$	5.061	0.223	5.061	5.058	[5.053 , 5.063]
	$X = 110$	1.014	0.176	1.014	1.013	[1.010 , 1.015]
$dt = \frac{1}{50}$	$X = 90$	12.661	0.333	12.661	12.663	[12.657 , 12.670]
	$X = 100$	5.104	0.221	5.104	5.106	[5.101 , 5.111]
	$X = 110$	1.038	0.177	1.038	1.036	[1.033 , 1.038]
$dt = \frac{1}{250}$	$X = 90$	12.672	0.329	NA	12.671	[12.664 , 12.678]
	$X = 100$	5.116	0.218	NA	5.117	[5.112 , 5.122]
	$X = 110$	1.044	0.176	NA	1.044	[1.041 , 1.046]

Table 6.13: Prices for fixed strike Asian call option for different values of dt and X . Here MC_{FM} denotes the Monte Carlo results for (Fusai & Meucci 2008) with SE_{FM} denoting the corresponding standard error. $Quad_{FM}$ denotes the results Fusai & Meucci (2008) obtained with numerical quadratures. MC denotes our Monte Carlo results.

Now that we have made a quick and rough verification of our Monte Carlo simulation, we will calculate the prices of fixed and floating strike Asian put and call options. Table 6.14 show the results for different gridsize values M , using parameter set 1 and path integration approach 1 (PI1).

		M = 100	M = 200	M = 400	MC	95% CI for MC
Fixed	Call	12.315	12.548	12.579	12.624	[12.617 , 12.631]
	Put	1.306	1.238	1.149	1.191	[1.187 , 1.194]
Floating	Call	5.235	5.177	5.190	5.170	[5.165 , 5.175]
	Put	3.344	3.387	3.378	3.359	[3.354 , 3.365]
Computing time:		43s	363s	3037s	50s	
Maximum error		9.66%	3.95%	3.53%		
Average error		3.45%	1.38%	1.21%		

Table 6.14: Asian option prices with PI1 for Set 1, with varying number of gridpoints M

From Table 6.14 we see that maximum and average percentage error is decreasing with decreasing gridsize, which naturally is a good sign. However, if we look at the prices, we see that they are not necessarily converging to the same price as for the Monte Carlo simulation. The maximum error is for all three M -values given by the fixed strike put option price, which is reasonable since these prices have the lowest values. If we calculate the absolute deviation, we see that the fixed strike call option price deviates the most, but closely followed by the fixed put price. We also see that the Monte Carlo simulation takes about 50 seconds to calculate, while the the path integration implementation is much more time-consuming in order to achieve convergence. Although it is not included in Table 6.14 to Table 6.17, we have also calculated the prices with $M = 700$, to verify that we actually have achieved convergence at $M = 400$.

		M = 100	M = 200	M = 400	MC	95% CI for MC
Fixed	Call	12.523	12.572	12.579	12.624	[12.617 , 12.631]
	Put	1.220	1.150	1.147	1.191	[1.187 , 1.194]
Floating	Call	5.272	5.189	5.190	5.170	[5.165 , 5.175]
	Put	3.360	3.379	3.379	3.359	[3.354 , 3.365]
Computing time:		60s	497s	3730s	50s	
Maximum error		2.43%	3.44%	3.69%		
Average error		1.31%	1.20%	1.26%		

Table 6.15: Asian option prices with PI2 for Set 1, with varying number of gridpoints M

Table 6.15 show the results of the same calculations as Table 6.14, but here approach 2 (PI2) have been used to calculate the prices. Comparing the results with Table 6.14, we see that these results are converging to the same prices. However, using PI2 we see that for $M = 100$, the prices are much closer to the converged prices.

In Table 6.16 and Table 6.17 we have done the same calculations as above, but using parameter set 2. Here we see the same deviation from the Monte Carlo results, but we

		M = 100	M = 200	M = 400	MC	95% CI for MC
Fixed	Call	14.562	14.577	14.578	14.750	[14.737 , 14.762]
	Put	2.750	2.718	2.717	2.817	[2.812 , 2.823]
Floating	Call	8.875	8.882	8.882	8.946	[8.935 , 8.958]
	Put	6.517	6.513	6.513	6.484	[6.477 , 6.492]
Computing time:		27s	282s	2259s	50s	
Maximum error		2.38%	3.51%	3.55%		
Average error		1.24%	1.46%	1.47%		

Table 6.16: Asian option prices with PI1 for Set 2, with varying number of gridpoints M

also observe a much faster convergence than we had for parameter set 1.

		M = 100	M = 200	M = 400	MC	95% CI for MC
Fixed	Call	14.577	14.578	14.578	14.750	[14.737 , 14.762]
	Put	2.717	2.717	2.717	2.817	[2.812 , 2.823]
Floating	Call	8.882	8.882	8.882	8.946	[8.935 , 8.958]
	Put	6.513	6.513	6.513	6.484	[6.477 , 6.492]
Computing time:		41s	361s	2670s	50s	
Maximum error		3.55%	3.55%	3.55%		
Average error		1.47%	1.47%	1.47%		

Table 6.17: Asian option prices with PI2 for Set 2, with varying number of gridpoints M

We have now seen that for monthly monitoring, our results using both PI1 and PI2 for parameter set 1 and 2 are not as accurate as we had expected. In section 6.3.1 we discuss possible elements that might have caused this inaccuracy in the results. We have in Table 6.18 and Table 6.19 calculated the option prices for both parameter sets with weekly monitoring. The results are found using only PI1, since the process of finding the dynamic grid combined with a high demand for an accurate grid, makes PI2 much less practical for smaller values of dt .

From Table 6.18 and Table 6.19 we see that the prices are actually not as bad one might have expected. For parameter set 1, we actually have three out of four prices inside the 95% confidence interval for the MC results. For parameter set 1 we also notice that by increasing M from 500 to 700, we get a considerable change in the prices. For parameter set 2, the prices have converged already at $M = 300$. This shows that the convergence depends on the parameter set we use. The surprisingly large change in the prices for set 1 when increasing M from 500 to 700, may be an indication that perhaps the prices with monthly monitoring has not converged completely. However, computing the monthly monitoring prices with $M = 700$ gives the same prices as for $M = 400$ for

both methods and both parameter sets.

		M = 300	M = 500	M = 700	MC	95% CI for MC
Fixed	Call	0.003	12.622	12.653	12.657	[12.650 , 12.663]
	Put	73.738	1.380	1.237	1.234	[1.230 , 1.238]
Floating	Call	0.038	5.209	5.219	5.215	[5.210 , 5.221]
	Put	6.056	3.405	3.398	3.410	[3.404 , 3.416]
Computing time:		12610s	23980s	74050s	200s	

Table 6.18: Asian option prices with PI1 for Set 1, with varying number of gridpoints M and with $dt = \frac{1}{50}$

		M = 300	M = 500	M = 700	MC	95% CI for MC
Fixed	Call	14.759	14.759	14.759	14.844	[14.831 , 14.857]
	Put	2.909	2.909	2.909	2.914	[2.908 , 2.919]
Floating	Call	8.962	8.962	8.962	9.054	[9.042 , 9.065]
	Put	6.601	6.601	6.601	6.594	[6.586 , 6.602]
Computing time:		9600	23025s	98430s	200s	

Table 6.19: Asian option prices with PI1 for Set 2, with varying number of gridpoints M and with $dt = \frac{1}{50}$

Seeing the somewhat inaccurate results for monthly monitoring above, it is quite surprising that the results for weekly monitoring in Table 6.18 and Table 6.19 are that accurate. The only element in Eq. (5.21) depending on the number of monitoring times N , is $f(z - z'; \Delta\tau)$, and this element is explicitly given for all values of $\Delta\tau$ by Eq. (3.11).

6.3.1 Error Analysis

As we can see from the results above, our prices for the monthly monitored Asian options are not as accurate as the results we had for the lookback options. Why is this so? The first reason we came up with, was naturally that there is some numerical inaccuracy in the implementation we have performed. Looking at Eq. (5.21) in section 5.2.2, we see that the only element that may cause such a numerical inaccuracy, is the interpolation we use when finding $f_{Z_{i-1}, A_{i-1}}$. Here we have used the built-in Matlab function `interp2()` with bicubic interpolation. If this interpolation is the reason for our inaccurate results, one should expect that the deviation in the option price will decrease as we increase the number of gridpoints M . This because the increased number of gridpoints that we make the interpolation based on, will give the interpolation more constraints.

Other factors that may have caused this numerical inaccuracy, is the choice of Z_{min} , Z_{max} , A_{min} and A_{max} . But as seen from Figure 6.1, the joint probability density function is far from nonzero at any of the endpoints. We also, for each time step, integrate up the PDF to make sure that the total probability of the entire grid is 1. For all our calculations this was the case, with only an absolute deviation in the magnitude of less than 0.005.

