On Monte Carlo approximation of the Snell envelope with application to the pricing of American options

Peter Wieslander

Spring, 2013
1 Introduction

In the 1970s two innovative papers were published independently of each other; Black and Scholes [3] and Merton [18]. With these papers the foundation of modern financial mathematics was laid. The authors presented a mathematical model for a financial market containing a financial instrument, known as derivative. This model is today known as the Black-Scholes model or Black-Scholes-Merton model. From this model it is possible to derive the well-known Black-Scholes formula, which is used for pricing European options. In 1997 Merton and Scholes received Nobel memorial price in economics for their pricing theory. Unfortunately, Black passed away 1995 and could not be part of the price.

Once the market acknowledged their theories, the volume of the derivative market exploded in 1980s and has continued to grow since then. However, after the financial crash in 1987 it become clear that the Black and Scholes model could not reflect properly the stock market for the purpose of derivative trading. Some of the criticism include the assumption of 1) constant volatility over time, and 2) constant interest rate 3) that the asset does not pay any dividends. Nevertheless, the model is still being used, but now mostly as a benchmark.

The global derivative market is huge. In 2010 [10] the global derivative market was estimated to $1.2 quadrillion or $1.2 \times 10^{15}$. This immense number can be compared to the world’s annual gross domestic product which is between $50 - $60 trillion. The estimated global derivative market is 20 times bigger. It should be noted that these figures regard the notional value of the derivatives, i.e. the value of the underlying asset that the derivative is based on.

The market can be divided into two: derivatives that are exchange-traded and over-the-counter (OTC) derivatives. On the exchange market individuals trade standardized contracts that have been defined by the exchange. The world’s oldest exchange, Chicago Board of Trade (which in 2007 merged with the Chicago Mercantile Exchange to form the CME group), opened in 1848. A few years later, in 1864, CBT listed the first standardized exchange-traded contract. This contract was called a futures contract. Futures are contracts to either buy or sell an asset on or before a future date. These contracts were originally used by farmers and merchants to standardize the quantity and quality of grains that were traded. Speculators soon found it to be more attractive to trade the contract itself instead of the underlying asset.

Over-the-counter trading is done directly between two financial institutions or a financial institution and one of its clients. An advantage over the exchange market is that contracts traded over-the-counter can be tailor-made without any terms specified by the exchange. A disadvantage is that there exists a small risk for over-the-counter trades. One of the parties may default and the contract will not be honored. The over-the-counter market is much larger than the exchange traded market.

These traded contracts are called derivatives. A derivative is a financial instrument whose price is derived from another asset, called underlying asset. Underlying assets can be everything from crude oil to gold. A common derivate is the stock option, which
derives its value from a company’s stock. The holder of a stock option has the right to either sell or buy a certain quantity of a company’s stock by a certain date for a certain price. The price in the contract is known as the strike price and the date is known as expiration date or maturity. A put option gives the holder right to sell the underlying asset, and a call option gives the holder right to buy the underlying asset. There exists several types of the stock option. The most traded type is the American option, which will be the derivative of interest in this paper. Unlike the European stock option which can be exercised at maturity only, American options can be exercised at any time up to maturity. The valuation of the option, even in the standard case of a lognormal process for the underlying asset, is to this date an active research topic. The American option valuation problem does not have a closed-form solution, and thus many numerical methods have been developed. The numerical methods can be categorized in three different categories; binomial trees, finite difference methods and Monte Carlo methods.

Monte Carlo methods have become one of the most important numerical algorithms of the 21th century. With Monte Carlo methods one aims to solve a problem by simulating the underlying process (e.g. a stock) and then calculate the average of some function of the simulated process. A great advantage of Monte Carlo methods is its flexibility, the methods can be applied to a large set of fundamental problems in statistics, numerical analysis and other scientific disciplines. Monte Carlo methods can also handle high dimensional problems effectively.

In this thesis we present a novel forward backward simulation-based method for solving the Snell envelope. The problem of valuating American options is special case of the Snell envelope. It is well-known that the problem of American option valuation can be solved by applying a backward dynamic programming formula where one has to, at each time step \( t \), compare the immediate payoff \( \Phi(S_t) \) with the expected future value \( C_{t+1} \) (often called the continuation value). Our novel method approaches the problem by first simulating a forward grid, denoted \( S \), using the density of the underlying asset. We proceed by simulating backwards, starting at \( t = T \) (maturity) and conditionally on the initial value \( S_0 \), to create a so-called backward mesh, denoted \( B \). With this choice of mesh, the continuation value can be estimated with a single value \( \hat{\Pi}(t+1, B_{t+1}) \). This should be an unbiased estimator of the continuation value. However, the numerical results in chapter 5 show that the estimator is not an unbiased estimator. We have not found an answer why the numerical results don’t agree with the theory of the method presented in chapter 3. Al tough, with the disappointed results of an unbiased estimator the numerical results do show that our novel method is faster and has lower variance than the well-established Stochastic Mesh method.

