Discrete space-simulation for Lévy processes

Simon Johansson

June 4, 2012

Abstract

In this thesis I will present a way of discretizing Lévy processes in space instead of in time. The foundation is built on work done by Adalbjörnsson,Quiroz and Wiktorsson, which shows how this is done for Brownian motions with constant drift and volatility. I then start by extending the method to multidimensional Brownian motions, which is then extended to multidimensional SDE:s by using an Euler approximation. The method is then extended to Jump-Diffusions. I also present an approximation method for approximating Infinite activity processes with Jump-Diffusions, and as result the simulation method is extended to Infinite activity processes. Since the method bounds process in space it's natural to consider path-dependent options. Case studies on Barrier options are performed in order to show the convergence of the algorithm.

Acknowledgement

I would like to thank my supervisor, Magnus Wiktorsson, for his help and guidance throughout this thesis.

Contents

Chapter 1

Introduction

1.1 Background

In Adalbjörnsson and Quiroz [1] a method for simulating exotic/path-dependent options is presented. The method consists of simulating the time it takes for a process to hit a space grid, i.e. instead of discretizing the process equidistantly in time it's discretized equidistantly in space. The purpose of this is that it automatically bounds the process in space, this is something that extends to path-dependent options, especially Barrier options, very naturally.

In Adalbjörnsson and Quiroz the method is, basically, only adapted to the Black-Scholes model and in this thesis I will extend the results to more advanced models. The models that will be considered are the Heston model, the Merton model and the Variance Gamma model.

1.2 Purpose

The main purpose of this thesis is to construct a framework for pricing exotic options, for stock models based on Lévy process, via a simulation method. The method will be tested in four different stock models, representing four different types of stock models. In each of the models it's performance will be investigated. It should be noted that the purpose is not to find the most efficient method in each case but rather to find a, hopefully, robust method for pricing exotic options.

1.3 Outline

The thesis is split into two main parts. Before that a brief mathematical background is presented. In the background some important stochastic processes are introduced. The reader is expected to be familiar with Stochastic Differential Equations (if not see Åberg [4]). The reader is also expected to be familiar with concepts such as arbitrage pricing and risk-neutral valuation, all parameters in this thesis will be considered to be risk neutral (for more on this see Björk [4], for BMs, or Cont and Tankov [5], for Lévy processes). The thesis is split into

two parts, in part I the simulation method is presented for Brownian Motions, firstly in the one-dimensional case (chapter 3) and the results are then extended to the multi-dimensional case (chapter 4).

Part II deals with extending the simulation method to Lévy processes. Chapter 5 presents a theoretical walkthrough on Lévy processes. In chapter 6 the results are from chapter 3 and 4 are extended to Jump-Diffusions. Chapter 7 presents a way for simulating Infinite activity processes by approximating them with Jump-Diffusions. In chapter 3, 4, 6 and 7 case studies are presented in order to show the performance of the algorithm.

Chapter 2

Mathematical Background

In this section the stochastic processes used in the thesis are presented. It starts by introducing Brownian Motions and presenting some useful properties for them, then Poisson process is introduced and finally the gamma process is defined.

2.1 Brownian Motion

One of the most widely used, especially in Financial modeling, stochastic process is the Wiener process.

Definition 2.1. *A Wiener process, W, is a stochastic process satisfying the following conditions*

1. *W(0) = 0 (almost surely).*
2. *W has independent increments.*
3. *$W(t+h) - W(t)$ is normally distributed with mean 0 and variance h, for $h > 0$.*
4. *W has continuous trajectories.*

The Wiener process is also known as the Standard Brownian Motion (SBM). The Wiener process is a building block in a lot of mathematical models, and one of the most important is the Brownian Motion (BM).

Definition 2.2. *A Brownian motion X_t, with drift (μ) and variance (σ^2), denoted $X \in BM[\mu, \sigma^2]$ is defined as*

$$X_t = \mu t + \sigma W_t$$

A useful property of a Brownian Motion is Brownian scaling

Lemma 2.1. *If W_t is a SBM then $\frac{1}{a}W_{a^2t}$ is a SBM.*

Proof. Follows from definition 2.1. □

This also implies that W_{a^2t} and aW_t are equal in distribution, a result that will prove to be very important in the scope of this thesis.

2.2 The Poisson Process

Another very important process is the Poisson process. Variations of it is included in a wide range of more advanced stock models and it's one of the building blocks for Lévy processes. It can be defined in a number of (equivalent) ways, the following is taken from Cont and Tankov [5]. The reason for choosing this definition is that a lot of the results regarding Lévy processes in this thesis is based on Cont and Tankov .

Definition 2.3. *(Poisson Process) Let $(T_i)_{i\geq 1}$ be a sequence of independent exponential random variables with mean $1/\lambda$ and $\mathbb{T}_n = \sum_{i=1}^{n} T_i$. The process $(N_t, t \geq 0)$ defined by*

$$N_t = \sum_{n\geq 1} \mathbf{1}(t \geq \mathbb{T}_n)$$

is called a Poisson process with intensity λ.

What this means is that a Poisson process is a process that counts the number of events over a time interval. The distribution of the Poisson process is given by the following theorem.

Theorem 2.1. *Let $(N_t)_{t\geq}$ be a Poisson process. Then, for any $t > 0$, N_t follows a Poisson distribution with parameter λt:*

$$\forall n \in \mathbb{N}, \mathbb{P}(N_t = n) = e^{-\lambda t}\frac{(\lambda t)^n}{n!}$$

Another important property of the Poisson process is the following.

Theorem 2.2. *Let U_1, U_2, ... , U_n be independent U(0,1)-distributed random variables, and let $U_{(1)} \leq U_{(2)} \leq ... \leq U_{(n)}$ be the ordered variables. Then*

$$\big((T_1, T_2, ..., T_n) \mid N_1 = n\big) \stackrel{d}{=} \big(U_{(1)}, U_{(2)}, ..., \leq U_{(n)}\big)$$

Proof. See Gut [6] p. 245. □

The implication of this theorem is that, given that we know the number of events on a time interval, the individual times of the events are uniformly distributed. Thus in order to simulate a Poisson process we first draw the number of events using the distribution above and then draw the individual times uniformly on the time interval (and finally sort them). The final property of a the Poisson process that will be presented is thinning.

Theorem 2.3. *Let $(N_t)_{t\geq 0}$ be a Poisson process with intensity λ. Define another process $(M_t)_{t\geq 0}$ by erasing the n:th event in N_t with probability 1 - q. This is known as thinning and M_t is then another Poisson process with intensity λq.*

2.2.1 The Compound Poisson Process

Even though the Poisson process is useful in many applications, it still has its limitations, one of them being that all jumps are of the same size. So a natural extension is to define a process which allows for jumps of different sizes. This is called the Compound Poisson Process and is defined as below.

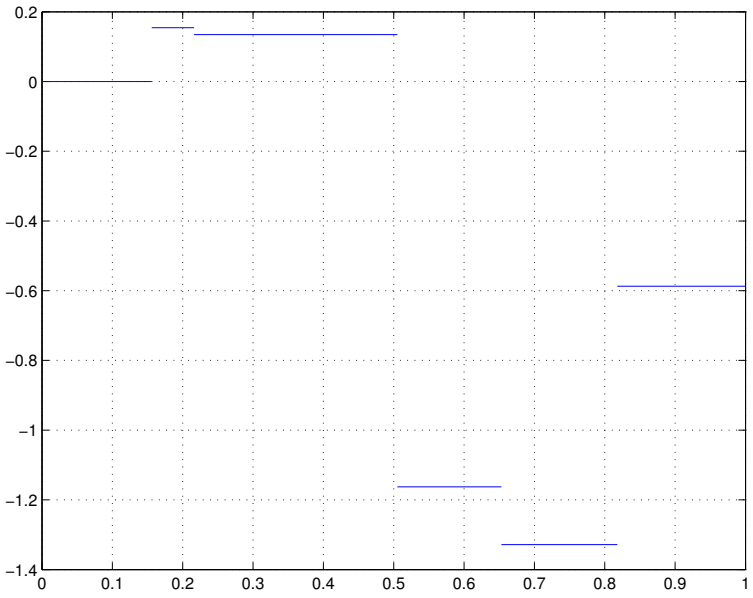


Figure 2.1: Realization of a Compound Poisson Process

Definition 2.4. *(Compound Poisson Process) A compound Poisson process with intensity $\lambda > 0$ and jump size distribution f is a stochastic process X_t defined as*

$$X_t = \sum_{i=1}^{N_t} Y_i$$

where jump sizes Y_i are i.i.d. with distribution f and (N_t) is a Poisson process with intensity λ, independent from $(Y_i)_{i\geq 1}$

Figure 2.1 shows a realization of a compound Poisson process with $\lambda = 10$, $t \in [0, T]$ and N(0,1)-distributed jumps.

2.3 Gamma Process

The third, and final, stochastic process that will appear in this thesis (chapter 7) is the Gamma process.

Definition 2.5. *A Gamma process, $G(t; \mu, v)$, is a stochastic process, satisfying the following conditions*

1. *$G(0; \mu, v) = 0$*
2. *G has independent increments*
3. *$G(t+h; \mu, v) - G(t; \mu, v)$ is gamma distributed with mean μh and variance vh, for $h > 0$*

It should be note that the Gamma process is a pure-jump process and that it's increasing. The gamma distribution is defined in the following way.

Definition 2.6. *A stochastic variable is said to be Gamma(γ, λ)-distributed if it has the following probability density function*

$$f(x; \gamma, \lambda) = \frac{1}{\Gamma(\gamma)\lambda^{\gamma}} x^{\gamma-1} e^{-\frac{x}{\lambda}}$$

with $\gamma > 0$ (shape) and $\lambda > 0$ (scale).

The gamma distribution has mean $\mu = \gamma\lambda$ and variance $v = \gamma\lambda^2$, which is equivalent to that $\gamma = \mu^2/v$ and $\lambda = v/\mu$.

Chapter 3

Exit Times - One-dimension

The purpose of this part is to introduce the simulation of exit times which is the building block of all the simulation methods in this paper. Exit times are based on the idea of simulating the time it takes for a stochastic process to reach a certain level in space, instead of simulating it via fixed time increments.

This part will deal with the one-dimensional case, whereas the next one will extend the results in this one to two or more dimensions. I will start by presenting the thought behind the approach, followed by the theory behind it, a presentation of the simulation algorithm and, finally, by applying the algorithm for pricing a barrier option in the Black-Scholes model.

3.1 Theoretical Background

As mentioned before we are interested in pricing exotic derivatives. In contrast to vanilla-options such as an ordinary (European) call-option, exotic options may be path-dependent. One example are Up-and-Out Barrier-options, which have the following payoff function:

$$\begin{aligned} \phi(S) &= \hat{\phi}(S_T)\mathbf{1}\{M_T < B\} \\ M_T &= \sup_{0\leq t\leq T} S_t \end{aligned}$$

where $\hat{\phi}$ is the payoff-function of an underlying derivative e.g. a European call-option. One way of approaching this, as presented in Adalbjörnsson and Quiroz [1], is to sample the process in space instead of, as the standard approach, in time. This means, that, instead of getting a collection of points:

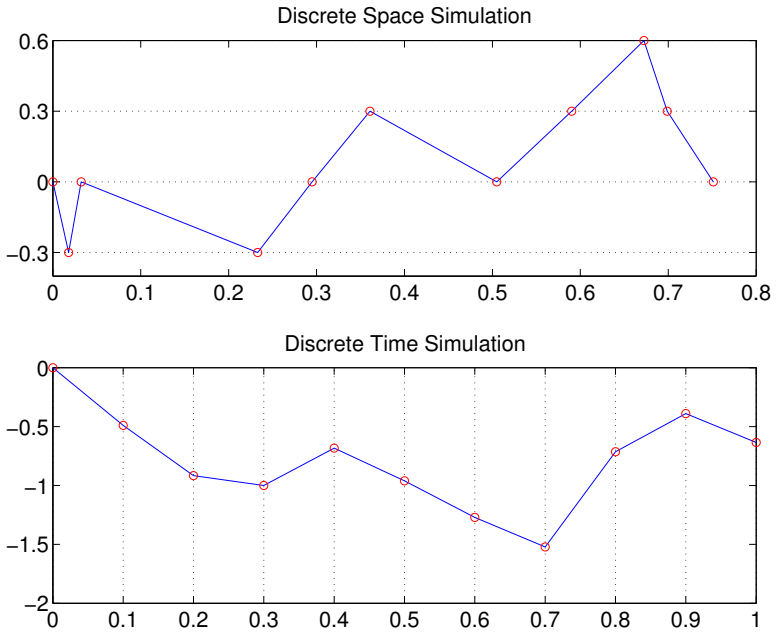
$$\{(S_t, t_i)\}_{i=1}^N \text{ such that } t_i - t_{i-1} = \Delta t$$

we get another collection of points

$$\{(S_t, \tau_i)\}_{i=1}^N \text{ such that } S_{\tau_i} - S_{\tau_{i-1}} = |\eta|$$

The advantage of this is that we will know if the stock crossed the barrier or not, in this specific time interval. In order to illustrate the difference between

the two types of discretization, simulations are shown in figure 3.1.



3.1.1 Problem statement

It can be shown that $S_{\tau_i} - S_{\tau_{i-1}}$ are i.i.d. $\forall i$ and that $(\tau_i - \tau_{i-1})$ are i.i.d. $\forall i$ (see Adalbjörnsson and Quiroz [1] pp. 29-30). This means that we only have to consider exit times as defined below.