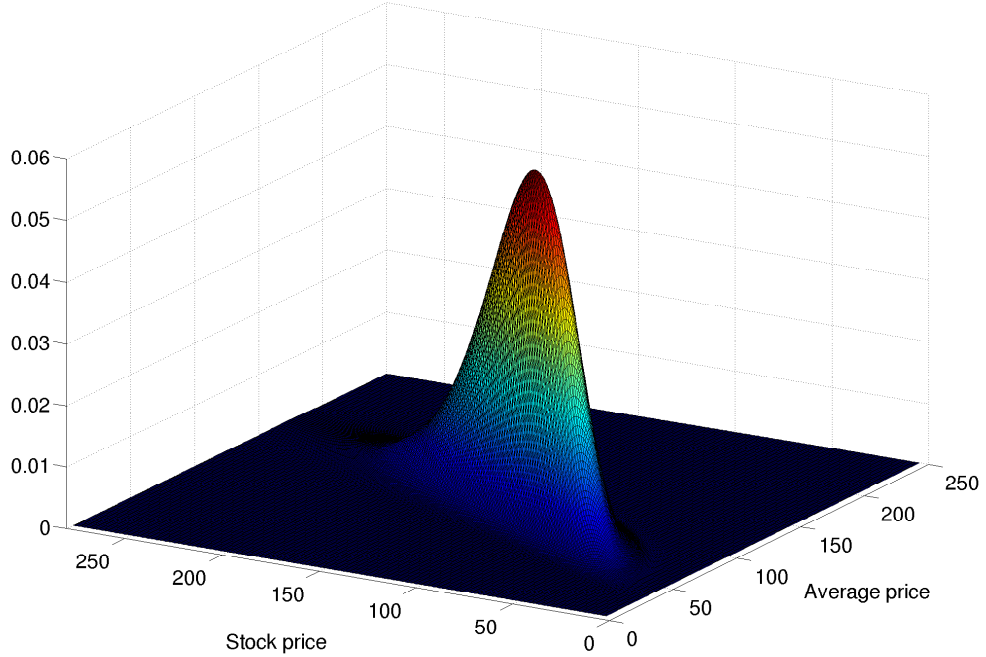


Figure 6.1: Plot showing the joint probability density function f_{Z_N, A_N} for parameter set 2 using PI1.

In the search for this numerical inaccuracy, we did not manage to improve the results by changing the grid values and the gridsize. Therefore we looked at the initial PDF, which also integrated up to 1, to see if this initial joint PDF were the source to our inaccurate results. First we tried to price an option using only two monitoring points, that is $N = 2$. These prices also deviated from the corresponding results found by Monte Carlo simulation. This seems strange, since the calculations leading to Eq. (5.25) and Eq. (5.28) are relatively straightforward. However, as a second attempt for a initial joint PDF, we tried to use the same procedure as Skaug & Naess (2005). $f_{Z_1, A_1}(z, y)$ is originally on the singular form

$$f_{Z_1, A_1}(z, y) = \delta\left(y - \frac{S_0}{2}(1 + e^z)\right) \cdot f(z; \Delta\tau)$$

In this approximation, we introduce $\tilde{Y}_1 = \frac{S_0}{2}(1 + e^{Z_1}) + E$ where $E \sim N(0, e^2)$ and e is sufficiently small. Then the nonsingular approximation of $f_{Z_1, A_1}(z, y)$ is given by

$$f_{Z_1, A_1}(z, y) = \Phi_N \left(y; \frac{S_0}{2}(1 + e^z), e^2 \right) \cdot f(z; \Delta\tau)$$

Using this initial PDF, and calculating Eq. (5.21) and Eq. (5.27) recursively for $i = 2, \dots, N$, we end up with results relatively identical to the results above. Hence, it seems as the initial joint PDF is not causing the inaccuracy.

As a final attempt to find the cause of the inaccuracy, we tried to implement a dynamic grid. The initial PDF is only nonzero at a very narrow range on the grid, especially compared to the PDF at $t = T$, and so it may give us a numerical inaccuracy to have a static grid large enough to include the entire final PDF. However, implementing such a dynamic grid, did not improve our results significantly.

The cause of the inaccurate prices for the Asian options presented in this section, remains a mystery. The theory in section 5.2.2 is relatively straight-forward, and should give us relatively exact results. As explained above, we have tried to go through the different aspects of this problem to find out where the inaccuracy occurs, but we have not succeeded.

Conclusion

In chapter 4 we have shown that the Normal Inverse Gaussian market model looks to have a good fit to historical market data, except perhaps for StatoilHydro. However, for practical use it may not be adequate to find the parameter values using historical data, since the market and the financial assets are driven by the investors expectations to the future development of the financial market. Therefore, we also calculated the implied parameters, where we use current option prices in the calibration. For some of the data sets, the calibration results are depending on which error measure we chose, which is an indication that the calibration risk is significant for some data sets. We proposed four different error measures, and which one to use, will depend on the problem at hand. Compared to the more commonly used Black-Scholes model, we have seen that the NIG market model fits market data better, both with historical and implied parameters.

In chapter 5 and 6 we focused on pricing different varieties of barrier, lookback and Asian options for fixed parameter sets. For the barrier options and both the fixed and floating strike lookback options the results were very accurate. The computing time was also smaller than for the Monte Carlo simulations. For the Asian option calculations, we experienced some inaccuracy compared to the Monte Carlo prices. However, from a practical point of view, the magnitude of these deviations are not deterrent, considering other uncertainty aspects of financial modeling, such as model and calibration risk etc. But still, there is an unknown source of inaccuracy in the implementation of the Asian options that we have not succeeded in uncovering.

When pricing financial derivatives with path integration, it is crucial that the numerical scheme used is accurate. In our implementation, the computational speed has not been our main focus. Especially, one can make huge improvements in the computing time when calculating the cumulative distribution function needed to price lookback options. These computations are independent of each other, and by running these calculations in parallel on a network of computers, one can reduce the computational speed significantly. Noticing that the transition probabilities in general are quite narrow, there is also a considerable gain to be made by concentrating the computation around the center of the mean. It is also worth noticing that we in this thesis have focused on the accuracy and convergence properties of our implementation and in practice, it may not be necessary to calculate the prices with this degree of accuracy, since the model and calibration risk may be quite significant compared to any numerical inaccuracy.

Bibliography

- Abramowitz, M. & Stegun, I. (1968). *Handbook of Mathematical Functions*, New York: Dover.
- Aukrust, E. G. & Naess, A. (2008). Pricing of barrier options with NIG dynamics using numerical path integration, *Submitted* .
- Barndorff-Nielsen, O. E. (1994). Normal Inverse Gaussian distribution and the modeling of stock returns, *Research Report no. 300, Department of Theoretical Statistics, Aarhus University* .
- Bjornland, H. C. (2008). Oil Price Shocks and Stock Market Booms in an Oil Exporting Country, *working paper from Norges Bank* **16**.
- Black, F. & Scholes, M. (1973). The pricing of options and corporate liabilities, *Journal of Political Economy* pp. 637–659.
- Detlefsen, K. & Härdle, W. K. (2006). Calibration Risk for Exotic Options, *SFB 649 Discussion Paper* .
- Fusai, G. & Meucci, A. (2008). Pricing discretely monitored Asian options under Lévy processes, *Journal of Banking and Finance* **32**: 2076–2088.
- J. Michael, W. S. & Haas, R. (1976). Generating random variates using transformations with multiple roots, *American Statistician* pp. 88–90.
- Jäckel, P. (2002). Monte Carlo Methods in Finance, *John Wiley & Sons* .
- Kalemanova, A. & Werner, R. (2006). A short note on the efficient implementation of the Normal Inverse Gaussian distribution.
- Madan, D. B., Carr, P. & Chang, E. C. (1998). The variance gamma process and option pricing, *European Finance Review* **2**: 79–105.
- Massey, F. J. (1951). The Kolmogorov-Smirnov Test for Goodness of Fit, *Journal of the American Statistical Association* **46**: 68–78.
- Myung, I. J. (2003). Tutorial on maximum likelihood estimation, *Journal on Mathematical Psychology* **47**: 90–100.

- Naess, A. & Moe, V. (2000). Efficient path integration methods for nonlinear dynamic systems, *Probabilistic Engineering Mechanics* **15**: 221–231.
- Nocedal, J. & Wright, S. J. (2000). *Numerical Optimization*, Springer.
- Rebonato, R. (2004). *Volatility and Correlation: The Perfect Hedger and the Fox*, John Wiley Sons; 2nd Edition.
- Schoutens, W. (2003). *Levy Processes in Finance*, Wiley series in probability and statistics.
- Skaug, C. & Naess, A. (2005). Pricing of Asian Options by Numerical Path Intergration, *Internal Report, Institute of Mathematical Sciences, NTNU* .
- Skaug, C. & Naess, A. (2007). Fast and accurate pricing of discretely monitored barrier options by numerical path intergration, *Computational Economics* **30**: 143–151.
- Wilmott, P., Howison, S. & Dewynne, J. (1995). *The Mathematics of Financial Derivatives*, The Press Syndicate of the University of Cambridge.