Apart from the theory of our novel method, I also cover the mathematical foundations for option pricing; Black & Scholes market, arbitrage pricing, risk neutral valuation, optimal stopping times, and the Snell envelope. The mathematical foundations are covered in chapter 2. In chapter 3 an introduction to Monte Carlo methods is given, including variance reduction techniques and simulation of Brownian motion. In chapter 4 I have made a compilation of the most common Monte Carlo methods for option
pricing. The thesis is wrapped up with numerical results in chapter 5, where I have studied the properties of three methods; the least squares method, the stochastic mesh and the backward grid method.

1.1 Acknowledges

I would also take the opportunity to thank my supervisor, Jimmy Olsson, for his guidance during my work with this thesis.
2 Preliminaries

When pricing derivatives using Monte Carlo methods, there are a few principles from the theory of mathematical finance that are particularly important:

**Arbitrage Pricing.** The general idea behind the arbitrage pricing principle is that arbitrage opportunities do not exist. An arbitrage opportunity is financial strategy which does not cost anything and may give a positive cash flow in the future. In other words, an arbitrage opportunity gives you a chance to earn money without any risk.

**The Hedge.** If a derivative can be perfectly replicated (hedged) through trading in other assets, then the price of the derivative is the cost of the replicating trading strategy. The assumption that a derivative can be perfectly hedged is crucial if we want to avoid arbitrage opportunities.

**Martingale Pricing.** Discounted asset prices are martingales under a probability measure associated with the choice of discount factor. Prices are expectations of discounted payoffs under such a martingale measure. A martingale is a process where the conditional expected value of the next observation, given all past observations, is equal to the last observation. A martingale can be viewed as a fair game. In a fair game, the gambler can expect to not lose any money nor earn any money. Counter examples of a fair game are all games in the casino where one plays against the bank. The games have been constructed with a slight advantage for the bank which in the long run will lead to a profit for the bank.

**Completeness.** In a complete market all derivatives can be hedged. In a complete market no arbitrage opportunities exist. In practice no market is truly complete and arbitrage opportunities do occur from time to time. However when they do appear, it is reasonable to assume that they are immediately taken advantage of and will thus be eliminated.

2.1 Black-Scholes-Merton Market

The Black-Scholes-Merton market consists of a stock \( S_t \) (risky asset) and a bank account \( B_t \) (risk free asset):

\[
\begin{align*}
    dS_t &= \mu S_t dt + \sigma S_t dW_t, \quad S_0 = s, \\
    dB_t &= rB_t dt, \quad B_0 = 1.
\end{align*}
\] (2.1)

The Black and Scholes model is a reference to the model assumption of \( S_t \) alone. The price process \((S_t)_{t \geq 0}\) of the stock is assumed to follow the geometric Brownian motion, i.e.

\[
dS_t = \mu S_t dt + \sigma S_t dW_t
\] (2.2)

This SDE has the closed form solution

\[
S_t = S_0 \exp \left( (\mu - \frac{\sigma^2}{2}) t + \sigma dW_t \right)
\] (2.3)
The process is defined by the drift $\mu$ and the volatility $\sigma$. The drift can be interpreted as the expected return of the stock in the time interval $dt$. For reasons that will become clear later, we need to have that the drift $\mu$ is equal to the interest rate $r$. The volatility of the stock is a measure of the return variability around its expectation $\mu$. The volatility can be estimated using historical data of the asset price. If the underlying asset pays dividend, we should instead consider the dynamics given by

$$dS_t = (r - \delta)S_t dt + \sigma S_t dW_t$$

(2.4)

In real markets stocks usually pay dividend at fixed dates, but we assume that stocks pay a continuously compounded dividend yield at rate $\delta$.

The driver of the process is the Wiener process $W = (W_t)$. $W$ is a Wiener process if it satisfies the following properties:

1. $W_0 = 0$
2. The increments are independent and stationary, i.e. for $0 \leq h \leq s \leq t$ $(W_s - W_{s-h})$ is independent of $(W_{t+h} - W_t)$.
3. An increment $(W_{t+h} - W_t)$ is normal-distributed with zero mean and variance $h$.
4. $W_t$ has continuous paths.

Modeling stock prices as geometric Brownian motion is accepted by practitioners for short and medium maturity.

Further we assume that there are no transaction costs when trading the underlying asset and that investor can invest at a risk-free rate, $r$.