Definition 3.1. *Let X_s be a SDE of the following form:*

$$dX_s = \mu(s, X)ds + \sigma(s, X)dW_s, X(0) = 0$$

then the exit time of X, $\tau_\eta(\mu(s, X), \sigma(s, X))$ is defined as

$$\tau_\eta(\mu(s, X), \sigma(s, X)) = \inf\{t > 0 : |X_t| \geq \eta\}$$

From now on the notation $\tau_\eta(\mu(s, X), \sigma(s, X))$ will always refer to the definition above, unless otherwise mentioned. It's worth noting that i have chosen the grid to be equidistant, the reason for this is simplicity. In this paper I will only consider the case when μ and σ are autonomous i.e. when:

$$dX_s = \mu(X)ds + \sigma(X)dW_s$$

In Milstein and Tretyakov [8] it's shown that the distribution of $\tau_\eta(\mu, \sigma)$ is equivalent to the solution of a PDE. In order to keep this paper from getting too technical/complex I will only present the results they present and use these results to derive the distribution.

Theorem 3.1. *The distribution of $\tau_\eta(\mu(X), \sigma(X))$ is given by $u(t,0)$, where $u(t,x)$ is the solution to the following boundary value problem:*

$$\begin{aligned} \frac{\partial u}{\partial t} = \frac{\sigma(x)^2}{2}\frac{\partial^2 u}{\partial x^2} + \mu(x)\frac{\partial u}{\partial x} \qquad & t > 0, x \in (-\eta, \eta) \\ u(0,x) = 0, \qquad & x \in [-\eta, \eta] \\ u(t,-\eta) = u(t,\eta) = 1, \qquad & t > 0 \end{aligned}$$

Note that $u(t,x)$ gives us the distribution of the exit time for

$$dX = \mu(X)ds + \sigma(X)dW_s, X(0) = x$$

Because of the equidistant grid and the autonomy of the parameters this is a problem that can be solved analytically relatively easy.

3.1.2 Exit Time for a SBM

But for now let's start by looking at the simplest possible example, namely the distribution of the time it takes for a SBM to exit a grid of size 1, i.e. we are interested in finding the distribution of $\tau_1(0,1)$. The result is presented in the following the theorem:

Lemma 3.1. *The distribution of $\tau_1(0,1)$, i.e. $P(\tau_1(0,1) \leq t)$, is given by one of the following two expressions:*

$$\mathcal{P}_1(t) = \quad 1 - \tfrac{4}{\pi}\textstyle\sum_{k=0}^{\infty} \frac{(-1)^k}{(2k+1)} \exp\left(-\frac{1}{8}\pi^2(2k+1)^2 t\right), \quad t > 0$$

$$\mathcal{P}_2(t) = \qquad 2\textstyle\sum_{k=0}^{\infty}(-1)^k \operatorname{erfc}\frac{2k+1}{\sqrt{2t}}, \qquad t > 0$$

where

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty \exp(-s^2)\,\mathrm{d}s$$

Proof. Inserting the values above into lemma 3.1 gives us the following system of equations:

$$\begin{aligned} \frac{\partial u}{\partial t} = \frac{1}{2}\frac{\partial^2 u}{\partial x^2} \qquad & t > 0, x \in (-\eta, \eta) \\ u(0,x) = 0, \qquad & x \in [-1, 1] \\ u(t,-1) = u(t,1) = 1, \qquad & t > 0 \end{aligned}$$

Solving this is done by first defining the function $v(t,x) = u(t,x) - 1$ and then solving the corresponding system of equations. This is done by either;

1. Using the technique Separation of Variables. This will give us P_1.
2. Extending/ Reflecting the problem to the whole axis and solving the obtained problem. This will give us P_2.

For details on solving PDE:s see e.g. Sparr and Sparr [11]. □

At a first glance this might not look like a very interesting result since it only deals with SBM for a specified grid, but by recalling the scaling property of Brownian Motions one realizes that, given a grid, η, and a constant variance σ^2.

$$\frac{\sigma}{\eta}W_t \overset{d}{=} W_{\frac{\sigma^2}{\eta^2}t}$$

and thus:

$$\tau_\eta(0,\sigma) \overset{d}{=} \frac{\eta^2}{\sigma^2}\tau_1(0,1)$$

i.e. if we want to simulate $\tau_\eta(0,\sigma)$ it's sufficient to simulate $\tau_1(0,1)$ and multiply it with $\frac{\eta^2}{\sigma^2}$.

3.1.3 Exit Time for a BM with drift

We now have the distribution of the exit time for a SBM, but in order to be able to use exit times in stock models we have to add drift. The distribution of the exit time for a BM with drift, i.e. for $\mu t + W_t$, is found using Lemma 3.1.

Lemma 3.2. *The distribution of $\tau_1(\mu,1)$, $\mathcal{P}(t;\mu)$ is given by one of the following two expressions:*

$$\begin{aligned}\mathcal{P}_1(t;\mu) = \quad & 1 - 2\pi \exp\left(-\tfrac{1}{2}\mu^2 t\right)(e^{\mu} + e^{-\mu}) \\ & \times \textstyle\sum_{k=0}^{\infty}(-1)^k \frac{(2k+1)}{\pi^2(2k+1)^2+4\mu^2} \exp\left(-\tfrac{1}{8}\pi^2(2k+1)^2 t\right)\end{aligned}$$

$$\begin{aligned}\mathcal{P}_2(t;\mu) = \quad & 1 - \tfrac{1}{2}\textstyle\sum_{k=0}^{\infty}(-1)^k e^{2\mu k}(\operatorname{erfc}\frac{2k-1+\mu t}{\sqrt{2t}} - \operatorname{erfc}\frac{2k+1+\mu t}{\sqrt{2t}}) \\ & -\tfrac{1}{2}\textstyle\sum_{k=0}^{\infty}(-1)^k e^{-2\mu k}(\operatorname{erfc}\frac{2k-1-\mu t}{\sqrt{2t}} - \operatorname{erfc}\frac{2k+1-\mu t}{\sqrt{2t}})\end{aligned}$$

Proof. The proof is very similar to that of lemma 3.1 and is therefore omitted. For the full proof see Milstein and Tretyakov pp. 761-762 [8]. □

We now know the distribution of the exit time with drift, but in order to make this really useful the ability to vary volatility and grid size is necessary. We already know that $\tau_\eta(0,\sigma) \overset{d}{=} \frac{\eta^2}{\sigma^2}\tau_1(0,1)$, but $\tau_\eta(\mu,\sigma)$ is not equal to $\frac{\eta^2}{\sigma^2}\tau_1(\mu,1)$. One solution to this, would be to find a new variable $\hat{\mu}$ such that $\tau_\eta(\mu,\sigma) \overset{d}{=} \frac{\eta^2}{\sigma^2}\tau_1(\hat{\mu},1)$. The transformation is presented in the following lemma.

Lemma 3.3. *(Transformation Lemma) Given the two Brownian Motions*

$$\begin{aligned} X_T &= \mu T + \sigma W_T \\ Y_t &= \hat{\mu} t + W_t \end{aligned}$$

we get that the following two relations hold if if $\hat{\mu} = \frac{\eta}{\sigma^2}\mu$.

1. $X_T \overset{d}{=} \eta Y_t$, *with* $T = \frac{\eta^2}{\sigma^2}t$
2. $\tau_\eta(\mu,\sigma) \overset{d}{=} \frac{\eta^2}{\sigma^2}\tau_1(\hat{\mu},1)$

Proof. By using Y as the starting point we get

$$\begin{aligned} Y_t &= \hat{\mu}t + W_t = \mu\frac{\sigma^2}{\eta}t + W_t = \mu\frac{\eta}{\sigma^2}\frac{\sigma^2}{\eta^2}\frac{\eta^2}{\sigma^2}t + \frac{\sigma}{\eta}W_{\frac{\eta^2}{\sigma^2}t} \\ &= \frac{\mu}{\eta}T + \frac{\sigma}{\eta}W_T = \frac{1}{\eta}X_T \end{aligned}$$

The first equality comes from the definition of Y_t, the second from inserting the expression for $\hat{\mu}$, the third from multiplying with $\frac{\eta^2}{\sigma^2}\frac{\sigma^2}{\eta^2} = 1$ everywhere and exploiting the scaling property of the Wiener process, and finally, the last equality comes from replacing t with T.

Using this, we get that the time that it takes for $\frac{1}{\eta}X_T$ to exit a grid of size 1 is equal to the time it takes Y_t to exit a grid of size 1 times $\frac{\eta^2}{\sigma^2}$ and, this is then obviously the same as the time it takes X_T to exit a grid of size η. □

Because of the results above we can conclude that in order to simulate exit times and BMs we only need to consider BMs on the form

$$X_t = \mu t + W_t, X_0 = 0$$

all others can be handled as transformations of this.

3.1.4 Distribution of Exit Points

We now have the distribution for the exit time of BM with drift from an equidistant grid, and we are now interested in finding the probabilities that the BM went up or down. Let's denote all exit times by τ in this section. If $\mu = 0$ then obviously P("up") = P("down"), since the scaling property of a SBM implies that

$$-\sigma W_\tau \stackrel{d}{=} \sigma W_{\frac{1}{(-1)^2}\tau} = \sigma W_\tau$$

Adding drift renders the problem slightly more difficult, but with the help of the following theorem we can deal with it.

Theorem 3.2. *(Reuters Theorem) First, let $X_t = \mu t + W_t$ with $X_0 = 0$. Second, let $\tau = \inf\{t > 0 : |X_t| \geq 1\}$. Then X_τ and τ are independent.*

Proof. See Rogers and Williams p. 84 [9]. □

We now have everything that is needed to find the probabilities, the results are presented in the following theorem.

Theorem 3.3. *First, let $X_t = \mu t + W_t$ with $X_0 = 0$. Second, let $\tau = \inf\{t > 0 : |X_t| \geq 1\}$. Then the probabilities that the BM went up or down are as follows.*

$$\begin{aligned} P(X_\tau = -1|\tau = \theta) &= \frac{1}{e^{2\mu} + 1} \\ P(X_\tau = 1|\tau = \theta) &= \frac{e^{2\mu}}{e^{2\mu} + 1} \end{aligned}$$

Proof. Since the two events are complementary we only have to show one of the cases e.g. the first one. Using theorem 3.2 we get that

$$P(X_\tau = -1|\tau = \theta) = P(X_\tau = -1)$$

We also know, by definition, that $X_\tau = |1|$ and thus we get

$$P(X_\tau = -1) = P(X_t = -1|X_t = |1|) = \frac{P(X_t = -1)}{P(X_t = -1) + P(X_t = 1)}$$

Inserting the distribution for a BM for an arbitrary time t gives us the results of the theorem. □

3.2 Simulation

3.2.1 One-Step Simulation

With all the probabilities in place we have everything needed to simulate a stochastic time step for a BM. But before describing the simulation algorithm let's take some time to discuss the implementation.

Implementation

In order to simulate we need to draw samples from $\mathcal{P}(t;\mu)$. One method of doing this is through inversion sampling (see appendix A). Since the probabilities in this case consists of infinite sums it is obviously impossible to find a analytical expression for the inverse of $\mathcal{P}(t;\mu)$. Thus we have to apply numerical methods to obtain the inverse. On p.33 in Adalbjörnsson and Quiroz [1] it's shown that the density function of $\tau_\eta(\mu,\sigma)$ can be written as (density functions will be denoted as $\hat{p}$).

$$\begin{aligned}
\hat{p}_\tau(t;\mu,\sigma,\eta) &= \frac{1}{2}\left(\exp(-\frac{\mu^2}{2\sigma^2}t)\left(\exp(-\frac{\mu\eta}{\sigma^2}) + \exp(\frac{\mu\eta}{\sigma^2})\right)\right)\hat{p}_\tau(t;0,\sigma,\eta) \\
&= \exp(-\frac{\mu^2}{2\sigma^2}t)\cosh(\frac{\mu\eta}{\sigma^2})\hat{p}_\tau(t;0,\sigma,\eta) \\
&= \exp(-\frac{\mu^2}{2\sigma^2}t)C\hat{p}_\tau(t;0,\sigma,\eta)
\end{aligned}$$

Since $\exp(-\frac{\mu^2}{2\sigma^2}t) < 1$ this is an excellent candidate for using acceptance-rejection sampling, with $\hat{p}_\tau(t;0,\sigma,\eta)$ as instrumental distribution. Thus we only need to draw samples from $\hat{p}_\tau(t;0,\sigma,\eta)$, but we have shown that $\tau_\eta(0,\sigma) \overset{d}{=} \frac{\eta^2}{\sigma^2}\tau_1(0,1)$ and thus we only have to sample from $\hat{p}_\tau(t;0,1,1)$, i.e. from $\mathcal{P}(t)$. In my implementation I have used a numerical approximation presented in Adalbjörnsson and Quiroz [1] in order to find the inverse of $\mathcal{P}$.

Euler Scheme

The way of simulating as described above is for the case when the drift and volatility are constant. In some stock models it's not possible to write the model on such a form so we need a way of dealing with that. The solution is the Euler scheme. The thought behind the Euler scheme is that, given that we have a model on the following form

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t$$

we approximate it with the following

$$\begin{aligned} X_{t_{i+1}} &= X_{t_i} + \mu(t_i, X_{t_i})(t_{i+1} - t_i) + \sigma(t_i, X_{t_i})(W_{t_{i+1}} - W_{t_i}) \\ &= X_{t_i} + \mu(t_i, X_{t_i})(t_{i+1} - t_i) + \sigma(t_i, X_{t_i})\sqrt{t_{i+1} - t_i}Z \end{aligned}$$

Where Z is a standard normal variable. This means that the drift and volatility are kept constant over an interval. It can be showed that the Euler scheme converges both strongly and weakly, for more on this see Adalbjörnsson and Quiroz p. 12 [1], i.e. the approximation gets better as $((t_{i+1} - t_i) \to 0$. In [1] it's also shown that the mean number of steps, $E[N_T]$ our algorithm takes for a BM with drift can be approximated by the following expression.

$$E[N_t] \approx \frac{\sigma^2 T}{\eta^2}$$

This means that the mean step length, $E[L_T]$, can be approximated by the following

$$E[L_T] = \frac{T}{E[N_T]} \approx \frac{\eta^2}{\sigma^2}$$

Which implies that as σ goes down the algorithm will take longer steps (on average). This imposes a problem when using the Euler scheme, if we lock the volatility when it's low, the algorithm will believe it can take long steps, but in reality σ might change fast and thus bias is introduced.