Appendix A

Source Code

All implementation have been programmed using Matlab. The code included below is the code used to produce all results in chapter 6, with some small modifications. Especially, we need to choose the range $[\eta_{min}, \eta_{max}]$ according to what values of S_0 we are working with. In order to reduce the number of inputs in our function calls, we have chosen to do these modifications inside the code. The Matlab functions presented below are:

- *NIGBarrierPI()*: - calculates the price of a barrier option or returns the PDF $q_N(z)$ under the NIG market model using numerical path integration
- *NIGBarrierMC()*: - calculates the price of a barrier option under the NIG market model using Monte Carlo simulation
- *NIGLookbackPIFixed()*: - calculates the price of a fixed strike lookback option under the NIG market model using numerical path integration
- *NIGLookbackPIFloating()*: - calculates the price of a floating strike lookback option under the NIG market model using numerical path integration
- *NIGLookbackMC()*: - calculates the price of both fixed strike and floating strike lookback options under the NIG market model using Monte Carlo simulation
- *NIGAsianPI1()*: - calculates the price of both fixed strike and floating strike Asian options under the NIG market model using path integration approach 1
- *NIGAsianPI2()*: - calculates the price of both fixed strike and floating strike Asian options under the NIG market model using path integration approach 2
- *NIGAsianMC()*: - calculates the price of both fixed strike and floating strike Asian options under the NIG market model using Monte Carlo simulation

```
function price = NIGBarrierPI(S0,X, r, d, T, L, U,alpha, beta, delta, M2, ...  
    priceFlag, CPflag)  
tic  
%input:
```

```

%S0      - initial stock price
%X       - strike price for the barrier option
%r       - interest rate
%d       - dividend yield
%T       - time to maturity (in years)
%L       - lower barrier
%U       - upper barrier
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%M2      - number of intervals in the stock grid
%priceFlag - 1 if price of call option is returned, 0 if probability F(B)
%is returned and else q(z) is returned
%CPflag  - 'c' if call option, put option otherwise

%output:
%price   - the price of the call option if priceFlag == 1, F(B) if
%priceFlag == 0 and q(z)

dt = 1/250; %steplength on the time grid (250 trading days per years)

%initializing:
M1 = 5*M2;
Smin = L;
Smax = U;
dS=(Smax-Smin)/M1; %Because of the rounding
M = M1+1;
N = round(T*(1/dt)); %Number of days
T = N/(1/dt); %To take care of rounding errors
help = [1:5:M];
Mhelp = length(help);

%mean correcting measure:
mc = dt*(r-d+delta*(sqrt(alpha^2 - (beta+1)^2) - sqrt(alpha^2-beta^2)));

%defining the possible stock prices:
Srange = Smin:dS:Smax;

%creating two temporary vectors:
S1 = repmat(Srange, [1, Mhelp]);
S2 = repmat(Srange(help), [M, 1]);
S2 = S2(:)';

%creating the matrix containing all possible z-values:
zFull = log(S2./S1) - mc;
zMatrix = reshape(zFull, M, Mhelp)';

%calculating a matrix containing all possible values of f(z;dt):
fExpMatrix = ((alpha*delta*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) ...
+ beta*zMatrix).* bessell(1, alpha*sqrt((delta*dt)^2 + zMatrix.^2))...
./ (sqrt((delta*dt)^2 + zMatrix.^2)));

%calculating the first step:
zOne = log(Srange/S0) - mc;
g = ((alpha*delta*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) + beta*zOne) ...
.* bessell(1, alpha*sqrt((delta*dt)^2 + zOne.^2))./(sqrt((delta*dt)^2 + ...
zOne.^2)));

gTemp = zeros(1, Mhelp);
for i = 2:N %looping over all points in time
    for j = 1:Mhelp %looping over all possible stock prices in the stock grid

```

```

        h = g.*fExpMatrix(j,:); %making the calculations inside the integral
        gTemp(j) = sum((h(1:(end-1)) + 0.5*diff(h)).*(diff(zOne))); %integrating
    end
    g = interp1(Srange(help), gTemp, Srange, 'cubic');
end

if(priceFlag == 1)
    if (CPflag == 'c') %returning price of call option
        price = exp(-r*T)*trapz(log(Srange/S0),max(0,(Srange-X)) .* g);
    else %returning price of put option
        price = exp(-r*T)*trapz(log(Srange/S0),max(0,(X-Srange)) .* g);
    end
elseif(priceFlag == 0)
    %returning probability F(B)
    price = trapz(log(Srange/S0),g);
else
    price = [g Srange];
end
toc
end

```

```

function [price,lower,upper] = NIGBarrierMC(S0, X, r, d, T, L, U, alpha, beta, ...
    delta, Nsim, CPflag)
tic
%input:
%S0      - initial stock price
%X       - strike price for the barrier option
%r       - interest rate
%d       - dividend yield
%T       - time to maturity (in years)
%L       - lower barrier
%U       - upper barrier
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%Nsim    - number of simulations
%CPflag  - 'c' if price of call option is returned, else price of put
           option is returned

%output:
%price   - the price of the call option if CPflag == 'c', else price of put
%lower   - lower limit of the 95% confidence interval
%upper   - upper limit of the 95% confidence interval

%initializing time grid:
N = round(T*250);
T = N/250;
dt = 1/250;

%mean correcting measure:
m = (r-d+delta*(sqrt(alpha^2 - (beta + 1)^2) - sqrt(alpha^2 - beta^2)));

activeArray = ones(Nsim,1); %array keeping track of whether each process ...
%has passed the barriers
S = S0*ones(Nsim,1); %initial stock price for all simulations

for j = 1:N %looping over all timesteps
    rand = nigrnd(alpha, beta, 0, delta*dt, Nsim, 1); %generates random ...
    %NIG distributed variables
    S = S.*exp(m*dt + rand); %calculating new stock prices

```

```

    %checking if the new stock prices have passed the barrier(s):
    activeArray = activeArray.*(1-(S<=L));
    activeArray = activeArray.*(1-(S>=U));
end
size(S)
size(activeArray)

%calculating price and CI for call and put option:
price_call = exp(-r*T)*max(0,S-X).*activeArray;
price_put = exp(-r*T)*max(0,X-S).*activeArray;
x_bar_call = mean(price_call);
x_bar_put = mean(price_put);
%sig_call = sqrt((1/((Nsim-1)*Nsim))*sum((price_call - x_bar_call).^2));
%sig_put = sqrt((1/((Nsim-1)*Nsim))*sum((price_put - x_bar_put).^2));
sig_call = std(price_call)/sqrt(Nsim);
sig_put = std(price_put)/sqrt(Nsim);
lower_call = x_bar_call - 1.96*sig_call;
lower_put = x_bar_put - 1.96*sig_put;
upper_call = x_bar_call + 1.96*sig_call;
upper_put = x_bar_put + 1.96*sig_put;

if (CPflag == 'c') %returning price of call option:
    price = x_bar_call;
    lower = lower_call;
    upper = upper_call;
else %returning price of put option:
    price = x_bar_put;
    lower = lower_put;
    upper = upper_put;
end
toc
end

```

```

function price = NIGLookbackPIFixed(S0,X, r, d, T, alpha, beta, delta, M2, ...
    dB1, CPflag)
tic
%input:
%S0      - initial stock price
%X       - strike price for the lookback option
%r       - interest rate
%d       - dividend yield
%T       - time to maturity (in years)
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%M2      - number of intervals in the stock grid
%dB1     - gridsize on the Barrier-grid
%CPflag  - 'c' if call option, put option otherwise

%output:
%price   - the price of the option

%defining the Barrier-grid:
if (CPflag == 'c')
    B = [70:dB1:(S0-5) (S0-4):1:(S0+4) (S0+5):dB1:250];
else
    B = [20:dB1:(S0-5) (S0-4):1:(S0+4) (S0+5):dB1:120];
end
length(B)
F = ones(1,length(B));

```

```

%calculating F(B):
for i=1:length(B)
    if (CPflag == 'c')
        F(i) = NIGBarrierPI(S0,X, r, d, T, 10, B(i),alpha, beta, delta, M2, ...
            0, 'c');
        if ((1-F(i) < 0.00001))
            break;
        end
    else
        F(i) = 1 - NIGBarrierPI(S0,X, r, d, T, B(i), 250,alpha, beta, delta, ...
            M2, 0, 'p');
        if (1 - F(i) < 0.00001)
            break;
        end
    end
end
i
toc

end
help = (F < 0);
F(help) = 0;

%interpolating F(B):
dB = 0.005;
Bintpl = B(1):dB:B(end);
Fintpl = interp1(B,F,Bintpl, 'cubic');
n=length(Fintpl);

subplot(2,1,1), plot(Bintpl, Fintpl);
xlabel('Barrier_B*');
ylabel('F(B*)');
title('The_cumulative_distribution_function_F(B*)')

%calculating the pdf f(B):
f = (Fintpl(2:n)-Fintpl(1:(n-1)))/dB;
Bintpl2 = Bintpl(1:(end-1));

a = simpsons(dB, f)
f = f/a;

subplot(2,1,2), plot(Bintpl2, f)
xlabel('Barrier_B*');
ylabel('f(B*)');
title('The_probability_density_function_f(B*)')

%calculating the price:
if (CPflag == 'c')
    price = exp(-r*T)*simpsons(dB,max(0, Bintpl2-X).*f);
else
    price = exp(-r*T)*simpsons(dB,max(0,X-Bintpl2).*f);
end

toc
end

function svar = simpsons(h,y)
svar = (h/3)*(y(1) + y(end) + 4*sum(y(2:2:(end-1))) + 2*sum(y(3:2:(end-2))));
end

```

```

function price = NIGLookbackPIFloating(S0, r, d, T, alpha, beta, delta, M2, ...