2.2 Derivatives

Now we are going to give a mathematical interpretation of a derivative. A derivative or a contingent claim is a stochastic variable of the form

$$X = \Phi(Z)$$

(2.5)

where $Z$ is the stochastic variable driving the process of the underlying asset. If we consider the Black and Scholes market, then the process $Z$ is the Wiener process defined above. The function $\Phi$ is called the contract function or payoff function.

The holder of a derivative contract will get some payoff, $\chi$, which can be negative, at a given time (defined by maturity or the holder of the contract). Arguably one of the most simple derivatives is the forward contract. It is an agreement to either sell or buy an asset at a certain future date for a price $K$. The payoff function for a forward contract is

$$\Phi(S_T) = \begin{cases} 
S_T - K, & \text{Call} \\
K - S_T, & \text{Put} 
\end{cases}$$

(2.6)

There exist many different kinds of derivatives. Looking at their core features they can be categorized in four classes:
**European Style.** This class of derivatives are the simplest and are often called *simple derivatives*. These derivatives do only depend on the underlying asset at one fixed date \( T \) (maturity date). These means that the payoff function can be written \( \Phi(S_T) \). Examples of simple derivatives are Forwards, European options and Binary options.

**Path dependent.** These derivatives depend on the underlying asset on the interval \([0, T]\). For example the *Asian option* gives the holder the option to choose the mean of the underlying asset during \([0, T]\) or some value \( K \).

**American Style.** American style derivatives are contracts where the holder decides to exercise the contract on the time interval \([0, T]\). Unlike other derivatives, American style derivatives depend not only on the path of the underlying asset, but also on some choice made by the holder. This extra feature make pricing American style derivatives much more complicated. Examples are American options and Bermudan options.

**Perpetual.** Perpetual derivatives are like American style derivatives but without an exercise date defined \((T = \infty)\). Examples are *Russian options*.

2.3 Arbitrage Pricing

Consider the Black and Scholes market and a European style derivative, i.e. \( \chi = \Phi(S_T) \). Assume that the interest rate \( r \) is a deterministic constant, that the derivative can be traded on a market and that the price process of the derivative at time \( t \) is of the form

\[
\Pi(t) = F(t, S_t),
\]

where \( F \in C^{1,2}[0, T] \times \mathbb{R}^+ \).

It can be shown that the arbitrage free price is given by \( \Pi(t) = F(t, S_t) \) where \( F \) satisfies the PDE

\[
F_t(t, x) + r x F_x(t, x) + \frac{1}{2} x^2 \sigma^2(t, x) F_{xx}(t, x) - r F(t, x) = 0 \tag{2.8}
\]

\[
F(T, x) = \Phi(x) \tag{2.9}
\]

where \( F_x \) denotes partial derivatives. The PDE given by the above equations is the famous *Black and Scholes Equations*. It is important to note here that the price of the derivative, \( \Pi \), is given as a function of the underlying asset \( S \). Derivatives are completely defined by the underlying asset, they *derive* their value from underlying asset. Being able to price the derivative in a way that is consistent with the price process of the underlying asset (or assets) is crucial if we want to avoid arbitrage opportunities.

The solution of the PDE can be obtained by the *Feynman-Kač formula*.

**Feynman-Kač.** Assume that \( F \) is the solution to the PDE

\[
F_t(t, x) + \mu(t, x) F_x(t, x) + \frac{1}{2} \sigma^2(t, x) F_{xx}(t, x) - r F(t, x) = 0 \tag{2.10}
\]

\[
F(T, x) = \Phi(x) \tag{2.11}
\]
Assume further that the process $e^{-r_s} \sigma(s, X_s) F_x(s, X_s)$ belongs to $L^2$\(^1\), then $F$ has representation

$$F(t, x) = e^{-r(T-t)} E_{t,x}[\Phi(X_T)]$$

(2.12)

where $X$ satisfies the SDE

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

(2.13)

$$X_t = x$$

(2.14)

The solution (2.12) obtained with Feynman-Kač is in the convenient form of a discounted expectation.

It can be stressed enough that the price obtained from the Black and Scholes equations is only valid when the Black and Scholes market is assumed and the derivative is of European style, i.e. the payoff function can be written $\Phi(S_T)$. Hence, we can not use this pricing formula for path dependent derivatives, e.g. American options. Without analytical formulas at hand, we are left with approximation methods to obtain the value of American options.