Simulation Algorithm

The algorithm for simulating one step for a pair of points

$$\{(X_\tau, \tau)\} = \{(X_t \pm \eta, t + \tau)\}$$

given $\{(X_t, t)\}$ is the following

Algorithm 1 DISCRETE SPACE ONE-STEP SIMULATION

repeat
  $\tau \leftarrow$ sample from $\hat{p}(t; 0, 1, 1)$
  $\tau \leftarrow \frac{\eta^2}{\sigma^2}\tau$
  U $\leftarrow$ sample from Uniform(0,1)-distribution
until $U < \exp(-\frac{\mu^2}{2\sigma^2}\tau)$
U $\leftarrow$ sample from Uniform(0,1)-distribution
$Y \leftarrow \frac{1}{\exp(2\mu)+1}$
if U $<$ Y **then**
  $X \leftarrow X - \eta$
else
  $X \leftarrow X + \eta$
end if
return (τ,X)

3.2.2 Simulation of a Path

We now have a way of simulating a collection of points. But since this only gives us stochastic time points and a Barrier Option has a deterministic end point, i.e. the Time To Maturity (TTM), we will need a way of dealing with this. Throughout this section results will only be presented for BMs on the form, $X_t = \mu t + W_t$, and grid sizes, $\eta = 1$, as showed in Lemma 3.3 all other cases can be dealt with through transformations.

Theoretical Results

In order to cope with the final point let's introduce the following stochastic variable

$$\hat{\tau} = \min(\tau, T - t) = \min(\tau, l)$$

where l denotes the time left until the TTM. Since l is deterministic this equals a truncated version of τ i.e.

$$P(\hat{\tau} \leq t) = \begin{cases} \mathcal{P}(t), & \text{if } t \leq l \\ 1, & \text{if } t > l \end{cases}$$

If $\hat{\tau} = l$, this equals the event that the BM did not hit the grid in the specified time. This can be described mathematically as

$$(\hat{\tau} = l) \Leftrightarrow (\tau \geq l) \Leftrightarrow (|X_s| < \eta, 0 < s < l)$$

Since τ has a continuous distribution the probability that $\tau = l$ will be zero and thus we must find the distribution of X_t given that it has not hit the barrier. This a distribution that will be used throughout this thesis and because of that let's define it properly.

Definition 3.2. *The distribution, $\mathcal{L}$, is defined as follows*

$$\mathcal{L}(\beta; t, \mu) := P(X_t < \beta | \tau \geq t) = \frac{P(X_t < \beta, \tau \geq t)}{P(\tau \geq t)}$$

Since we know the denominator the only thing we need to find is $P(X_t < \beta, \tau \geq t)$. The first towards finding this is given in the following theorem.

Theorem 3.4. *$\mathcal{L}(\beta; t, \mu)$ is given as $u(t, 0)$ where $u(t, x)$ is the solution to the following foundry value problem*

$$\begin{aligned} \frac{\partial u}{\partial t} = \frac{1}{2}\frac{\partial^2 u}{\partial x^2} + \mu\frac{\partial u}{\partial x} \qquad & t > 0, x \in (-\eta, \eta) \\ u(0, x) = 0, \qquad & x \in [-\eta, \eta] \\ u(t, -1) = u(t, 1) = 0, \qquad & t > 0 \end{aligned}$$

Proof. See p. 764 in Milstein and Tretyakov 764 [8]. □

As a result of this theorem we now have a way of finding analytical expressions for $\mathcal{L}(\beta; t, \mu)$.

Theorem 3.5. *$\mathcal{L}(\beta; t, \mu)$ is given by one of the following two expressions.*

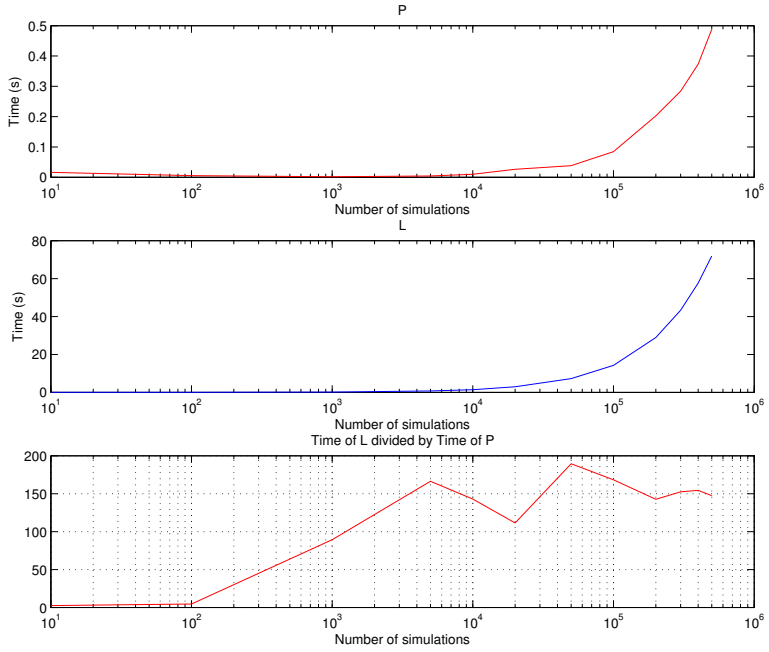
$$\begin{aligned} \mathcal{L}_1(\beta; t, \mu) &= \frac{4}{1 - \mathcal{P}(t; \mu)} \exp(-\mu^2 t/2) \sum_{k=0}^{\infty} \frac{1}{\pi^2(2k+1)^2 + 4\mu^2} \\ &\quad \times \Bigg((-1)^k \frac{\pi(2k+1)}{2} \exp(-\mu) \\ &\quad + \exp(\mu\beta) \left[\mu \cos\frac{\pi(2k+1)\beta}{2} + \frac{\pi(2k+1)}{2} \sin\frac{\pi(2k+1)\beta}{2} \right] \Bigg) \\ &\quad \times \exp\left(-\frac{1}{8}\pi^2(2k+1)^2 t \right) \\ \mathcal{L}_2(\beta; t, \mu) &= \frac{1}{2(1 - \mathcal{P}(t; \mu))} \\ &\quad \times \sum_{k=0}^{\infty} \Bigg(\exp(4\mu k) \left[\operatorname{erfc}\frac{4k - \beta + \mu t}{\sqrt{2t}} - \operatorname{erfc}\frac{4k + \beta + \mu t}{\sqrt{2t}} \right] \Bigg) \\ &\quad \left[\operatorname{erfc}\frac{4k + 3 + \mu t}{\sqrt{2t}} - \operatorname{erfc}\frac{4k + 2 - \beta + \mu t}{\sqrt{2t}} \right] \end{aligned}$$

3.2.3 Implementation

We now have all the probabilities that we need to implement the simulation of one path for the BM $X_t = \mu t + \sigma W_t$ on the time interval $t \in [0, T]$. In order to simulate from $\mathcal{L}$ we will use inversion sampling. Unfortunately we, again, have the problem that the distribution of $\mathcal{L}$ consists of infinite sums and we haven't been able to find an analytical inverse for it. In the case with $\mathcal{P}$ we managed to deal with this with the help of acceptance-rejection sampling. In this case we haven't been able to find an instrumental distribution and the only option left is to make the inversion numerically. As the method for doing this we have chosen the binary search, for more information on the binary search see appendix A.

This method is obviously a lot slower than using an analytical expression, but because terms in the sums in $\mathcal{L}$ tends to zero very fast we can truncate the sums,

and thus make the algorithm a lot faster without losing much in accuracy. It's still a lot slower than the numerical approximation used for $\mathcal{P}$. As will be seen later in the thesis $\mathcal{L}$ can be used for more things than just simulating the end points of a path, and $\mathcal{P}$ and $\mathcal{L}$ will prove to be the two main parts which the simulation method is built around for Lévy processes. Therefore a simulation study was performed, comparing the two distributions. The study was performed by comparing the times it took to sample from the distributions for different number of simulations. The results are shown below and as can be seen $\mathcal{L}$ was about a hundred times slower than $\mathcal{P}$. In matlab the symbols $\mathcal{L}$ and $\mathcal{P}$ are not available, therefore, L means $\mathcal{L}$ and P means $\mathcal{P}$ in the plot.



3.2.4 Simulation Algorithm

The resulting algorithm is the following.

Algorithm 2 DISCRETE SPACE SIMULATION

$i \leftarrow 1$
while $\tau_{i-1} < T$ **do**
  $(\tau_i, X_{\tau_i}) \leftarrow$ simulate using Algorithm 1
  if $\tau_i \geq T - \tau_{i-1}$ **then**
    $\tau_i \leftarrow$ T
    $X_{\tau_i} \leftarrow$ sample from $\mathcal{L}$
  end if
  $i \leftarrow i + 1$
end while
return $(\bar{\tau}, \bar{X})$

3.2.5 Discrete Space Monte Carlo (DSMC)

Using the algorithm above we now have a way of simulating a path for $X_t = \mu t + W_t, t \in [0, T]$ giving us a sequence of points

$$\{(X_{\tau_i}, \tau_i)\}_{i=1}^{N} \text{ such that } S_t \in [S_{\tau_{i-1}} - \eta, S_{\tau_{i-1}} + \eta], t \in [\tau_{i-1}, \tau_i]$$

Using this we will now implement the simulation algorithm to get Monte Carlo estimates, in accordance with Adalbjörnsson and Quiroz [1] this will be called Discrete Space Monte Carlo. The algorithm is the following

Algorithm 3 DISCRETE SPACE Monte Carlo

1: $i \leftarrow 1$
2: **for** $i = 1 : N$ **do**
3: $(\tau_i, X_{\tau_i}) \leftarrow$ simulate using Algorithm 2
4: $\phi_i \leftarrow \phi_i(X_{\tau_i})$
5: **end for**
6: $\bar{\phi} \leftarrow \frac{1}{N} \sum_{k=0}^{N} \phi_k$
7: **return** $(\bar{\phi})$

3.3 Case Study: Black-Scholes Model

With all the theory in place it's now time to verify that the algorithm actually works. In order to do this we will test it for different types of options in the Black-Scholes model. The log-price in the B-S-model is given by the following expression.

$$Z_t = \log(S_t) = (r - \frac{\sigma^2}{2})t + \sigma W_t, Z_0 = \log(S_0)$$

Since this has constant drift and volatility we can simulate the exit times exactly, regardless of the size of the grid. In order to show this we start by pricing a Single Barrier Up-and-Out call option with the following set of parameters. Since it's possible to calculate the analytical price for Single Barrier call-options in the B-S-model this has been used as reference.

Variables	Abbreviation	Value
Number of Simulations	N	100000
Stock value today	S_0	50
Upper barrier	U	60
Strike price	K	50
Time to maturity	T	1
Risk-free rate	r	0.1
Volatility	σ	0.1

The results are shown in Figure 3.1. As seen, the results are indifferent to the size of the grid, thus when faced with a process constant drift and volatility we

can set the grid equal to the distance to the barrier, without loss of precision. In order to show how the Euler scheme effects the process we look at a Double Barrier call-option with the same set of parameters and a lower barrier of 40 and price it using the ordinary Black-Scholes model. The Black-Scholes model is given by the following expression.

$$dS_t = rS_t dt + \sigma S_t dW_t$$

Discretisizing this using the Euler scheme yields.

$$\begin{aligned} S_{t_{i+1}} &= S_{t_i} + rS_{t_i}(t_{i+1} - t_i) + \sigma S_{t_i}(W_{t_{i+1}} - W_{t_i}) \\ &= S_{t_i} + rS_{t_i}(t_{i+1} - t_i) + \sigma S_{t_i}\sqrt{t_{i+1} - t_i}Z \end{aligned}$$

A Double Barrier option has the following payoff function.

$$\begin{aligned} \phi(S) &= \hat{\phi}(S_T)\mathbf{1}\{M_T < B_U\}\mathbf{1}\{m_T > B_L\} \\ M_T &= \sup_{0\leq t\leq T} S_t \\ m_T &= \inf_{0\leq t\leq T} S_t \end{aligned}$$

Where B_U denotes the upper barrier and B_L the lower barrier. Figure 3.2 shows the convergence of the price, as a reference point the same algorithm, but for the log-price was used. We know that the expected number of steps is proportional to η^{-2}, see Figure 3.3 for a plot of the times for different η, which means that the time our algorithm takes should also be proportional to η^{-2}. It would therefore be interesting to find a way to find a relation between η and the bias the Euler discretization results in as a result of η. Unfortunately I haven't been able to work this out.