```



```

    dB1, CPflag)
tic
%input:
%S0      - initial stock price
%r       - interest rate
%d       - dividend yield
%T       - time to maturity (in years)
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%M2      - number of intervals in the stock grid
%dB1     - gridsize on Barrier calculations
%CPflag  - 'c' if call option, put option otherwise

%output:
%price   - the price of the option

dt = 1/250; %daily monitoring

%calculating mean correcting measure:
mc = dt*(r-d+delta*(sqrt(alpha^2 - (beta+1)^2) - sqrt(alpha^2-beta^2)));

%initializing the grid:
if (CPflag == 'p')
    B = [70:dB1:(S0-5) (S0-4):1:(S0+4) (S0+5):dB1:170];
    dSint = (170-20)/M2;
    Sintp = [20:dSint:170];
    Zintp = log(Sintp/S0)-mc;
else
    B = [30:dB1:(S0-5) (S0-4):1:(S0+4) (S0+5):dB1:150];
    dSint = (240-50)/M2;
    Sintp = [50:dSint:240];
    Zintp = log(Sintp/S0)-mc;
end

%calculating the cumulative probability:
F = zeros(length(Zintp),length(B));
for i=1:length(B)
    if (CPflag == 'p')
        temp = NIGBarrierPI(S0,100, r, d, T, 20, B(i),alpha, beta, delta, ...
            M2, 2, 'c');
        F(:,i) = interp1(log(temp((end/2+1):end)/S0)-mc,temp(1:(end/2)), ...
            Zintp, 'cubic');
    else
        temp = NIGBarrierPI(S0,100, r, d, T, B(i), 240,alpha, beta, delta, ...
            M2, 2, 'p');
        F(:,i) = interp1(log(temp((end/2+1):end)/S0)-mc,temp(1:(end/2)), ...
            Zintp, 'cubic');
    end
end

%interpolating:
dB = 0.005;
Bintpl = B(1):dB:B(end);
[BI,ZI] = meshgrid(Bintpl,Zintp);
Fintpl = interp2(B,Zintp,F,BI,ZI,'cubic');

%finding the partial derivative with respect to B:
if (CPflag == 'p')
    f = (-Fintpl(:,(1:end-1))+Fintpl(:,(2:end)))/dB;
else
    f = (Fintpl(:,(1:end-1))-Fintpl(:,(2:end)))/dB;

```

```

end

%correcting numerical inaccuracy:
if (CPflag == 'p')
    help1 = (repmat(Sintp',1,length(Bintpl2)) > repmat(Bintpl2,length(Sintp),1));
else
    help1 = (repmat(Sintp',1,length(Bintpl2)) < repmat(Bintpl2,length(Sintp),1));
end
f(help1) = 0;
help2 = (f < 0);
f(help2) = 0;

CP = 0;
if (CPflag == 'c')
    CP = 1;
else
    CP = -1;
end

%calculating the price:
temp2 = zeros(1,length(Zintp));
for i=1:length(Zintp)
    temp2(i) = simpson(Bintpl2,max(CP*(Sintp(i) - Bintpl2),0).*f(i,:));
end
price = exp(-r*T)*trapz(Zintp,temp2);

toc
end

function int = simpson(x,y)
n = length(x);
h = (x(end) - x(1))/(n-1);
int = (h/3)*(y(1) + y(end) + 4*sum(y(2:2:(end-1))) + 2*sum(y(3:2:(end-2))));
end

```

```

function [price,lower,upper] = NIGLookbackMC(S0,X, r, d, T, alpha, beta, ...
    delta, Nsim,FixedFlag, CPflag)
tic
%input:
%S0      - initial stock price
%X       - strike price for the barrier option
%r       - interest rate
%d       - dividend yield
%T       - time to maturity (in years)
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%Nsim    - number of simulations
%FixedFlag - 1 if fixed strike price, 0 if floating strike price
%CPflag  - 'c' if price of call option is returned, else price of put
           option is returned

%output:
%price   - the price of the call option if CPflag == 'c', else price of put
%lower   - lower limit of the 95% confidence interval
%upper   - upper limit of the 95% confidence interval

%initializing time grid:
N = round(T*250);
T = N/250;
dt = 1/250;

```

```

%mean correcting measure:
mc = (r-d+delta*(sqrt(alpha^2 - (beta + 1)^2) - sqrt(alpha^2 - beta^2)));

S = S0*ones(Nsim,1);    %initial stock price for all simulations
Smax = zeros(Nsim,1);    %vector keeping track of the maximum ...
    %of all simulations
Smin = 10000*ones(Nsim,1);    %vector keeping track of the ...
    %minimum of all simulations

for j = 1:N    %looping over all time steps
    rand = nigrnd(alpha, beta, 0, delta*dt, Nsim, 1); %generates random ...
    %NIG distributed variables
    S = S.*exp(mc*dt + rand);    %calculating new stock prices

    %updating the minimum and maximum vectors:
    Smax = (S < Smax).*Smax + (1 - (S <= Smax)).*S;
    Smin = (S > Smin).*Smin + (1 - (S >= Smin)).*S;
end

%calculating price and CI for call and put option:
if (CPflag == 'c')
    if (FixedFlag == 1)
        price_vec = exp(-r*T)*max(Smax - X,0);
    else
        price_vec = exp(-r*T)*max(0,S - Smin);
    end
else
    if (FixedFlag == 1)
        price_vec = exp(-r*T)*max(X - Smin,0);
    else
        price_vec = exp(-r*T)*max(0,Smax - S);
    end
end
x_bar = mean(price_vec);
sig = std(price_vec)/sqrt(Nsim);
lower = x_bar - 1.96*sig;
upper = x_bar + 1.96*sig;
price = x_bar;

toc
end

```

```

function [priceFixedCall, priceFixedPut, priceFloatCall, priceFloatPut] = ...
    NIGAsianPI1(S0,X, r, d,dt, T,alpha, beta, delta, M2)
tic
%input:
%S0    - initial stock price
%X     - strike price for the barrier option
%r     - interest rate
%d     - dividend yield
%dt    - number of monitoring points per year
%T     - time to maturity (in years)
%alpha - parameter of the NIG distribution
%beta  - parameter of the NIG distribution
%delta - parameter of the NIG distribution
%M2    - number of intervals in the stock grid

%output:
%priceFixedCall    - the price of the fixed strike call option
%priceFixedPut     - the price of the fixed strike put option

```

```

%priceFloatingCall - the price of the floating strike call option
%priceFloatingPut - the price of the floating strike put option

N = round(T*(1/dt)); %Number of days
T = N/(1/dt); %To take care of rounding errors

%mean correcting measure:
mc = dt*(r-d+delta*(sqrt(alpha^2 - (beta+1)^2) - sqrt(alpha^2-beta^2)));

%initializing grid:
Smin = 5;
Ymin = 5;
Smax = 280;
Ymax = 250;
M1 = 5*M2;
dS=(Smax-Smin)/M1;
dY=(Ymax-Ymin)/M1;
M = M1 + 1;

%to decrease computing time, calculatngs are not made in all gridpoints:
intpl = [1:15:floor(M/4) floor(M/4)+5:5:floor(3*M/4) floor(3*M/4)+15:15:M];
if (intpl(end) == M)
    M = M;
else
    intpl = [intpl M];
end
Mintpl = length(intpl);

%defining the possible stock prices:
Srange = Smin:dS:Smax;
Yrange = Ymin:dY:Ymax;

%creating two temporary vectors:
S1 = repmat(Srange, [1, Mintpl]);
S2 = repmat(Srange(intpl), [M, 1]);
S2 = S2(:)';

%creating the matrix containing all possible z-values:
zFull = log(S2./S1) - mc;
zMatrix = reshape(zFull, M, Mintpl)';

%calculating a matrix containing all possible values of f(z - z'; dt):
fExpMatrix = ((alpha*delta*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) ...
+ beta*zMatrix).* bessell(1, alpha*sqrt((delta*dt)^2 + zMatrix.^2))...
./ (sqrt((delta*dt)^2 + zMatrix.^2)));

%calculating f_{z_1}:
Zrange = log(Srange/S0) - mc;
f1 = ((alpha*delta*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) + beta*Zrange) ...
.* bessell(1, alpha*sqrt((delta*dt)^2 + Zrange.^2))./(sqrt((delta*dt)^2 + ...
Zrange.^2)));

%computing f_{z_2}:
f2Temp = zeros(1, Mintpl);
for j = 1:Mintpl
    h = f1 .* fExpMatrix(j, :);
    f2Temp(j) = trapz(Zrange, h);
end
f2 = interp1(Srange(intpl), f2Temp, Srange, 'cubic');

%calculating f_{y_2, z_2}:
temp1 = (3/S0)*repmat(Yrange, M, 1) - 1 - repmat(Srange', 1, M)/S0;

```

```

help5 = (temp1 <= 0);
temp1(help5) = 10^(-50);
zTemp = log(temp1) - mc;
f2Temp = ((alpha*delta*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) + beta*zTemp) ...
.* besseli(1, alpha*sqrt((delta*dt)^2 + zTemp.^2))./(sqrt((delta*dt)^2 + ...
zTemp.^2)));
f2Temp2 = f2Temp.*abs((1./temp1)*(3/S0));
f = f2Temp2.*repmat(f2',1,M);

%making sure that the initial joint pdf integrates up to 1:
q = trapz(Zrange,f,1);
w = trapz(Yrange,q)
toc

for i = 3:N %looping over all points in time
    fNew = zeros(Mintpl,Mintpl);
    for j = 1:Mintpl %looping over z_{i}
        %interpolating f_{z_{i-1},y_{z_{i-1}}}:
        Yold = ((i+1)/i)*Yrange(intpl) - (1/i)*Srange(intpl(j));
        [YI,SI] = meshgrid(Yold,Srange);
        fTemp = ((i+1)/i)*interp2(Yrange,Srange,f,YI,SI,'cubic');

        %if any values of Yold are outside the grid, Matlab's interp2
        %returns NaN, so these values we set to zero:
        help = isnan(fTemp);
        fTemp(help) = 0;

        %calculating f_{z_{i},y_{i}} for all y_{i}:
        a = repmat(fExpMatrix(j,:),1,Mintpl).*fTemp;
        fNew(j,:) = trapz(Zrange,a,1);
    end

    %interpolating to get values at the entire grid:
    [YI2,SI2] = meshgrid(Yrange,Srange);
    f = interp2(Yrange(intpl),Srange(intpl),fNew,YI2,SI2,'cubic');

    %correcting any possible numerical inaccuracy:
    help3 = (f < 0);
    f(help3) = 0;

    %making sure the joint pdf still integrates up to 1:
    x = trapz(Zrange,f,1);
    x2 = simpson(Yrange,x)
    f = f/x2;
end

%plotting f_{z_{N},y_{N}}:
surf(Yrange(intpl),Srange(intpl),f(intpl,intpl))
xlabel('average')
ylabel('S')

%calculating the prices:
V = zeros(1,M);
for i=1:M
    V(i) = trapz(Zrange,f(:,i)');
end
priceFixedCall = exp(-r*T)*simpson(Yrange,max((Yrange - X),0).*V);
priceFixedPut = exp(-r*T)*simpson(Yrange,max((X - Yrange),0).*V);

V2Call = zeros(1,M);
V2Put = zeros(1,M);
for i=1:M

```

```

V2Call(i) = simpson(Yrange,max((Srange(i) - Yrange),0).*f(i,:));
V2Put(i) = simpson(Yrange,max((Yrange - Srange(i)),0).*f(i,:));
end

priceFloatCall = exp(-r*T)*trapz(Zrange,V2Call);
priceFloatPut = exp(-r*T)*trapz(Zrange,V2Put);

toc
end

```

```

function [priceFixedCall, priceFixedPut, priceFloatCall, priceFloatPut] = ...
    NIGAsianPI2(S0,X, r, d,dt, T,alpha, beta, delta, M2)
tic
%input:
%S0      - initial stock price
%X       - strike price for the barrier option
%r       - interest rate
%d       - dividend yield
%dt      - number of monitoring point during a year
%T       - time to maturity (in years)
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%M2      - number of intervals in the stock grid

%output:
%priceFixedCall    - the price of the fixed strike call option
%priceFixedPut     - the price of the fixed strike put option
%priceFloatingCall - the price of the floating strike call option
%priceFloatingPut  - the price of the floating strike put option

N = round(T*(1/dt)); %Number of days
T = N/(1/dt); %To take care of rounding errors

%mean correcting measure:
mc = dt*(r-d+delta*(sqrt(alpha^2 - (beta+1)^2) - sqrt(alpha^2-beta^2)));

%initializing the dynamic grid:
Ymax = 1.5*S0;
Ymin = 0.5*S0;
for j = 1:N %looping over all timesteps
    Ymax = [Ymax (2.4*j + 1)*S0]; %for set 2 and 1.8*j for set 1
    Ymin = [Ymin (-0.4*j + 1)*S0]; %for set 2 and 0.2*j for set 1
end
Smin = 2;
Smax = 280;
M1 = 5*M2;
dS=(Smax-Smin)/M1;
dY=(Ymax(3)-Ymin(3))/M1;
M = M1 + 1;

%to decrease computing time, calculatngs are not made in all gridpoints:
intpl = [1:5:floor(3*M/4) floor(3*M/4)+15:15:M];
if (intpl(end) == M)
    M = M;
else
    intpl = [intpl M];
end
Mintpl = length(intpl);

%defining the possible stock prices:

```

```

Srange = Smin:dS:Smax;
Yrange = Ymin(3):dY:Ymax(3);

%creating two temporary vectors:
S1 = repmat(Srange, [1,Mintpl]);
S2 = repmat(Srange(intpl), [M,1]);
S2 = S2(:)';

%creating the matrix containing all possible z-values:
zFull = log(S2./S1) - mc;
zMatrix = reshape(zFull,M,Mintpl)';

%calculating a matrix containing all possible values of f(z-z';dt):
fExpMatrix = ((alpha*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) ...
+ beta*zMatrix).* bessell(1, alpha*sqrt((delta*dt)^2 + zMatrix.^2))...
./((sqrt((delta*dt)^2 + zMatrix.^2)));

%calculating f_{z_1}:
Zrange = log(Srange/S0) - mc;
f1 = ((alpha*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) + beta*Zrange) ...
.* bessell(1, alpha*sqrt((delta*dt)^2 + Zrange.^2))./(sqrt((delta*dt)^2 + ...
Zrange.^2)));

%computing f_{z_2}:
f2Temp = zeros(1,Mintpl);
for j = 1:Mintpl
    h = f1.*fExpMatrix(j,:);
    f2Temp(j) = trapz(Zrange,h);
end
f2 = interp1(Srange(intpl), f2Temp, Srange, 'cubic');

%calculating f_{y_2,z_2}:
temp1 = (1/S0)*repmat(Yrange,M,1) - 1 - repmat(Srange',1,M)/S0;
help5 = (temp1 <= 0);
temp1(help5) = 10^(-50);
zTemp = log(temp1) - mc;
f2Temp = ((alpha*dt/pi) * exp(delta*dt*sqrt(alpha^2-beta^2) + beta*zTemp) ...
.* bessell(1, alpha*sqrt((delta*dt)^2 + zTemp.^2))./(sqrt((delta*dt)^2 + ...
zTemp.^2)));
f2Temp2 = f2Temp.*abs((1./temp1)*(1/S0));
f = f2Temp2.*repmat(f2',1,M);

%making sure that the initial joint pdf integrates up to 1:
q = trapz(Zrange,f,1);
w = trapz(Yrange,q)
toc

for i = 3:N %looping over all points in time
    %changing the Y-grid:
    dY=(Ymax(i+1)-Ymin(i+1))/M1;
    YrangeOld = Yrange;
    Yrange = Ymin(i+1):dY:Ymax(i+1);
    fNew = zeros(Mintpl,Mintpl);
    for j = 1:Mintpl %looping over Z
        %interpolating f_{z_{i-1},y_{i-1}}:
        Yold = Yrange(intpl) - Srange(intpl(j));
        [YI,SI] = meshgrid(Yold,Srange);
        fTemp = interp2(YrangeOld,Srange,f,YI,SI,'cubic');

        %if any values of Yold are outside the grid, Matlab's interp2
        %returns NaN, so these values we set to zero:
        help = isnan(fTemp);

```

```

    fTemp(help) = 0;

    %calculating  $f_{\{z_{\{i\}}, y_{\{i\}}\}}$  for all  $y_{\{i\}}$ :
    a = repmat(fExpMatrix(j, :)', 1, Mintpl).*fTemp;
    fNew(j, :) = trapz(Zrange, a, 1);
end

    %interpolating to get values at the entire grid:
    [YI2, SI2] = meshgrid(Yrange, Srange);
    f = interp2(Yrange(intpl), Srange(intpl), fNew, YI2, SI2, 'cubic');

    %correcting any possible numerical inaccuracy:
    help3 = (f < 0);
    f(help3) = 0;

    %making sure the joint pdf still integrates up to 1:
    x = trapz(Zrange, f, 1);
    x2 = trapz(Yrange, x);
    f = f/x2;
end

    %plotting  $f_{\{z_{\{N\}}, y_{\{N\}}\}}$ :
    surf(Yrange(intpl), Srange(intpl), f(intpl, intpl))
    xlabel('average')
    ylabel('S')

    %calculating the prices:
    V = zeros(1, M);
    for i=1:M
        V(i) = trapz(Zrange, f(:, i)');
    end
    priceFixedCall = exp(-r*T)*simpson(Yrange, max((1/(N+1))*Yrange - X), 0).*V);
    priceFixedPut = exp(-r*T)*simpson(Yrange, max((X - (1/(N+1))*Yrange), 0).*V);

    V2Call = zeros(1, M);
    V2Put = zeros(1, M);
    for i=1:M
        V2Call(i) = simpson(Yrange, max((Srange(i) - (1/(N+1))*Yrange), 0).*f(i, :));
        V2Put(i) = simpson(Yrange, max((1/(N+1))*Yrange - Srange(i)), 0).*f(i, :));
    end

    priceFloatCall = exp(-r*T)*trapz(Zrange, V2Call);
    priceFloatPut = exp(-r*T)*trapz(Zrange, V2Put);
toc
end

```

```

function [price, lower, upper] = NIGAsianMC(S0, X, r, d, dt, T, alpha, beta, ...
    delta, Nsim, FixedFlag, CPflag)
tic
%input:
%S0      - initial stock price
%X       - strike price for the barrier option
%r       - interest rate
%d       - dividend yield
%dt      - number of monitoring points during a year
%T       - time to maturity (in years)
%alpha   - parameter of the NIG distribution
%beta    - parameter of the NIG distribution
%delta   - parameter of the NIG distribution
%Nsim    - number of simulations
%FixedFlag - 1 if fixed strike price, 0 if floating strike price