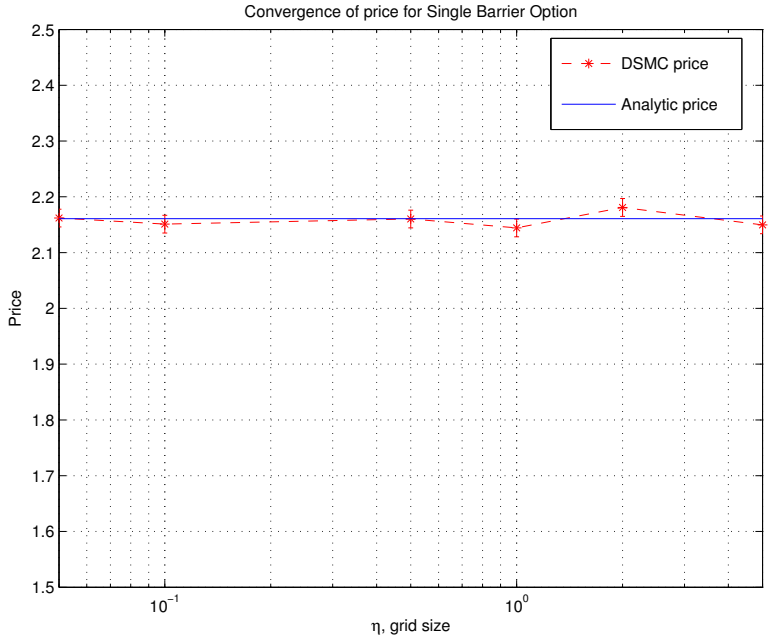


Figure 3.1: Convergence of Single Barrier Option Price as a function of η

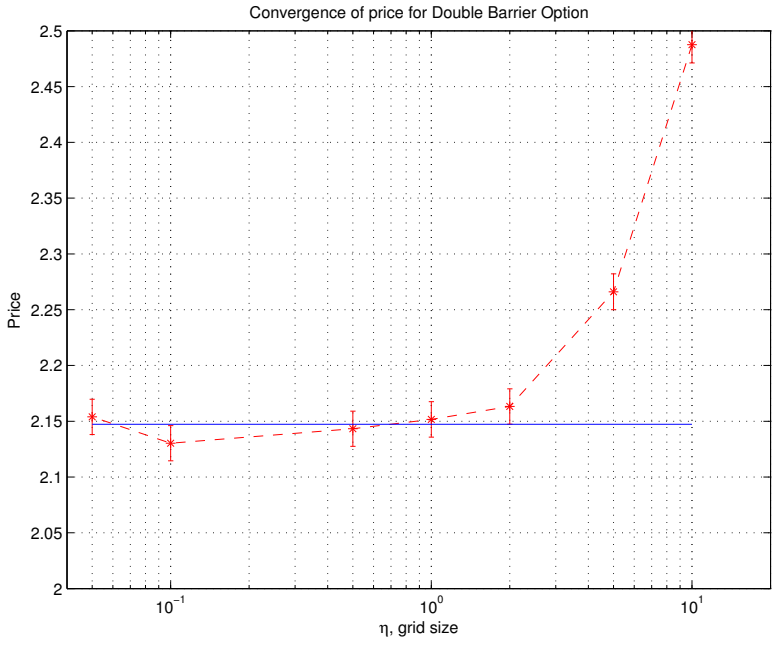


Figure 3.2: Convergence of Double Barrier Option Price as a function of η

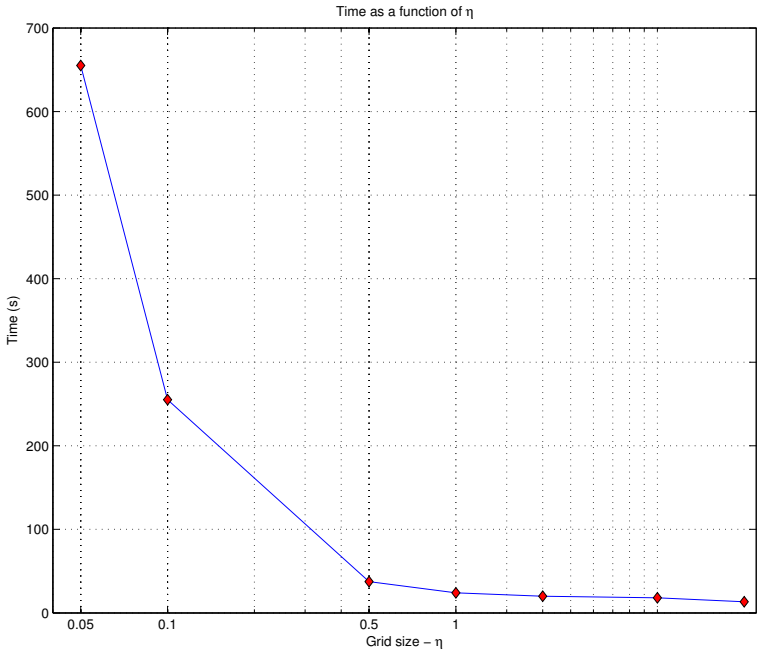


Figure 3.3: Time as a function of η

Chapter 4

Multi-dimensional Exit Times

In the previous chapter we dealt with the pricing of Barrier Options in the Black-Scholes model, in this chapter we will extend the results from the previous one to the multidimensional case. This will allow us to price the options under more complex, and useful, stock models such as the Heston Model. The chapter starts with presenting the theory behind the multidimensional simulation method. The method is then presented and finally a simulation study is performed.

4.1 Theory

Let's start by restating the problem in the multidimensional case. We are dealing with BMs on the form.

$$d\bar{X}_t = \mu dt + \sigma dW_t, X(0) = 0$$

Where $\bar{X}$ is a N-dimensional BM, μ is a N-dimensional column vector, σ is a matrix of dimension $N \times N$ and W_t is a N-dimensional SBM. Thus our problem consists of finding the exit time, as given below.

$$\tau_\eta(\mu(s, X), \sigma(s, X)) = \inf\{t > 0 : |X_t| \geq \bar{\eta}\}$$

Where η is also a N-dimensional column vector and $|X_t|$ denotes the absolute value of each element in X_t. This means that we are interested in finding the first time that any of the processes in X exits it's corresponding grid. In order to do this we will first see how this is done in the case when all elements of X are independent of each other and then use the results obtained there to extend it to all cases.

4.2 Independent Case

In the independent case the sigma-matrix is obviously a diagonal matrix and thus each element of X can be written in the following way

$$d\bar{X}_t^i = \mu^i dt + \sigma^{i,i} dW_t, X^i(0) = 0$$

Since all elements are independent that also means that they have individual and independent exit times, i.e.

$$\tau^i_{\eta^i}(\mu^i(s,X),\sigma^{i,i}(s,X)) = \inf\{t>0 : |X^i_t| \geq \eta^i\}$$

This also tells us the following, with $\tau^i = \tau^i_{\eta^i}(\mu^i(s,X),\sigma^{i,i}(s,X))$.

$$\tau = \min(\tau^1,\tau^2,\ldots,\tau^{N-1},\tau^N)$$

Thus in order to simulate τ we simulate each of the τ^i independently and choose the lowest one. If we know that one element has hit the grid we now only need to know the probability of whether it went up or down, and this is obviously known from the previous chapter. At time τ we also know that the other ones haven't hit the grid, we are interested in finding the distribution of their placement given that they haven't hit the grid, but this is the same as $\mathcal{L}$, and thus we have everything needed to simulate one step of the process. Extending this to a whole path is done by adding an end time (i.e. Time to maturity) in the same way as in the one-dimensional case.

4.2.1 Simulation Algorithm

Given the results above the simulation algorithm, for a process with N-elements, is as follows

Algorithm 4 MULTI-DIMENSIONAL DISCRETE SPACE SIMULATION

$i \leftarrow 0$
while $\tau_i < T$ **do**
  $i \leftarrow i+1$
  for j = 1:N **do**
    $(t_j, X_{t_j}) \leftarrow$ simulate using Algorithm 1
  end for
  $\tau_i \leftarrow$ take the minimum of $\bar{t}$ and $T - \tau_{i-1}$
  for j = 1:N **do**
    if $t_j! = \tau_i$ **then**
      $X_{\tau_i} \leftarrow$ sample from $\mathcal{L}$
    end if
  end for
end while
return $(\bar{\tau}, \bar{X})$

4.3 Multidimensional Simulation

We now have an algorithm for simulating a path for the case when the elements in X are independent, but in e.g. the Heston-model there is a dependence between the two processes and we are thus interested in finding a way of simulating this.

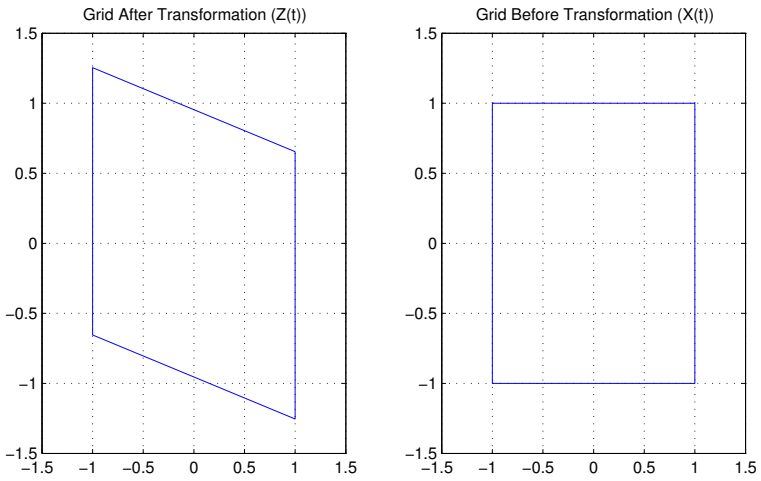


Figure 4.1: An illustration of the grids before and after the transformation

In this thesis the way of doing this will be to try to find a transformation such that we get a process with independent elements and then use the algorithm above to simulate our step before finally transforming it back.

Our original problem is to find the time such that

$$\bar{X}_t = \mu t + \Sigma W_t \text{ exits } \eta$$

assuming that Σ is invertible, this is obviously the same as finding the time such that

$$\bar{X}_t = \Sigma\Sigma^{-1}\mu t + \Sigma W_t \text{ exits } \eta$$

by setting $U = \Sigma^{-1}\mu$ and multiplying with Σ^{-1} everywhere we get that this is as the same as finding the time such that

$$\bar{Z}_t = \Sigma^{-1}\bar{X}_t = Ut + W_t \text{ exits } \Sigma^{-1}\eta$$

But Z_t obviously has independent elements and we thus know how to simulate it's exit times, as long as $\Sigma^{-1}\eta$ is equidistant in each dimension. Thus we set $\bar{\eta} = \Sigma^{-1}\eta$ to be equidistant, this means that we will get the time it takes for X to exit a transformed grid. Figure 4.1 shows an illustration of what this might look like in two dimensions.

4.4 Case Study: Heston-model

With the algorithm in place we test it for a multidimensional stock model, namely the Heston model. The Heston model is given by the following expressions.

$$\begin{aligned} dS_t &= rS_t dt + \sqrt{v_t} S_t dW_t^1 \\ dv_t &= \kappa(\theta - v_t)dt + \xi\sqrt{v_t}(\rho dW_t^1 + \sqrt{1-\rho^2}dW_t^2) \end{aligned}$$

where S denotes the stock price and v the volatility of the stock. In order to simulate this, the process must be discretized. By once again using the Euler discretization and using the log-transform on the stock process, in order to remove bias in the stock price direction, we get

$$\begin{aligned} \log(S_{n+1}) &= \log(S_n) + (r - \frac{v_n}{2})\Delta t + \sqrt{v_n} W_{\Delta t}^1 \\ v_{n+1} &= |v_n + \kappa(\theta - v_n)\Delta t + \xi\sqrt{v_n}(\rho W_{\Delta t}^1 + \sqrt{1-\rho^2}W_{\Delta t}^2)| \end{aligned}$$

The absolute sign on the volatility process is there to keep the volatility process from becoming negative. There are a number of ways of doing this and this has been chosen because of the simplicity to implement it.

4.4.1 Double Barrier Option - Case I

In this section a Double Barrier call-option is priced using the algorithm. The main object of this part is to test if the algorithm and how long it takes. As a reference point a PDE-solver based on the Crank-Nicolson method was used. The simulation was performed by setting η equal for the stock- and volatility-process and then varying it. The parameters used were the following.

Variables	Abbreviation	Value
Number of Simulations	N	100000
Stock value today	S_0	50
Volatility value today	V_0	0.16
Upper barrier	U	60
Lower barrier	L	40
Strike price	K	50
Time to maturity	T	1
Risk-free rate	r	0.1
Volatility of volatility	ξ	0.1
Mean reversion rate	κ	1
Mean reversion level	θ	0.1
BM correlation	ρ	-0.3

Figure 4.2 shows the results of the simulation, as seen the algorithm converges to the right price. Another point of interest is how long the algorithm takes for different η, this is shown in Figure 4.3, as a reference the PDE-solver took 22 seconds. The mean number of steps is, as we know, proportional against $1/\eta^2$ and at (almost) every step we have to sample from both $\mathcal{L}$ and $\mathcal{P}$. Since sampling from $\mathcal{L}$ is fairly slow it would be of interest to keep η as high as possible.

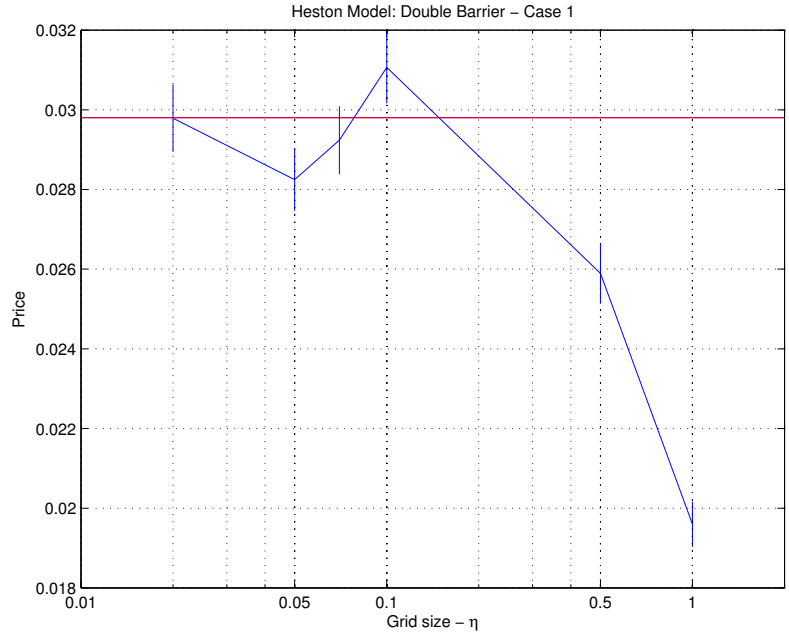


Figure 4.2: Convergence of Double Barrier Option Price as a function of η

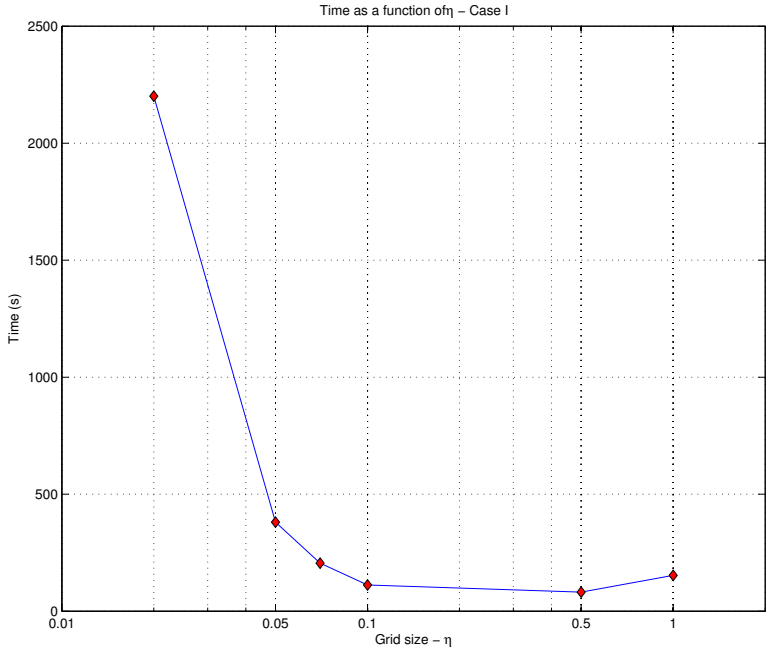


Figure 4.3: Time as a function of η - Case I

4.4.2 Double Barrier Option - Case II

For the Heston-model there exists something called the Feller condition, it states that if the following inequality holds

$$2\kappa\theta \geq \xi^2$$

then the volatility process is always positive. If the condition doesn't hold the volatility can become zero. The implication of this in our model would be that the process assumes that it can take very long steps and thus bias is introduced. In order to see the effect of this a simulation with the following set of parameters was performed.