```



```

%CPflag    - 'c' if price of call option is returned, else price of put
%option is returned

%output:
%price     - the price of the call option if CPflag == 'c', else price of put
%lower     - lower limit of the 95% confidence interval
%upper     - upper limit of the 95% confidence interval

%initializing time grid:
N = round(T/dt)
T = N*dt;

%mean correcting measure:
m = r-d+delta*(sqrt(alpha^2 - (beta + 1)^2) - sqrt(alpha^2 - beta^2));

S = S0*ones(Nsim,1); %initial stock price for all simulations
sum1 = S0*ones(Nsim,1); %keeping track of the growing sums

for j = 1:N %looping over all timesteps
    rand = nigrnd(alpha, beta, 0, delta*dt, Nsim, 1); %generates random ...
        %NIG distributed variables
    S = S.*exp(m*dt + rand); %calculating new stock prices
    sum1 = sum1 + S; %updating the sums
end

%calculating the averages:
S_av = sum1/(N+1);

CP = 0; %binary variable: 1 if call, -1 if put
if(CPflag == 'c')
    CP = 1;
else
    CP = -1;
end

%calculating the option prices of each simulation:
if (FixedFlag == 1)
    priceVec = exp(-r*T)*max(CP*(S_av - X),0);
else
    priceVec = exp(-r*T)*max(CP*(S - S_av),0);
end

%calculating the outoput values:
x_bar = mean(priceVec);
sig = std(priceVec)/sqrt(Nsim)
lower = x_bar - 1.96*sig;
upper = x_bar + 1.96*sig;
price = x_bar;

toc
end

```

Appendix B

Option Prices

Table B.1-B.4 show the option data used for the calibration in chapter 4. All tables show the calculated prices using the implied parameters obtained by the four different error measures, $ARPE_{spread}$, $ARPE$, $RMSE$ and AAE . The market data is collected from Bloomberg on 27th of April 2009.

T	Strike	Bid	Ask	$NIG_{ARPE_{spread}}$	NIG_{ARPE}	NIG_{RMSE}	NIG_{AAE}
0.07	147	69	72.75	71.07	71.06	71.13	71.12
0.07	150	66	69.75	68.12	68.11	68.18	68.17
0.07	155	61	64.75	63.21	63.19	63.27	63.26
0.07	160	56.25	60	58.32	58.3	58.39	58.38
0.07	165	51.25	55	53.46	53.44	53.54	53.53
0.07	170	46.5	50.25	48.63	48.61	48.72	48.71
0.07	175	41.5	45.25	43.84	43.82	43.95	43.93
0.07	180	36.75	40.5	39.12	39.09	39.23	39.21
0.07	185	32	35.75	34.46	34.44	34.58	34.56
0.07	190	28	31	29.9	29.88	30.02	29.99
0.07	195	23.5	26	25.47	25.45	25.59	25.55
0.07	200	19.75	21.5	21.2	21.18	21.31	21.26
0.07	205	16	17.75	17.15	17.14	17.24	17.18
0.07	210	12.25	14	13.4	13.39	13.45	13.39
0.07	215	9.5	10.5	10.03	10.04	10.04	9.99
0.07	220	6.9	7.5	7.17	7.19	7.12	7.09
0.07	225	4.7	5.9	4.91	4.93	4.81	4.81
0.07	230	2.95	4.3	3.26	3.27	3.13	3.18
0.07	235	2.05	2.45	2.13	2.14	2.01	2.08
0.07	240	1.25	1.65	1.4	1.4	1.29	1.38
0.07	245	0.7	1.1	0.93	0.93	0.85	0.93
0.15	165	52.5	56.25	54.58	54.55	54.75	54.69
0.15	175	43.5	47.25	45.44	45.41	45.65	45.58

0.15	180	39	42.75	41.02	40.98	41.23	41.16
0.15	185	34.75	38.5	36.71	36.68	36.92	36.85
0.15	190	30.5	34.25	32.55	32.52	32.75	32.68
0.15	195	26.75	29.25	28.55	28.52	28.74	28.67
0.15	200	23	25.5	24.75	24.72	24.92	24.84
0.15	205	20	21.75	21.17	21.15	21.32	21.24
0.15	210	16.5	18.25	17.85	17.83	17.96	17.88
0.15	215	13.5	15.25	14.82	14.81	14.89	14.82
0.15	220	11	12.75	12.1	12.1	12.13	12.07
0.15	225	9	10	9.72	9.73	9.7	9.67
0.15	230	6.7	7.9	7.68	7.69	7.62	7.61
0.15	235	5.5	6	5.98	5.99	5.89	5.91
0.15	240	4.1	4.6	4.6	4.61	4.48	4.52
0.15	245	3.1	3.45	3.49	3.51	3.37	3.43
0.23	180	41	44.75	42.66	42.62	42.98	42.83
0.23	185	37	40.75	38.62	38.58	38.93	38.79
0.23	190	33	36.75	34.74	34.7	35.04	34.89
0.23	195	30	31	31.02	30.99	31.31	31.17
0.23	200	26.5	27.5	27.5	27.46	27.76	27.62
0.23	205	23	24	24.17	24.14	24.41	24.28
0.23	210	20	21	21.07	21.05	21.28	21.15
0.23	215	17	18	18.2	18.19	18.37	18.25
0.23	220	14.5	15.25	15.58	15.57	15.71	15.6
0.23	225	12	13.5	13.21	13.2	13.3	13.21
0.23	230	10	10.75	11.09	11.09	11.14	11.07
0.23	235	8.25	9	9.23	9.23	9.24	9.19
0.23	240	6.7	7.2	7.61	7.61	7.58	7.55
0.23	245	5.3	5.9	6.21	6.22	6.16	6.15

Table B.1: OBX Call Option prices

T	Strike	Bid	Ask	$NIG_{ARPE_{spread}}$	NIG_{ARPE}	NIG_{RMSE}	NIG_{AAE}
0.06	30	47.7	52.5	50.14	50.14	50.14	50.14
0.06	31	46.7	51.5	49.14	49.14	49.14	49.14
0.06	32	45.7	50.5	48.14	48.15	48.14	48.14
0.06	33	44.7	49.5	47.15	47.15	47.15	47.15
0.06	34	43.7	48.5	46.15	46.15	46.15	46.15
0.06	35	42.65	47.45	45.16	45.16	45.16	45.16
0.06	36	41.65	46.45	44.16	44.16	44.16	44.16
0.06	37	40.65	45.45	43.17	43.17	43.17	43.17

0.06	38	39.65	44.45	42.17	42.17	42.17	42.17
0.06	39	38.65	43.45	41.18	41.18	41.18	41.17
0.06	40	37.7	42.5	40.19	40.18	40.18	40.18
0.06	41	36.7	41.5	39.19	39.19	39.19	39.18
0.06	42	35.7	40.5	38.2	38.19	38.2	38.19
0.06	43	34.7	39.5	37.2	37.2	37.2	37.19
0.06	44	33.7	38.5	36.21	36.2	36.21	36.2
0.06	45	32.7	37.5	35.21	35.21	35.21	35.2
0.06	46	31.7	36.5	34.22	34.22	34.22	34.21
0.06	47	30.7	35.5	33.23	33.22	33.23	33.22
0.06	48	29.7	34.5	32.23	32.23	32.23	32.22
0.06	49	28.7	33.5	31.24	31.23	31.24	31.23
0.06	50	27.7	32.5	30.25	30.24	30.24	30.23
0.16	49	28.5	33.3	31.22	31.2	31.21	31.18
0.16	50	27.55	32.35	30.24	30.22	30.23	30.2
0.16	51	26.6	31.4	29.26	29.23	29.25	29.22
0.16	52	25.6	30.4	28.28	28.25	28.27	28.23
0.16	53	24.6	29.4	27.3	27.27	27.29	27.25
0.16	54	23.6	28.4	26.32	26.29	26.31	26.27
0.16	55	22.65	27.45	25.34	25.31	25.33	25.29
0.16	56	21.65	26.45	24.36	24.33	24.35	24.31
0.16	57	20.7	25.5	23.39	23.35	23.38	23.33
0.16	58	19.7	24.5	22.41	22.38	22.4	22.35
0.16	59	18.75	23.55	21.43	21.4	21.42	21.37
0.16	60	17.8	22.6	20.46	20.42	20.45	20.4
0.16	61	16.85	21.65	19.49	19.45	19.48	19.42
0.16	62	15.9	20.7	18.52	18.47	18.5	18.45
0.16	63	14.95	19.75	17.54	17.5	17.53	17.47
0.16	64	14	18.8	16.57	16.53	16.56	16.5
0.16	65	13.1	17.9	15.61	15.56	15.59	15.53
0.16	66	12.15	16.95	14.64	14.59	14.63	14.56
0.16	67	11.3	16.1	13.68	13.62	13.66	13.59
0.16	68	10.4	15.2	12.71	12.66	12.7	12.62
0.16	69	9.55	14.35	11.75	11.7	11.74	11.66
0.24	30	47.3	52.1	49.75	49.76	49.75	49.76
0.24	35	42.3	47.1	44.86	44.86	44.85	44.85
0.24	40	37.35	42.15	39.97	39.96	39.96	39.94
0.24	45	32.45	37.25	35.09	35.07	35.08	35.05
0.24	46	31.45	36.25	34.11	34.09	34.11	34.07
0.24	47	30.45	35.25	33.14	33.12	33.13	33.1
0.24	48	29.5	34.3	32.17	32.14	32.16	32.12
0.24	49	28.5	33.3	31.2	31.17	31.19	31.15
0.24	50	27.5	32.3	30.22	30.2	30.21	30.17

0.24	51	26.55	31.35	29.25	29.22	29.24	29.2
0.24	52	25.55	30.35	28.28	28.25	28.27	28.22
0.24	53	24.6	29.4	27.31	27.28	27.3	27.25
0.24	54	23.6	28.4	26.35	26.31	26.34	26.28
0.24	55	22.65	27.45	25.38	25.34	25.37	25.31
0.24	56	21.7	26.5	24.41	24.37	24.4	24.34
0.24	57	20.75	25.55	23.45	23.4	23.44	23.37
0.24	58	19.8	24.6	22.49	22.44	22.47	22.4
0.24	59	18.85	23.65	21.52	21.47	21.51	21.43
0.24	60	17.9	22.7	20.56	20.51	20.55	20.47
0.24	61	17	21.8	19.6	19.54	19.59	19.5
0.24	62	16.1	20.9	18.64	18.58	18.63	18.54
0.42	30	46.85	51.65	49.35	49.37	49.35	49.37
0.42	31	45.85	50.65	48.39	48.4	48.39	48.4
0.42	32	44.85	49.65	47.42	47.44	47.42	47.43
0.42	33	43.9	48.7	46.46	46.47	46.46	46.46
0.42	34	42.9	47.7	45.49	45.5	45.49	45.49
0.42	35	41.9	46.7	44.53	44.54	44.53	44.52
0.42	36	40.9	45.7	43.57	43.57	43.56	43.55
0.42	37	39.9	44.7	42.61	42.61	42.6	42.59
0.42	38	38.95	43.75	41.64	41.64	41.64	41.62
0.42	39	37.95	42.75	40.68	40.68	40.68	40.66
0.42	40	37	41.8	39.72	39.72	39.72	39.69
0.42	41	36	40.8	38.77	38.75	38.76	38.73
0.42	42	35.05	39.85	37.81	37.79	37.8	37.77
0.42	43	34.05	38.85	36.85	36.83	36.84	36.8
0.42	44	33.1	37.9	35.89	35.87	35.88	35.84
0.42	45	32.15	36.95	34.94	34.91	34.93	34.88
0.42	46	31.15	35.95	33.98	33.96	33.97	33.92
0.42	47	30.2	35	33.03	33	33.02	32.96
0.42	48	29.25	34.05	32.08	32.04	32.07	32.01
0.42	49	28.3	33.1	31.13	31.09	31.11	31.05
0.42	50	27.35	32.15	30.18	30.13	30.16	30.09
0.68	30	46.3	51.1	48.75	48.8	48.76	48.79
0.68	31	45.3	50.1	47.81	47.85	47.81	47.84
0.68	32	44.35	49.15	46.87	46.91	46.87	46.89
0.68	33	43.35	48.15	45.93	45.96	45.93	45.95
0.68	34	42.35	47.15	44.99	45.01	44.99	45
0.68	35	41.4	46.2	44.05	44.07	44.04	44.05
0.68	36	40.45	45.25	43.11	43.13	43.11	43.11
0.68	37	39.45	44.25	42.17	42.18	42.17	42.16
0.68	38	38.5	43.3	41.23	41.24	41.23	41.22
0.68	39	37.55	42.35	40.3	40.3	40.29	40.28

0.68	40	36.6	41.4	39.37	39.36	39.36	39.33
0.68	41	35.65	40.45	38.43	38.43	38.42	38.39
0.68	42	34.7	39.5	37.5	37.49	37.49	37.45
0.68	43	33.75	38.55	36.57	36.55	36.56	36.52
0.68	44	32.8	37.6	35.64	35.62	35.63	35.58
0.68	45	31.9	36.7	34.72	34.69	34.7	34.64
0.68	46	30.95	35.75	33.79	33.76	33.78	33.71
0.68	47	30.05	34.85	32.87	32.83	32.85	32.78
0.68	48	29.15	33.95	31.95	31.9	31.93	31.85
0.68	49	28.2	33	31.02	30.97	31.01	30.92
0.68	50	27.35	32.15	30.11	30.05	30.09	29.99
0.94	30	45.85	50.65	48.15	48.23	48.16	48.23
0.94	31	44.9	49.7	47.23	47.3	47.23	47.3
0.94	32	43.9	48.7	46.31	46.38	46.31	46.37
0.94	33	42.95	47.75	45.39	45.45	45.4	45.44
0.94	34	42	46.8	44.47	44.53	44.48	44.52
0.94	35	41.05	45.85	43.56	43.61	43.56	43.59
0.94	36	40.1	44.9	42.65	42.69	42.65	42.67
0.94	37	39.15	43.95	41.74	41.77	41.73	41.75
0.94	38	38.25	43.05	40.83	40.85	40.82	40.83
0.94	39	37.3	42.1	39.92	39.94	39.91	39.91
0.94	40	36.4	41.2	39.01	39.03	39	38.99
0.94	41	35.5	40.3	38.1	38.11	38.1	38.08
0.94	42	34.55	39.35	37.2	37.2	37.19	37.16
0.94	43	33.65	38.45	36.3	36.29	36.29	36.25
0.94	44	32.75	37.55	35.4	35.39	35.39	35.34
0.94	45	31.85	36.65	34.5	34.48	34.49	34.43
0.94	46	30.95	35.75	33.61	33.58	33.59	33.52
0.94	47	30.05	34.85	32.71	32.68	32.69	32.62
0.94	48	29.2	34	31.82	31.78	31.8	31.72
0.94	49	28.3	33.1	30.93	30.88	30.91	30.81
0.94	50	27.45	32.25	30.05	29.99	30.02	29.92
1.2	40	36.2	41	38.59	38.63	38.59	38.6
1.2	41	35.3	40.1	37.71	37.74	37.7	37.71
1.2	42	34.4	39.2	36.83	36.86	36.83	36.82
1.2	43	33.55	38.35	35.96	35.98	35.95	35.93
1.2	44	32.65	37.45	35.09	35.09	35.08	35.05
1.2	45	31.8	36.6	34.22	34.22	34.2	34.16
1.2	46	30.95	35.75	33.35	33.34	33.34	33.28
1.2	47	30.05	34.85	32.49	32.46	32.47	32.4
1.2	48	29.25	34.05	31.62	31.59	31.6	31.53
1.2	49	28.4	33.2	30.76	30.72	30.74	30.65
1.2	50	27.55	32.35	29.91	29.86	29.89	29.78

1.2	51	26.75	31.55	29.06	29	29.03	28.91
1.2	52	25.9	30.7	28.21	28.14	28.18	28.05
1.2	53	25.1	29.9	27.36	27.28	27.33	27.19
1.2	54	24.35	29.15	26.52	26.42	26.48	26.33
1.2	55	23.55	28.35	25.68	25.57	25.64	25.47
1.2	56	22.75	27.55	24.84	24.73	24.8	24.62
1.2	57	22	26.8	24.01	23.89	23.97	23.77
1.2	58	21.25	26.05	23.19	23.05	23.14	22.93
1.2	59	20.5	25.3	22.36	22.21	22.32	22.09
1.2	60	21.15	23	21.55	21.39	21.5	21.25
1.7	40	35.8	40.6	37.93	38.08	37.95	38.08
1.7	41	34.95	39.75	37.12	37.25	37.13	37.24
1.7	42	34.1	38.9	36.3	36.42	36.31	36.41
1.7	43	33.3	38.1	35.49	35.59	35.5	35.58
1.7	44	32.45	37.25	34.68	34.77	34.68	34.75
1.7	45	31.65	36.45	33.87	33.95	33.88	33.93
1.7	46	30.8	35.6	33.07	33.14	33.07	33.1
1.7	47	30	34.8	32.27	32.32	32.27	32.28
1.7	48	29.2	34	31.47	31.51	31.47	31.47
1.7	49	28.45	33.25	30.68	30.71	30.67	30.65
1.7	50	27.65	32.45	29.89	29.9	29.87	29.84
1.7	51	26.9	31.7	29.1	29.1	29.09	29.04
1.7	52	26.15	30.95	28.32	28.31	28.3	28.23
1.7	53	25.35	30.15	27.54	27.51	27.52	27.43
1.7	54	24.6	29.4	26.76	26.72	26.74	26.64
1.7	55	23.9	28.7	25.99	25.94	25.96	25.84
1.7	56	23.15	27.95	25.23	25.16	25.2	25.06
1.7	57	22.45	27.25	24.46	24.38	24.43	24.27
1.7	58	23.1	25	23.71	23.61	23.67	23.5
1.7	59	22.4	24.3	22.96	22.85	22.92	22.72
1.7	60	21.75	23.65	22.21	22.09	22.17	21.95