Variables	Abbreviation	Value
Number of Simulations	N	100000
Stock value today	S_0	100
Volatility value today	V_0	0.09
Upper barrier	U	120
Lower barrier	L	80
Strike price	K	100
Time to maturity	T	1
Risk-free rate	r	0.05
Volatility of volatility	ξ	1
Mean reversion rate	κ	1
Mean reversion level	θ	0.09
BM correlation	ρ	-0.3

The results are shown below. As seen the price did not converge for the grid sizes considered. It could be argued that the prices seems to be converging to the right value, but simulating for $\eta = 0.002$ took an unreasonable amount of time (almost 40 minutes). As suggested above one reason for the slow convergence rates might be that the process believes it can take very long steps, thus introducing bias. I tried dealing with this limiting the length of the time steps but it didn't really help.

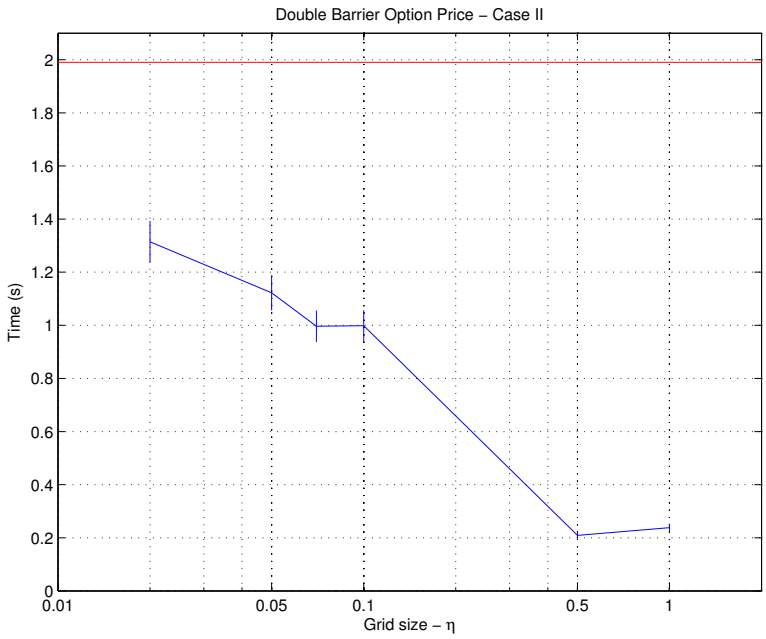


Figure 4.4: Convergence of Double Barrier Option Price as a function of η

Chapter 5

Lévy Processes

In this section Lévy processes are introduced. First the Lévy process is defined and an explanation of the definition is provided. Then the Lévy-Itô decomposition is introduced, it's then explained how it can be used to characterize a Lévy process. Finally some theorem, building on the Lévy-Itô decomposition are presented .

5.1 Definition

Lévy processes are a type of stochastic processes used as building blocks in a number of modern stock models. In order to define the Lévy process i will first define the characteristics that make up it's definition. The first one is that it has independent increments.

Definition 5.1. *(Independent Increments) A stochastic process is said to have independent increments if:* $X(t_0)$, $X(t_1) - X(t_0)$, ... , $X(t_n) - X(t_{n-1})$ *are independent for every choice of times such that* $t_0 \leq t_1 \leq ... \leq t_n$.

The second characteristic of the Lévy process is that it has stationary increments.

Definition 5.2. *(Stationary Increments) A stochastic process is said to have stationary increments if: the distribution of* $X_{t+h} - X_t$ *doesn't depend of t.*

The third characteristic of the Lévy process is that it has stochastic continunity.

Definition 5.3. *(Stochastic Continuity) A stochastic process is said to have stochastic continuity if:* $\forall \varepsilon$, $\lim_{h \to 0} \Pr(|X_{t+h} - X_t| \geq \varepsilon) = 0$.

Please note that this is not the same as continuous sample paths, the purpose of the definition is to exclude jumps at fixed times. The fourth, and final, characteristics of a Lévy process is that it's cadlag.

Definition 5.4. *(Cadlag Function) A function f:* $[0,T] \to \mathbb{R}^d$ *is said to be cadlag if: it's right-contiuous with left limits, i.e. for each* $t \in [0,T]$ *the limits*

$$f(t-) = lim_{s \leftarrow t, s<t} f(s) \qquad f(t+) = lim_{s \leftarrow t, s>t} f(s)$$

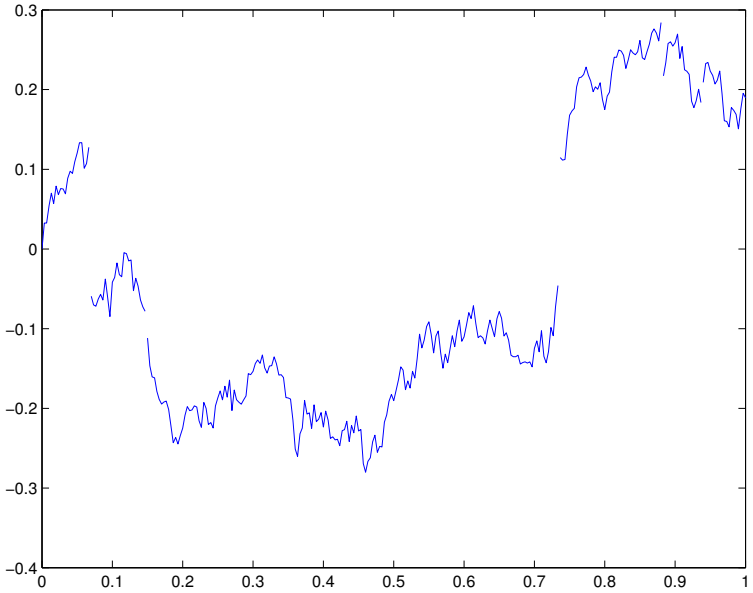
exist and $f(t) = f(t+)$

It's worth noting here that all continuous functions are cadlag. With all definitions in place I will now define the Lévy process.

Definition 5.5. *(Lévy Process) A* $\underline{\text{cadlag}}$ *stochastic process* $(X_t)_{t \geq 0}$ *on* $(\omega, \mathcal{F}, P)$ *with values in* $\mathbb{R}^d$*, such that* $X_0 = 0$ *(a.s.) is called a Lévy process if it has the following properties:*

1. *Independent Increments*
2. *Stationary Increments*
3. *Stochastic Continuity*

One example of a Lévy process is the Wiener Process, it has stationary and independent increments by definition and the fact that it has continuous sample paths implies that it's both cadlag and that it has stochastic continuity. Below a realization of another Lévy process is shown.



In this thesis the purpose of including Lévy processes is to use them to define stock models. A lot of stock models are defined for the log-price of the stock and they are thus defined on the following form

$$X_t = X_0 e^{Z_t}$$

This implies that in order to get a fair price we must be able to find the exponential moments of the process. This is done using the following two theorem from Cont and Tankov [5] (here, dots marks the inner product).

Theorem 5.1. *(Characteristic function of a Lévy process) Let $(X_t)_{t\geq 0}$ be a Lévy process on $\mathbb{R}^d$. There exist a continuous function ψ: $\mathbb{R}^d \to \mathbb{R}$ called the characteristic exponent exponent of X, such that:*

$$E[e^{iz.X_t}] = e^{t\psi}, z \in \mathbb{R}^d.$$

Theorem 5.2. *(Exponential Moments of a Lévy process) Let $(X_t)_{t\geq 0}$ be a Lévy process on $\mathbb{R}$ with characteristic triplet (A, ν, γ) and let $u \in \mathbb{R}$. The exponential moment $E[e^{uX_t}] = e^{t\psi(-iu)}$ is finite for all $t > 0$ if and only if $\int_{|x|\geq 1} e^{ux}\nu(dx) < \infty$. In this case the exponential moment $\Lambda(u)$ is given by the following relation*

$$E[e^{uX_t}] = e^{t\psi(-iu)} = e^{t\Lambda(u)}$$

where ψ is the characteristic exponent.

5.2 Characterisation of Lévy processes

We now have a definition of Lévy processes, but this definition is very wide and perhaps a bit hard to grasp. In order to make things a bit clearer let's start by considering one of the most famous Lévy processes, namely the compound Poisson Process.

As seen in chapter 2 the compound Poisson process is a generalization of the Poisson process, where, instead of just having jumps of size 1, we have jumps of different sizes. In the standard Poisson process the intensity is a measure of how often jumps occur and it would be interesting to find an equivalent measure for the compound process. The difference would of course be that this measure should show us how often jumps of a certain size occur, instead of just how often jumps occur. An alternate formulation would be that we are interested in the average number of jumps, of a certain size, per time unit. Finding this is something that is interesting for all Lévy process, not just for compound Poisson processes, and it's called the Lévy measure. The definition is found below.

Definition 5.6. *(Lévy Measure) Let $(X_t)_{t\geq 0}$ be a Lévy process defined on $\mathbb{R}^d$. The measure ν on $\mathbb{R}^d$ is defined as:*

$$\nu(A) = E[\#\{t \in [0,1] \,:\, \Delta X_t \neq 0,\ \Delta X_t \in A\}],\ A \in \mathcal{B}(\mathbb{R}^d)$$

is called the Lévy measure of X: $\nu(A)$ is the expected number, per unit time, of jumps whose size belong to A.

$\mathcal{B}$ denotes the Borel σ-algebra, the important implication of this (for the scope of this thesis) is that A can be any open or closed subset of $\mathbb{R}^d$. For more details on Borel σ-algebras, see Chapter 2 in Cont and Tankov . It turns out that the Lévy measure is a crucial part in characterizing a Lévy process and the reason why is given in the following theorem, called the Lévy-Itô decomposition.

Theorem 5.3. *Let $(X_t)_{t\geq 0}$ be a Lévy process defined on $\mathbb{R}^d$ and ν it's Lévy measure.*

- *ν is a Radon measure on $\mathbb{R}\setminus\{0\}$ and verifies:*

$$\int_{|x|\leq 1}|x|^2\nu(dx)<\infty \qquad \int_{|x|\geq 1}\nu(dx)<\infty.$$

- *The jump measure of X, denoted by J_x, is a Poisson random measure on $[0,\infty[\times\mathbb{R}^d$ with intensity measure $\nu(dx)dt$.*

- *There exist a vector γ and a d-dimensional Brownian motion $(W_t)_{t\geq 0}$ with covariance matrix A such that*

$$\begin{aligned} X_t &= \gamma t + W_t + X_t^l + \lim_{\varepsilon\downarrow 0}\tilde{X}_t^\varepsilon, \\ X_t^l &= \int_{|x|\geq 1, s\in[0,t]} xJ_x(ds\times dx) \\ \tilde{X}_t^\varepsilon &= \int_{\varepsilon\leq|x|\leq 1, s\in[0,t]} x\{J_x(ds\times dx)-\nu(dx)ds\} \\ &= \int_{\varepsilon\leq|x|\leq 1, s\in[0,t]} x\{\tilde{J}_x(ds\times dx)\} \end{aligned}$$

The terms in the first equality are independent and the convergence in the last term is almost sure and uniform in t on $[0,T]$

This theorem is Proposition 3.7 from Cont and Tankov . The interpretation of J_x is that given a set $B=([t_1,t_2]\times A)$ with $A\subset\mathbb{R}$, $J_x(B)$ counts the number of jumps between t_1 and t_2 such that the jump size belongs to A. The reason for subtracting ν in the integral in $\tilde{X}_t^\varepsilon$ is to make sure that the integral converges. Below is a summary of some important implications of the decomposition.

- For every Lévy process there exist a triplet (A,ν,γ), called the Lévy triplet, that uniquely determine it's distribution.
- X_t^l is a compound Poisson process
- $\tilde{X}_t^\varepsilon$ is an infinite sum of independent compensated compound Poisson processes, were compensated means that a process is centered i.e. it has mean zero.

All of this together implies that every Lévy process can be seen as combination of a BM and an, possibly, infinite sum of compound Poisson processes. This result allows us to split Lévy processes in to two categories: Jump-Diffusions and Infinite Activity processes. Jump-Diffusions represent the processes with finitely many jumps and equivalently Inifinite activity processes represent processes with infinitely many jumps.

The Lévy-Itô decomposition also gives us the ability to calculate the mean and the variance of a Lévy process as functions of it's Lévy-triplet, by the following theorem.

Theorem 5.4. *Let $(X_t)_{t\geq 0}$ be a Lévy process on $\mathbb{R}$ with Lévy triplet (A,ν,γ). The n-th absolute moment of X_t, $E[|X_t|^n]$ is finite for every $t>0$ if and only if $\int_{|x|\geq 1}|x|^n\nu(dx)<\infty$. If this holds the mean and variance of X_t equals*

$$\begin{aligned} E[X_t] &= t(\gamma + \int_{|x|\geq 1} x\nu(dx)) \\ Var[X_t] &= t(A + \int_{-\infty}^{\infty} x^2\nu(dx)) \end{aligned}$$

This is theorem is a rewritten version of proposition 3.13 in Cont and Tankov . It also allows us to the characteristic exponent for compound Poisson processes as function of it's Lévy measure

Theorem 5.5. *Let $(X_t)_{t\geq 0}$ be a compound Poisson process on $\mathbb{R}$. It's characteristic exponent is then given by the following expression*

$$\psi(z) = \int_{-\infty}^{\infty} (e^{izx} - 1)\nu(dx) \forall z \in \mathbb{R}$$

Chapter 6

Jump-Diffusions

In the last chapter a class of processes called Jump-Diffusions were introduced, the purpose of this chapter is to extend our results regarding exit times from BMs to Jump-Diffusions. We start by looking on some theoretical results needed for this, then the simulation algorithm is presented and finally a simulation study is presented.

6.1 Theoretical Results

The idea of this chapter is to fit Jump-Diffusion processes in to the framework established in Chapter 2, i.e. we are interested in finding the times such that our process hits a grid of size η. In this chapter only one-dimensional Jump-Diffusions will be considered but the results are can easily be extended to the multi-dimensional case. Recalling, Jump-Diffusions are processes on the following form

$$\begin{aligned} X_t &= Q_t + Z_t \\ Y_t &= \gamma t + \sigma W_t \\ Z_t &= \sum_{i=0}^{N_t} Y_i \end{aligned}$$

where N_t is a Poisson process, and thus, X_t is a sum of a BM with drift and a compound Poisson process. As a result of the Lévy-Ito decomposition, we know that the Q_t and Z_t are independent. We also know that Z_t has finite activity, i.e. it has a finite number of jumps. The original definition of an exit time was, as follows

$$\tau_\eta(X) = \inf\{t > 0 : |X_t| \geq \eta\}$$

The results of Chapter 3 were given for BMs with drift and obviously it would be very hard to find a distribution for the exit time incorporating both the BM and the compound Poisson process. Instead, because of the independence of Y_t and Z_t, we can consider the following time

$$\hat{\tau}_\eta(X) \quad = \quad \min(\tau_\eta(Y), \mathbb{T}(Z_t))$$

where $\tau_\eta(Y_t)$ denotes the exit time for a BM defined as before and $\mathbb{T}(Z_t)$ denotes the time to the next jump for the compound Poisson process. Because of the independence we can simulate the two times independently and just take the min. We obviously know how to simulate exit times for a BM and a walk-through on simulating a compound Poisson process follows in the next section. If $\hat{\tau}_\eta(X) = \tau_\eta(Y)$ we know that the process has hit the barrier, but also that a jump has not occurred. On the other hand, if, $\hat{\tau}_\eta(X) = \mathbb{T}(Z_t)$ we know that Y_t has not hit the barrier, and we can simulate it's position using $\mathcal{L}$, and that a jump has occurred.

6.2 Simulation

6.2.1 The compound Poisson process

In order to simulate our exit time all that's left to work out is how to simulate a compound Poisson process (with finite activity). The compound Poisson process is a Lévy process, and the results of this chapter will be based on the assumption that it's given by it's characteristic triplet.

Since a compound Poisson process has no drift or BM-part the characteristic triplet will be given by $(A, \nu, \gamma) = (0, \nu, 0)$. From the Lévy-Ito decomposition we know that a Lévy process is uniquely determined by it's characteristic triplet and this implies that a compound Poisson process is uniquely determined by it's Lévy measure. In order to relate our Lévy measure to the jump size distribution f of the compound Poisson process we use the following theorem.

Theorem 6.1. *Let $(X_t)_{t\geq 0}$ be a compound Poisson process with intensity λ and jump size distribution f. It's jump measure J_X is a Poisson measure on $\mathbb{R}^d \times [0, \infty[$ with intensity measure $\mu(dx \times dt) = \nu(dx)dt = \lambda f(dx)dt$.*

The most important, in regards to this thesis, implication of this is that

$$\nu(dx)dt = \lambda f(dx)dt$$

If we now integrate over $\mathbb{R}$ w.r.t. x on both sides we get

$$\left(\int \nu(dx)\right) dt = \left(\int \lambda f(dx)\right) dt = \lambda dt$$

The second equality follows from the fact that f is a distribution. By dividing the original expression with λ on both sides we get.

$$f(dx)dt = \frac{\nu(dx)}{\lambda}dt = \frac{\nu(dx)}{\int \nu(dx)}dt$$

We now have everything needed to simulate a compound Poisson process. As a result of theorem 2.1 the number of, simulated, jumps k on a time interval [0,t]

is given by drawing from $Po(\lambda t)$. The time of the jumps are found by drawing k-samples from the U(0,t)-distribution. Finally, the size of jump is given by sampling from f(x), i.e. the jump size distribution. The simulation algorithm is shown below.

Algorithm 5 Compound Poisson Process Simulation (CPPS)

$k \leftarrow$ sample from $Po(\lambda t)$
for $i = 1 : k$ **do**
  $T_i^j \leftarrow$ draw sample from $U(0, t)$
end for
$\bar{T}^j \leftarrow$ sort $\bar{T}^j$ in growing order
for $i = 1 : k$ **do**
  $X_i^j \leftarrow$ draw sample from $f(x)$
end for
return $(\bar{T}^j, \bar{X}^j)$

The bar represents the whole vectors and the j is used to mark that we are dealing with jumps.

6.2.2 Jump-Diffusions

Given the results above the simulation algorithm for a whole path of Jump-Diffusion process is as follows.

Algorithm 6 DISCRETE SPACE SIMULATION FOR JUMP-DIFFUSIONS

$(\bar{T}^j, \bar{X}^j) \leftarrow$ simulate using the CPPS algorithm
while $\tau_{i-1} < T$ **do**
  $(\tau_i, X_{\tau_i}) \leftarrow$ simulate using Algorithm 3.2.1
  $\tau_i \leftarrow$ minimum of (τ_i, T, T_n^j)
  if $\tau_i = T$ **then**
    $X_{\tau_i} \leftarrow$ sample from $\mathcal{L}$
  else if $\tau_i = T_n^j$ **then**
    $X_{\tau_i} \leftarrow$ sample from $\mathcal{L}$
    $i \leftarrow i + 1$
    $\tau_i \leftarrow \tau i - 1$
    $X_{\tau_i} \leftarrow X_{\tau_{i-1}} + \bar{X}_n^j$
    $n \leftarrow n + 1$
  end if
  $i \leftarrow i + 1$
end while
return $(\bar{\tau}, \bar{X})$

6.3 Case Study: Merton Model

In order to verify the simulation algorithm for Jump-Diffusions it will be applied by pricing Barrier options for the Merton model. The log-price of the stock in the Merton model is defined in the following way

$$X_t = X_0 \exp\{\mu_{BM} t + \sigma_{BM} W_t + \sum_{i=1}^{N_t} Y_i\},$$

with $Y_i \in N(\mu_j, \sigma_j^2)$ and an intensity of jumps λ_j. It has the following Lévy density and characteristic exponent

$$\begin{aligned} \nu(x) &= \frac{\lambda_j}{\sigma_j\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma_j^2}\right\} \\ \psi(u) &= -\frac{\sigma_{BM}^2 u^2}{2} + i\mu_{BM} u + \lambda_j\{e^{-\sigma_j^2 u^2/2 + i\mu_j u} - 1\} \end{aligned}$$

In X_t, $\mu_B M$ shall be chosen such that discounted value of the stock-price, i.e. $S_t = S_0 \exp(X_t)$, is martingale, or in particular such that $E[S_T] = S_0 \exp(rT)$. The following expression for $\mu_B M$ ensures that.

$$\mu_{BM} = r - \frac{\sigma_{BM}^2}{2} - \lambda\{e^{\mu_j + \sigma_j^2/2} - 1\}$$

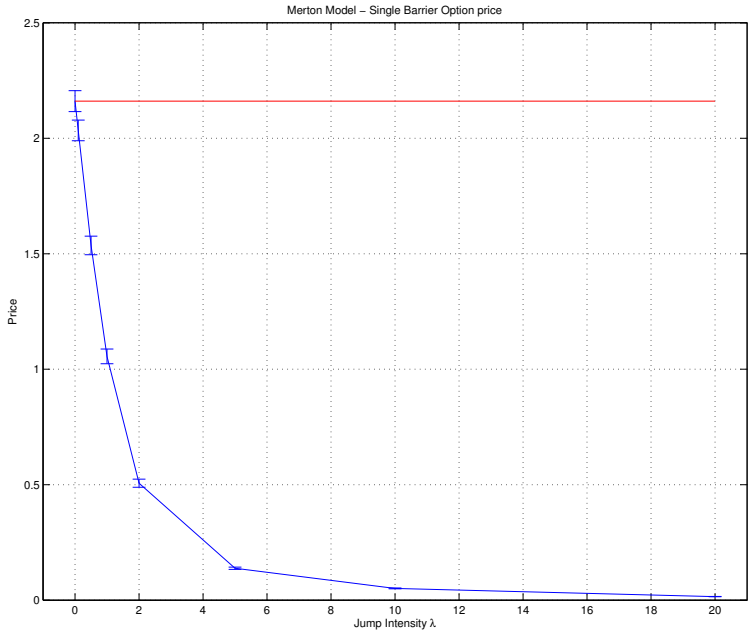
The jump intensity and jump size distribution are as follows (in this case they are actually given in the definition of the Merton model, but in order to show how to calculate them they are still included here)

$$\begin{aligned} \lambda &= \int \nu(dx) = \int \frac{\lambda_j}{\sigma_j\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma_j^2}\right\} dx = \lambda_j \\ f(x) &= \frac{\nu}{\int \nu} = \frac{1}{\sigma_j\sqrt{2\pi}} \exp\left\{-\frac{(x-\mu)^2}{2\sigma_j^2}\right\} \end{aligned}$$

Let's start by pricing an Up-and-Out Barrier call option. Since the drift and volatility are constant for the log-process the grid size, for the same reason as in the Black and Scholes case, won't impact the result of the simulation and should be set high to get a fast algorithm. Thus, $\eta = 0.5$. Instead of a varying grid size, let's vary the intensity of the jumps. As the intensity gets higher the probability of hitting the barrier gets higher and thus the price should get lower. The rest of the parameters are given below.

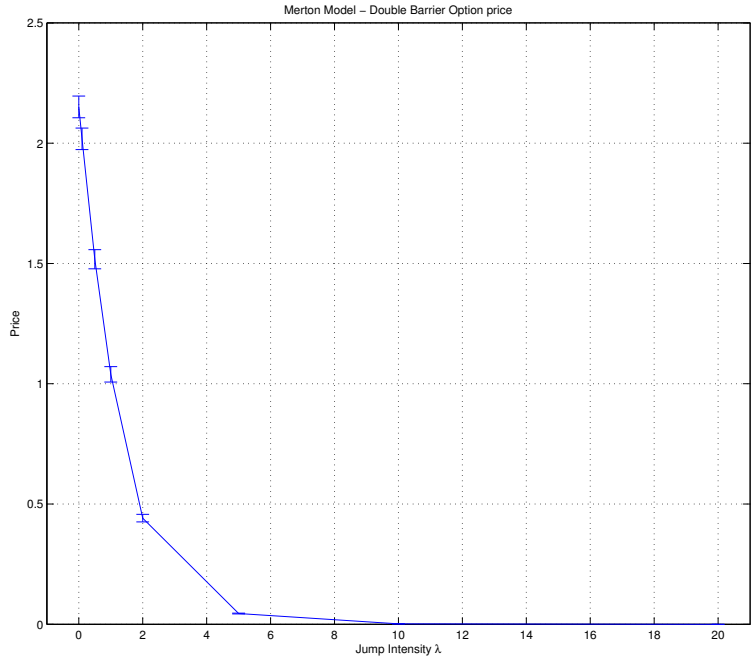
Variables	Abbreviation	Value
Number of Simulations	N	100000
Stock value today	S_0	50
Upper barrier	U	60
Strike price	K	50
Time to maturity	T	1
Risk-free rate	r	0.1
Volatility of BM	σ_{BM}	0.1
Mean Jump Size	μ_j	-0.1
Jump Size Variance	σ_j	0.09

The figure below shows the result of the simulation with the theoretical price in the Black and Scholes model, i.e. $\lambda_j = 0$, as a reference point (unfortunately I couldn't find another simulation algorithm to use as reference). It shows that the price gets lower as the intensity grows, almost exponentially.



The figure below shows the result of the same set of parameters but for for a Double Barrier option with upper barrier 60 and lower barrier 40. Once again the price seems to be diminishing exponentially with the jump intensity.

The algorithm seems like a perfect match for Jump-Diffusions. Since Jump-Diffusions have constant drift and volatility the grid only have to consider the barriers, and the only times it has to stop is at the jump times. This means that, especially for process with low intensity, that the algorithm will be fast. As an example the algorithm took 42 seconds for $\lambda = 2$ in the Merton model. The algorithm is also relatively simple to understand and implement, as a reference Barrier options for Jump-Diffusions can be priced by solving something called Partial Integro Differential Equations (PIDEs). PIDEs are like partial differential equations but with terms containing integrals and they are obviously more challenging to solve. Thus, our method seems like a good alternative.