Table B.2: DJI Call Option prices

T	Strike	Bid	Ask	$NIG_{ARPE_{spread}}$	NIG_{ARPE}	NIG_{RMSE}	NIG_{AAE}
0.05	37.5	4.8	4.9	4.87	4.82	4.78	4.84
0.05	40	2.55	2.6	2.59	2.51	2.5	2.54
0.05	42.5	0.8	0.9	0.77	0.73	0.82	0.73
0.05	45	0.1	0.15	0.07	0.1	0.11	0.09
0.06	32.5	9.7	9.9	9.76	9.75	9.74	9.76

0.06	35	7.3	7.5	7.3	7.27	7.25	7.29
0.06	37.5	5.1	5.2	4.9	4.84	4.8	4.87
0.06	40	3	3.2	2.66	2.57	2.57	2.61
0.06	42.5	1.5	1.6	0.89	0.84	0.92	0.84
0.06	45	0.55	0.6	0.11	0.15	0.17	0.14
0.06	47.5	0.1	0.2	0.01	0.02	0.01	0.02
0.32	25	17.2	17.6	17.26	17.25	17.24	17.26
0.32	27.5	14.6	15.1	14.79	14.76	14.75	14.78
0.32	30	12.3	12.5	12.35	12.3	12.27	12.33
0.32	32.5	9.9	10.1	9.97	9.88	9.84	9.93
0.32	35	7.6	7.8	7.72	7.57	7.53	7.65
0.32	37.5	5.6	5.7	5.64	5.46	5.43	5.55
0.32	40	3.8	3.9	3.84	3.64	3.65	3.73
0.32	42.5	2.35	2.45	2.38	2.21	2.26	2.28
0.32	45	1.3	1.35	1.33	1.21	1.29	1.25
0.32	47.5	0.6	0.7	0.65	0.6	0.67	0.62
0.32	50	0.25	0.3	0.27	0.27	0.31	0.28
0.32	52.5	0.05	0.15	0.1	0.12	0.13	0.11
0.32	55	0.05	0.1	0.03	0.05	0.05	0.04
0.58	25	17.2	17.4	17.32	17.29	17.28	17.31
0.58	27.5	14.8	15	14.9	14.84	14.81	14.87
0.58	30	12.4	12.6	12.55	12.44	12.4	12.5
0.58	32.5	10.2	10.4	10.3	10.14	10.09	10.22
0.58	35	8.2	8.3	8.21	7.99	7.95	8.1
0.58	37.5	6.3	6.4	6.32	6.06	6.04	6.19
0.58	40	4.7	4.8	4.68	4.41	4.42	4.54
0.58	42.5	3.3	3.5	3.32	3.07	3.11	3.18
0.58	45	2.2	2.35	2.24	2.04	2.1	2.13
0.58	47.5	1.4	1.5	1.44	1.3	1.36	1.36
0.58	50	0.8	0.9	0.88	0.79	0.85	0.82
0.58	52.5	0.45	0.55	0.51	0.46	0.51	0.48
0.58	55	0.2	0.3	0.28	0.26	0.3	0.27
0.74	20	22.2	22.6	22.31	22.3	22.29	22.3
0.74	22.5	19.7	20.1	19.84	19.82	19.81	19.83
0.74	25	17.2	17.6	17.4	17.35	17.33	17.38
0.74	27.5	14.8	15.2	15.02	14.93	14.9	14.98
0.74	30	12.5	12.9	12.72	12.58	12.54	12.66
0.74	20	22.2	22.6	22.31	22.3	22.29	22.3
0.74	22.5	19.7	20.1	19.84	19.82	19.81	19.83
0.74	25	17.2	17.6	17.4	17.35	17.33	17.38
0.74	27.5	14.8	15.2	15.02	14.93	14.9	14.98
0.74	30	12.5	12.9	12.72	12.58	12.54	12.66
0.74	32.5	10.2	10.7	10.54	10.35	10.3	10.45

0.74	35	8.2	8.6	8.53	8.28	8.24	8.41
0.74	37.5	6.5	6.9	6.72	6.43	6.41	6.57
0.74	40	5	5.3	5.14	4.84	4.85	4.98
0.74	42.5	3.8	3.9	3.81	3.52	3.56	3.66
0.74	45	2.7	2.85	2.73	2.48	2.54	2.6
0.74	47.5	1.8	1.95	1.89	1.7	1.76	1.78
0.74	50	1.15	1.3	1.27	1.12	1.19	1.18
0.74	52.5	0.7	0.85	0.82	0.73	0.79	0.76
0.74	55	0.4	0.5	0.51	0.46	0.51	0.48
0.74	60	0.1	0.2	0.18	0.17	0.2	0.18
1.74	17.5	24.5	25.3	25.26	25.23	25.22	25.24
1.74	20	22	22.9	22.89	22.83	22.82	22.86
1.74	22.5	19.5	20	20.58	20.48	20.45	20.53
1.74	25	17.2	17.7	18.34	18.19	18.15	18.27
1.74	27.5	15	15.6	16.2	15.99	15.94	16.1
1.74	30	13	13.8	14.18	13.9	13.86	14.05
1.74	32.5	11.1	11.7	12.3	11.97	11.92	12.14
1.74	35	9.5	10.1	10.58	10.19	10.16	10.39
1.74	37.5	8	8.4	9.01	8.59	8.57	8.81
1.74	40	6.9	7.3	7.61	7.17	7.17	7.4
1.74	42.5	5.8	6.2	6.38	5.94	5.95	6.16
1.74	45	4.5	5.1	5.3	4.87	4.9	5.08
1.74	47.5	3.6	4.2	4.37	3.96	4	4.16
1.74	50	3	3.4	3.58	3.2	3.25	3.38
1.74	55	1.75	2.1	2.35	2.05	2.11	2.19
1.74	60	1.1	1.3	1.5	1.28	1.35	1.38
1.74	65	0.5	0.7	0.94	0.79	0.84	0.86
1.74	70	0.3	0.55	0.58	0.48	0.52	0.52

Table B.3: KO Call Option prices

T	Strike	Bid	Ask	$NIG_{ARPE_{spread}}$	NIG_{ARPE}	NIG_{RMSE}	NIG_{AAE}
0.15	80	39	42	40.4	40.92	40.47	40.42
0.15	85	34.25	36.75	35.49	36.18	35.57	35.51
0.15	90	29.5	31.25	30.65	31.51	30.75	30.65
0.15	95	24.75	26.5	25.91	26.92	26.02	25.9
0.15	100	20.25	22	21.32	22.44	21.45	21.3
0.15	105	15.75	17.25	16.96	18.11	17.1	16.94
0.15	110	11.75	13	12.94	14	13.05	12.91
0.15	115	8.25	9.5	9.4	10.22	9.4	9.35

0.15	120	5.5	6.7	6.46	6.92	6.27	6.38
0.15	125	3.5	4.4	4.21	4.34	3.75	4.08
0.15	130	2.15	3.1	2.62	2.62	1.92	2.44
0.15	140	0.75	1.15	0.95	1	0.22	0.75
0.15	150	0.15	0.65	0.34	0.42	0	0.21
0.15	160	0.01	0.5	0.12	0.18	0	0.06
0.15	170	0.01	0.45	0.04	0.08	0	0.02
0.15	180	0.01	0.5	0.01	0.03	0	0
0.41	80	39	42	40.11	41.09	41.22	40.5
0.41	85	34.5	37	35.54	36.81	36.61	35.89
0.41	90	30.25	32.75	31.13	32.65	32.15	31.45
0.41	95	25.75	27.5	26.93	28.65	27.88	27.2
0.41	100	21.75	23.5	22.98	24.82	23.84	23.21
0.41	105	18.25	19.75	19.32	21.2	20.07	19.51
0.41	110	14.75	16.5	15.99	17.82	16.6	16.13
0.41	115	12.25	13.5	13.03	14.71	13.47	13.12
0.41	120	9.75	10.75	10.44	11.9	10.7	10.48
0.41	125	7.7	8.75	8.23	9.44	8.29	8.22
0.41	130	6	6.9	6.37	7.32	6.26	6.33
0.41	140	3.3	4.3	3.64	4.15	3.25	3.54
0.41	150	2	2.5	1.93	2.19	1.46	1.83
0.41	160	0.75	1.4	0.95	1.08	0.54	0.86
0.41	170	0.3	0.8	0.41	0.47	0.16	0.35
0.41	180	0.01	0.55	0.14	0.17	0.03	0.12
0.67	95	27	31	26.49	28.09	28.51	27
0.67	105	20.25	23	19.76	21.63	21.44	20.14
0.67	110	17.75	20	16.81	18.7	18.3	17.13
0.67	115	15	17.25	14.15	15.99	15.44	14.41
0.67	120	12.5	14.75	11.79	13.52	12.88	11.99
0.67	125	10.75	12.75	9.7	11.28	10.59	9.86
0.67	130	9	10.5	7.89	9.29	8.59	8
0.67	140	5.8	7.3	5.02	6.03	5.39	5.06
0.67	150	3.8	5.4	3	3.64	3.14	3
0.67	160	2.1	3.65	1.64	2.01	1.66	1.63
0.67	170	1.3	2.2	0.79	0.98	0.76	0.77
0.67	180	0.7	1.45	0.3	0.38	0.27	0.29
0.67	190	0.3	1.05	0.07	0.08	0.05	0.06
0.67	200	0.01	0.75	0	0	0	0

Table B.4: STL Call Option prices