Merton Model – Double Barrier Option price
Price
Jump Intensity λ
0
0.5
1
1.5
2
2.5
0
2
4
6
8
10
12
14
16
18
20

Chapter 7

Infinite Activity Processes

The purpose of this chapter is to extend the results of chapter 3 to Infinite Activity processes. An approximation is introduced that approximates Inifinite Activity processes with Jump-Diffusions. The thought behind the approximation is to truncate all jumps smaller than a specified value, ε, and replace them with a BM, cf. Asmussen and Rosinski [2]. At the end of the chapter a simulation study, in where Barrier options are priced in the Variance Gamma model, is performed.

7.1 Theoretical Results

As oppose to Jump-Diffusions, Inifinite Activity process have an infinite amount of jumps, i.e. jumps occur infinitely often. This also mean that they do not have to contain a BM-part and, therefore, we can't directly apply the results from our previous chapter to this one. But, recalling one of the results of the Lévy-Itô decomposition we know that any Lévy process can be approximated arbitrarily well by a Jump-Diffusion. Therefore, the idea as that we will simulate Inifinite activity processes by first approximating them with a Jump-Diffusion and then simulate them as in the previous chapter.

The Lévy-Itô decomposition told us that every Lévy process can be written on the following form.

$$X_t \quad = \quad \gamma t + W_t + X_t^l + \lim_{\varepsilon \downarrow 0} \tilde{X}_t^\varepsilon$$

Recalling, this is the sum of a BM ($\gamma t + W_t$), a compound Poisson process (X_t^l) and a limit of compensated compound Poisson processes. The approximation is done in two steps, first, all jumps smaller than a specified number, $\varepsilon > 0$, are truncated, giving us the approximation process Y_t^ε.

$$Y_t^\varepsilon = \gamma t + \sigma W_t + X_t^l + \tilde{X}_t^\varepsilon$$

Since $\tilde{X}_t^\varepsilon$ have an finite amount of jumps it doesn't need to be compensated and thus we can define $J_t^\varepsilon = X_t^l + \tilde{X}_t^\varepsilon$, this is a compound Poisson process with

jumps larger than ε. It's now clear that Y_t^ε is the sum of a BM and a compound Poisson process, i.e. a Jump-Diffusion. The approximation error of this process, R_t^ε, is given by the following expression.

$$R_t^\varepsilon = X_t - Y_t^\varepsilon = -\tilde{X}_t^\varepsilon + \lim_{\varepsilon \downarrow 0} \tilde{X}_t^\varepsilon$$

This is obviously also a Lévy process, that consist of all compensated jumps smaller than ε, with Lévy triplet $(A, \mathbf{1}\{|x| \leq \varepsilon\}\nu(dx), 0)$. The second part of the approximation is to approximate this error with a BM and add it to the approximation. The motivation behind this is that it can be shown that, under certain conditions, R_t^ε converges to a BM (in distribution). The first step towards proving this is to calculate the mean and variance of the R, this is done using theorem 5.4.

$$\begin{aligned} E[R_t^\varepsilon] &= 0 \\ V[R_t^\varepsilon] &= t \int_{|x|<\varepsilon} x^2 \nu(dx) \end{aligned}$$

We continue by defining $\sigma_\varepsilon^2 = \int x^2 \nu(dx)$ and $Q_t^\varepsilon = \sigma_\varepsilon^{-1} R_t^\varepsilon$. If we then assume that the following condition is satisfied.

$$\frac{\sigma_\varepsilon}{\varepsilon} \to \infty \quad as \quad \varepsilon \to 0$$

It means that, since the jumps of R are bounded by ε, that the jumps of Q will be bounded by a number that tends to zero. This, means that Q will tend towards a continuous Lévy process. This process will have mean zero and variance t and thus Q tends towards a SBM. Because of this we replace R with a BM with variance σ_ε^2, $\sigma_\varepsilon Q_t$.

$$\begin{aligned} Z_t^\varepsilon &= Y_t^\varepsilon + \sigma_\varepsilon Q_t \\ &= \gamma t + \sigma W_t + J_t^\varepsilon + \sigma_\varepsilon Q_t \end{aligned}$$

Since W_t and Q_t are obviously independent this can be rewritten in the following way.

$$Z_t^\varepsilon = \gamma t + (\sigma + \sigma_\varepsilon) W_t + J_t^\varepsilon$$

This is a Jump-Diffusion and can thus be simulated the same way as in the precious chapter. As a final note, convergence rates of this approximation are of course interesting when simulating the process, and the following theorem in Cont and Tankov [5] gives us a way of quantifying this

Theorem 7.1. *Let f be a real-valued differentiable function such that $|f'(x)| < C$ for some constant C. Then*

$$|E[f(X_T^\varepsilon + R_T^\varepsilon]) - f(X_T^\varepsilon + \sigma_\varepsilon W_T])| \leq A\rho(\varepsilon) C \sigma_\varepsilon,$$

where $\rho(\varepsilon) \equiv \frac{\int_{-\varepsilon}^{\varepsilon} |x|^3 \nu(dx)}{\sigma_\varepsilon^3}$ and A is a constant satisfying $A < 16.5$.

7.2 Case Study: Variance Gamma

Introduced in 1998 in Madan et al [7], the Variance Gamma model is an extension of the Black and Scholes model which replaced the deterministic time with a random time change. Introducing a random time change is used in other models too and it's often thought of as representing important financial time changes. The log-price in the Variance Gamma model is defined as follows

$$X_t = \omega t + \theta\Gamma(t;1,v) + \sigma W(\Gamma(t;1,v))$$

i.e. the deterministic time changes are replaced with a Gamma process, as defined in chapter 2. The process has three free parameters; θ, the drift of the BM, σ, the volatility of the BM, and v the variance of the underlying Gamma process. ω is there to ensure that the discounted stock-process is a martingale. The Lévy measure for X is given by.

$$\begin{aligned}\nu(x) &= \frac{1}{v|x|}e^{Ax-B|x|}\\ A &= \frac{\theta}{\sigma^2}\\ B &= \frac{\sqrt{\theta^2+2\sigma^2/v}}{\sigma^2}\end{aligned}$$

7.2.1 Implementation

In order to simulate this we approximate the process with

$$Z_t^\varepsilon = \mu_{BM}t + \sigma_{BM}W_t + J_t^\varepsilon$$

We start by calculating σ_{BM}

$$\begin{aligned}\sigma_{BM}^2 &= \int x^2\nu(dx) = \int_{-\varepsilon}^{\varepsilon} x^2\nu(dx) = \int_{-\varepsilon}^{0} x^2\nu(dx) + \int_0^{\varepsilon} x^2\nu(dx)\\ &= S_\varepsilon^1 + S_\varepsilon^2\\ S_\varepsilon^1 &= \int_0^{\varepsilon} x^2\nu(dx) = \frac{1}{v(A-B)}\left(\varepsilon e^{(A-B)\varepsilon} - \frac{e^{(A-B)\varepsilon}-1}{(A-B)}\right)\\ S_\varepsilon^2 &= \int_{-\varepsilon}^{0} x^2\nu(dx) = -\frac{1}{v(A+B)}\left(\varepsilon e^{-(A+B)\varepsilon} + \frac{e^{-(A+B)\varepsilon}-1}{(A+B)}\right)\end{aligned}$$

The second task is to calculate the intensity and jump size distribution of J_t^ε, but in order to simplify simulation we split J_t^ε into two parts, one for the positive jumps, $J_t^{\varepsilon+}$, and for the negative, $J_t^{\varepsilon-}$. These are also compound processes with Lévy densities $\nu^{\varepsilon+}(x) = \nu(x)\mathbf{1}\{x>\varepsilon\}$ and $\nu^{\varepsilon-}(x) = \nu(x)\mathbf{1}\{-\varepsilon>x\}$. The idea is to simulate them one by one and then merge the resulting vectors. Let's start by finding a way to simulate the positive part.

In order to simplify simulation even more, we see that

$$\nu^{\varepsilon+}(x) = \frac{1}{v|x|}e^{(A-B)x}\mathbf{1}\{x > \varepsilon\} \leq \frac{1}{v\varepsilon}e^{(A-B)x}\mathbf{1}\{x > \varepsilon\} \equiv g^{\varepsilon+}(x)$$

This means that $J_t^{\varepsilon+}$ can be seen as a thinned version of another process with Lévy density $g^{\varepsilon+}(x)$. Recalling results from chapter 2, we then know that $J_t^{\varepsilon+}$ can be simulated by simulating from this other process and then discarding jumps with probability $\nu^{\varepsilon+}(x)/g^{\varepsilon+}(x)$. All that is left to do is to calculate the intensity and jump size distribution for $g^{\varepsilon+}(x)$.

$$\begin{aligned}
\lambda_{g^{\varepsilon+}} &= \int g^{\varepsilon+}(dx) = \frac{1}{v\varepsilon(B-A)}e^{(A-B)\varepsilon} \\
f_{g^{\varepsilon+}}(x) &= \frac{g^{\varepsilon+}(x)}{\lambda_{g^{\varepsilon+}}} = (B-A)e^{-(B-A)\varepsilon}\mathbf{1}\{x > \varepsilon\}
\end{aligned}$$

We see that $f_{g^{\varepsilon+}}(x)$ has the same distribution as a shifted exponentially distributed variable and is thus very easy to simulate. Analogously the negative jumps can be simulated using the following functions

$$\begin{aligned}
g^{\varepsilon-}(x) &= \frac{1}{v\varepsilon}e^{-(A+B)x}\mathbf{1}\{x < -\varepsilon\} \\
\lambda_{g^{\varepsilon+}} &= = \frac{1}{v\varepsilon(A+B)}e^{-(A+B)\varepsilon} \\
f_{g^{\varepsilon+}}(x) &= (A+B)e^{-(A+B)\varepsilon}\mathbf{1}\{x < -\varepsilon\}
\end{aligned}$$

Finally, in order to in order to ensure that the discounted version of the approximated process is still a martingale, we need to find the exponential moment of J_t^{ε}. Since J_t^{ε} is a compound Poisson process it's characteristic exponent can be found using Theorem 5.5. This leads us to it's exponential moment.

$$\begin{aligned}
\psi_J(z) &= \int (e^{iux}-1)\nu^{\varepsilon}(dx) = \int_{\varepsilon}^{\infty}(e^{iux}-1)\nu(dx) + \int_{\varepsilon}^{\infty}(e^{iux}-1)\nu(dx) \\
\Lambda_J &= \int_{\varepsilon}^{\infty}(e^{x}-1)\nu(dx) + \int_{\varepsilon}^{\infty}(e^{x}-1) = \Lambda_{J^+} + \Lambda_{J^-} \\
\Lambda_{J^+} &= \frac{1}{\nu}\left(\int_{(A-B-1)\varepsilon}^{\infty}\frac{e^{-t}}{t}dt - \int_{(A-B)\varepsilon}^{\infty}\frac{e^{-t}}{t}\right) \\
\Lambda_{J^+} &= \frac{1}{\nu}\left(\int_{(A+B+1)\varepsilon}^{\infty}\frac{e^{-t}}{t}dt - \int_{(A+B)\varepsilon}^{\infty}\frac{e^{-t}}{t}\right)
\end{aligned}$$

Integrals of the form $\int_x^{\infty}\frac{e^{-t}}{t}dt$ are known as exponential integrals and they can't be calculated analytically. In order to calculate them we need some numerical tool, they are e.g. implemented in Matlab as the function "expint". So, μ_{BM} must be given by the following expression.

$$\mu_{BM} = r - \frac{\sigma_{BM}^2}{2} - \Lambda_J$$

7.2.2 Results

In order to show how the price converges for different ε we start by pricing a single barrier option, the results are then compared with a pricing algorithm presented in Avramidis et al [3]. Since, the BM-part of the model has constant drift and volatility, we know that the choice of η doesn't change the convergence rates. We only have to choose η such that the barrier is taken into consideration, i.e. $\eta_t = B - S_t$.The parameters used in the simulation were the following.

Variables	Abbreviation	Value
Number of Simulations	N	100000
Stock value today	S_0	50
Upper barrier	U	60
Strike price	K	30
Time to maturity	T	1
Risk-free rate	r	0.1
Drift of BM	θ	-0.31
Volatility of BM	σ	0.17
Variance of Gamma process	v	0.13

The results of the simulation are presented in figure 7.1, the straight line represents the other simulation algorithm. As seen in the Figure the price converges, towards the same value as the other algorithm, as ε tends to zero. There is one problem with our algorithm, as $\varepsilon \to 0$ the intensity of jumps increases, and thus our process will get to jump times rather than exit times more and more often. The problem with this is that we have to simulate from $\mathcal{L}$ instead of $\mathcal{P}$, it was showed in Chapter 3 that simulating from $\mathcal{L}$ is about 100 times slower than simulating from $\mathcal{P}$. It would thus be of interest to find a ε such that the number of jumps doesn't blow but still such that the bias is small. In this case $\varepsilon = 0.01$ could be a good candidate, the time of the simulation hasn't really blown up yet and the price have converged fairly well (the confidence interval covers the reference price). It should also be noted that the simulation of the jumps also takes a lot of time.

Figure 7.2 shows how the time changed with ε for the realization in Figure 7.1. As reference the reference algorithm took 120 seconds for the same number of simulations.

A simulation on Double Barrier option, using the same parameters but with K = 50 and the addition of a lower barrier at 40, was also performed. Unfortunately the reference algorithm wasn't adapted to Double Barrier options and thus no reference has been submitted. The results are shown in figure 7.3 and they are in accordance with the results for the Single Barrier option (the time aren't shown because they were basically identical to the ones in the Single Barrier case). The biggest difference is that the price didn't converge as fast as for the Single Barrier case, this seems to be due to the nature of the different contracts. Once again $\varepsilon = 0.01$ seems to be a good level truncation level.

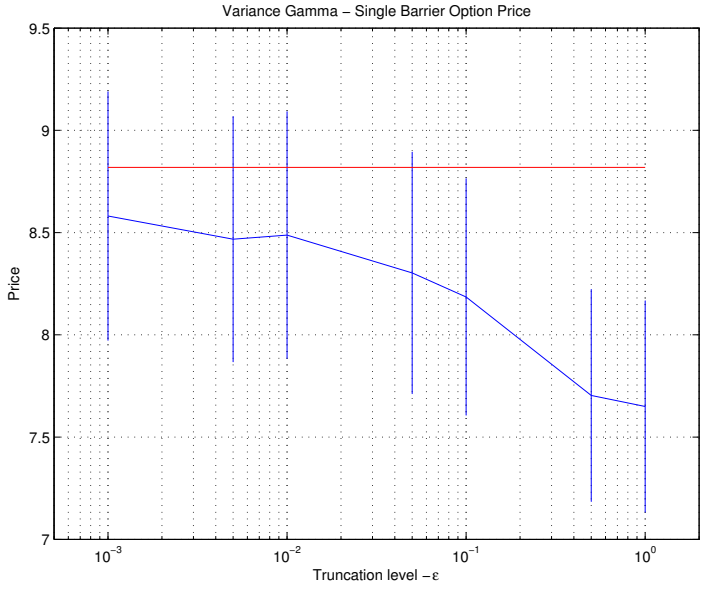


Figure 7.1: Convergence of Single Barrier Option Price

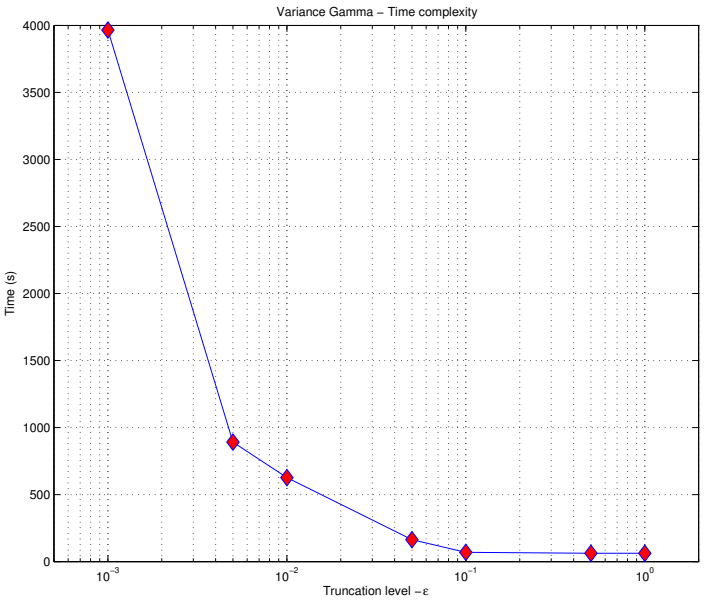


Figure 7.2: Simulation time as a function of the truncation level ε

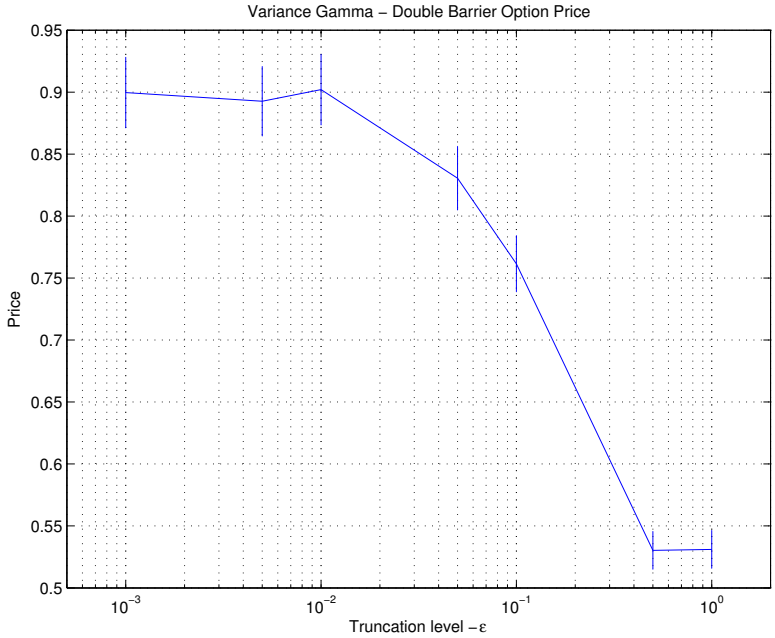


Figure 7.3: Convergence of Double Barrier Option Price

Chapter 8

Discussion

8.1 Summary

I started out by implementing a method for simulating a one-dimensional Brownian Motion on a discrete space grid, described in Adalbjörnsson and Quiroz . The algorithm was then extended to multi-dimensional Brownian Motions. After that I presented some results on Lévy processes, which was split in to two cases; Jump-Diffusions and Infinite activity processes. I continued by extending the simulation method to Jump-Diffusions and then presented a way of approximating Infinite activity processes with Jump-Diffusions and thus providing a way of using the simulation algorithm for an arbitrary Lévy process. In each case a case study was presented, in where I priced Barrier options under a given stock model.

8.2 Conclusion

Extending the method to an arbitrary Lévy process proved to work fairly well. The only real problem was in the Heston-model when the Feller condition was not fulfilled, with extremely slow convergence rates as a result. One other problem with the model was speed. Simulating $\mathcal{L}$ takes a lot of time.

On the positive side is that it's very easy to price path-dependent options in the process, for processes both with and without jumps. Altering the method to fit different models is very easy as opposed to most methods which are model specific (or at least confined to a small number of models). If the process has constant drift and volatility the method works very well since we can reduce the simulation to a very small number of steps without loss of precision.

The approximation process presented in chapter 7 also proved to work well, the price converged well before the number of jumps exploded.

8.3 Further Development

- The biggest problem of the algorithm was obviously speed. One of the main reasons for this was the fact that $\mathcal{L}$ was very slow to simulate. One way of speeding up the calculations would be to try and find a numerical approximation in the same way as for $\mathcal{P}$. A first step towards this is that it can be shown that:

$$\begin{aligned}\mathcal{L}'(\beta;t,\mu) &= \frac{1-\mathcal{P}(t)}{1-\mathcal{P}(t;\mu)}\exp(-\mu t/2)\exp(\mu\beta)\mathcal{L}'(\beta;t)\\ \mathcal{L}'(\beta;t) &= \frac{1}{1-\mathcal{P}(t)}\sum_{k=0}^{\infty}\cos\frac{\pi(2k+1)\beta}{2}\times\exp\{-\frac{1}{8}\pi^2(2k+1)^2t\}\end{aligned}$$

  Where the derivatives are w.r.t. β. A result of this is the following inequality.

$$\mathcal{L}'(\beta;t,\mu) \leq \frac{1-\mathcal{P}(t)}{1-\mathcal{P}(t;\mu)}\exp(-\mu t/2)\exp(\mu)\mathcal{L}'(\beta;t)$$

  This means that we can sample from $\mathcal{L}'(\beta;t,\mu)$ using acceptance-rejection sampling with $\mathcal{L}(\beta;t)$ as instrumental distribution. The thought is then to numerically approximate $\mathcal{L}(\beta;t)$ with a polynomial, or some other relatively simple analytical expression, and finding it's inverse. We would then have a much faster way of simulating $\mathcal{L}(\beta;t,\mu)$.

  Another approach could be to approximate $\mathcal{L}(\beta;t,\mu)$ with the following probability

$$\hat{\mathcal{L}}(\beta;t,\mu) = P(X_t<\beta)\mathbf{1}\{-1<\beta<1\}$$

- In chapter 7 a approximation method for approximating Infinite activity processes with Jump-Diffusions was presented. In order to improve it's performance it would be a good idea to, in some way, try to find the perfect relationship between performance and speed. One idea would be to keep the workload (K), i.e. basically the time, constant. Since simulating from $\mathcal{L}$ is much slower than simulating from $\mathcal{P}$, we define K as $K \equiv N\lambda T_\lambda$, i.e. the number of simulations N times the intensity of the jumps times the time each jump takes to simulate. We could then find some measure of the error and minimize it given that $K = N\lambda T_\lambda$. One measure of the error could be the squared bias plus the variance.

  Or, formally, given that we view our simulation algorithm as an estimator, δ of the price, θ of an option, we can define a loss function $L(p,\delta)$. δ is then a function of λ_ε and N. Let's say we choose the estimator with the lowest expected loss i.e. we're interested in the following.

$$\arg\min_{\delta\in\Delta} E_{\theta\in\Theta}[L(p,\delta)]$$

With $\Delta = \{\delta : N\lambda_\varepsilon T_\lambda = K\}$ and L given by e.g. $E_{\theta\in\Theta}[L(p,\delta)] \equiv Bias^2(\varepsilon) + Variance(\varepsilon, N)$. The bias could be given by either the expected bias or some bound on the (expected) bias.

The same sort of reasoning could be applied to the choice of the grid size η when the Euler approximation is needed but with η instead of ε.

- Instead of using Euler approximation one could try other approximation routines such as e.g. the Milstein second order approximation.
- Extending the concept of exit times to other stochastic processes such as the Gamma process.
- By using theorem 3.1 and solving the corresponding boundary value problem perhaps the distribution function of exit time for SDEs on the following form (i.e. the Black-Scholes model) can be found.

$$dXt = \mu X_t dt + \sigma X_t dW_t, X_0 = 0$$

- Finding a way of getting better convergence rates for the Heston model when the Feller condition is not fulfilled.
- Testing the method for other stock models.

Appendix A

Simulation Methods

A.1 Inversion Sampling

Let's say we are interested in sampling from a distribution with cumulative distribution function $F(x)$, let's also assume that $F(x)$ is continuous and increasing and that we have a way of generating uniformly distributed on the interval $[0, 1]$, then Inversion Sampling provides a way of sampling from the distribution. If we define the general inverse as.

$$F^{-1}(x) = \inf\{x \in \mathbb{R}; F(x) \geq u\}$$

Inversion sampling is then given by the following algorithm.

Algorithm 7 Inversion Sampling

1: $u \leftarrow$ sample from U(0,1)
2: $X \leftarrow F^{-1}(u)$
3: **return** (X)

X is then has distributed according to F, for a proof on this see Sköld [10].

A.1.1 Binary Search

Sometimes a analytical for F^{-1} can't be found and we're then forced to use numerical methods for finding $F^{-1}(u)$. If we start by defining a tolerance level and assume that $F^{-1}(u) \in [a, b]$, with b $>$ a, the binary search is given by the following algorithm.

Algorithm 8 Binary Search

1: $u \leftarrow$ sample from U(0,1)
2: $A \leftarrow F(a)$
3: $B \leftarrow F(b)$
4: **repeat**
5: $c \leftarrow \frac{a+b}{2}$
6: $C \leftarrow F(C)$
7: **if** C $>$ u **then**
8: $b \leftarrow c$
9: $B \leftarrow F(b)$
10: **else**
11: $a \leftarrow c$
12: $A \leftarrow F(a)$
13: **end if**
14: **until** $|G - u| < tol$
15: **return** (c)

C is then an approximation of $F^{-1}(u)$, the error $|G - u|$ will be bounded by $2^{-n}(b - a)$, where n denotes the number of iterations in the loop.

A.2 Acceptance-Rejection sampling

Let's assume we are interested in sampling from a distribution $f(x)$, defined on $\mathbb{R}$, but it's very difficult to sample from. Let g be another distribution, also defined on $\mathbb{R}$, such that $\forall x \in \mathbb{R}, f(x) \leq Kg(x)$, for some constant K $< \infty$. The acceptance-rejection method is then given by the following.

Algorithm 9 Acceptance-Rejection samping

1: **repeat**
2: $X \leftarrow$ sample from g
3: $u \leftarrow$ sample from U(0,1)
4: **until** $u \leq \frac{f(X)}{Kg(X)}$
5: **return** (X)

X is then a sample from f, for proof a see Sköld [10]. g is called the instrumental distribution in Acceptance-Rejection sampling.

A.3 Monte-Carlo simulation

Assume that we are interested in estimating the integral (or expected value).

$$\tau = E[\psi(X)] = \int \phi(x) f(x) dx$$

The (Basic) Monte-Carlo estimate of τ is then given by the following algorithm.

Algorithm 10 Acceptance-Rejection samping

1: **for** i = 1:N **do**
2: $X_i \leftarrow$ sample from f
3: **end for**
4: $t_N \leftarrow \frac{1}{N}\sum_{i=1}^{N}\phi(X_i)$
5: **return** (t_N)

t_N is then a unbiased estimate of τ and it can be shown that $\sqrt{E[(t_n-\tau)^2]} = O(n^{-1/2})$, for more on Monte-Carlo simulation see Sköld [10].

Bibliography

[1] S. I. Adalbjörnsson and M. Quiroz, *Discrete space simulation with applications to barrier options*, 2009.

[2] S. Asmussen and J. Rosinski, *Approximation of small jumps of lévy processes with a view towards simulation*, Journal of Applied Probabilistics, no. 38, 482–491.

[3] A. N. Avramidis and P. L'Ecuyer, *Efficient monte carlo and quasi-monte carlo option pricing under the variance gamma-model*, Management Science **52** (2006), no. 12, 1930–1944.

[4] T. Björk, *Arbitrage theory in continuous time*, 3 ed., Oxford University Press, 2009.

[5] R. Cont and P. Tankov, *Financial modelling with jump processes*, Chapman Hall, 2004.

[6] A. Gut, *An intermediate course in probability theory*, 2 ed., Springer-Verlag, 2009.

[7] Carr P. P. Madan, D. B. and E. C. Chang, *The variance gamma process and option pricing*, European Finance Review **2** (1998), 79–105.

[8] G. N. Milstein and M. V. Tretyakov, *Simulations of space-time bounded diffusions*, The annals of applied probability **9** (1999), no. 3, 732–739.

[9] L. C. G. Rogers and D. Williams, *Diffusions, markov processes and martingales*, vol. 2, Wiley, 1987.

[10] M. Sköld, *Computer intensive statistical methods - lecture notes for fms091/mas221*, 2 ed., Centre for Mathematical Sciences, Mathematical Statistics, Lund University, 2006.

[11] G. Sparr and A. Sparr, *Kontinuerliga system*, Studentlitteratur AB, 2000.