### 2.4 Risk Neutral Valuation

From the previous section we know that the price of derivative with payoff function $\Phi$ is given by

$$\Pi(t) = e^{-r(T-t)} E[\Phi(X_T)]$$

(2.15)

where $X$ has the dynamics

$$\begin{align*}
    dX_s &= rX_s ds + \sigma X_s dW_s \\
    X_t &= x
\end{align*}$$

(2.16)

However, the initial assumption of the dynamics for the stock was

$$S_t = \mu S_t dt + \sigma S_t dW_t$$

(2.17)

Which is almost (2.16), apart from the drift $\mu$. If we study the equation (2.8) we see that this does not depend on on the drift term. This means that the drift term is not necessary in valuation of options and we do not have to estimate the drift of the underlying asset when using Monte Carlo methods to price options.

---

\(^1\) A process $g$ belongs to the class $L^2$ if the following conditions are satisfied

- $\int_a^b E[g^2(s) ds] < \infty$,
- The process $g$ is adapted to the $\mathcal{F}_t^W$-filtration.

The second condition says that, at every time, the filtration gives us enough information to find the value of the process $g$. 

8
The reason we set $\mu$ to be equal to the risk free rate $r$ is that the discounted price process $S_t/B_t$ is then a martingale. Why this is useful for us will be explained shortly, but first we say a few words about this property. The property is known as the Martingale property, which can be summarized by the following proposition (where we follow Björk’s [2] notations)

**The Martingale Property.** *In the Black and Scholes model, the price process $\Pi(t)$ for every traded asset, be it the underlying or derivative asset, has the property that the normalized price process

$$Z(t) = \frac{\Pi(t)}{B(t)}$$

is a martingale under the measure $Q$."

In the proposition the $Q$-measure is a reference to the dynamics given by (2.16). How does the martingale Property help us pricing derivatives? Remember that a martingale process is a process where the conditional expected value of the next observation, given all past observations, is equal to the last observation. The formal definition is

**Martingale Process.** A discrete random process $\{X_t; t = 0, 1, \ldots\}$ is a Martingale if

$$E[X_m|\mathcal{F}_n] = X_n, \text{ for all } n \leq m,$$

where $\mathcal{F}_n$ is the filtration\(^2\) of $X$ at time $n$.

From the definition we can conclude that the best prediction of the value tomorrow is the value today, $X_t$. This is indeed a very attractive feature when we want to determine the price today of an derivative. Consider the price process given by (2.18). Using the fact that it is a Martingale process, it can be rewritten as

$$\frac{\Pi(t)}{B(t)} = E \left[ \frac{\Pi(T)}{B(T)} | \mathcal{F}_t \right].$$

(2.20)

Multiplying with $B(t)$ yields

$$\Pi(t) = B(t)E \left[ \frac{\Pi(T)}{B(T)} | \mathcal{F}_0 \right],$$

(2.21)

where the expectation is taken under the $Q$-measure. The equation (2.21) is known as the risk neutral valuation formula. $B(t)$ can be any stochastic discount factor, but in the $B$&$S$-market $B(t)$ represents the bank account and is deterministic for all $t$. If we assume $B(0) = 1$, we thus get

$$\Pi(0) = e^{-rT}E [\Pi(T)|\mathcal{F}_t].$$

(2.22)

\(^2\mathcal{F}_t^X\) is a $\sigma$-algebra generated by all the random variables $\{X_s\}_{s \leq t}$. Basically, the filtration $\mathcal{F}_t^X$ contains all information available up to time $t$ (e.g. how the price process of a stock has evolved up until time $t$).
2.5 Optimal Stopping Theory and the Snell Envelope

We are now going to give a short introduction to the Snell envelope, which will be an important tool for valuation of American options. Actually, the valuation of American options is a special case of the Snell envelope. The idea was developed by Snell [20] in the 50’s.

**Definition. The Snell Envelope.** If \( X = (X_n)_{n=0}^N \) is a sequence adapted to a filtration \( \mathcal{F}_n \) with \( E (|X_n|) < \infty \), the sequence \( Z = (Z_n)_{n=0}^N \) defined by

\[
\begin{cases}
  Z_N = X_N, \\
  Z_n = \max (X_n, E[Z_{n+1}|\mathcal{F}_n]) & n \leq N - 1,
\end{cases}
\]

(2.23)

is called the Snell envelope of \( X \).

The sequence \( Z \) can be considered to be an optimal value process,

\[
Z_n = \sup_{n \leq \tau \leq T} E[X_\tau|\mathcal{F}_n],
\]

(2.24)

where “sup” in this case denotes the essential supremum. The recursion (2.23) defines the optimal stopping strategy to find the optimal value \( Z_n \). This recursion is known as the method of dynamic programming, which we will study in more detail in chapter 4. The time that maximizes the (essential) supremum (2.24) is called a optimal stopping time and is denoted \( \hat{\tau}_n \).

**Definition** A nonnegative random variable \( \tau \) is called an (optional) stopping time w.r.t. the filtration \( F = \{\mathcal{F}\}_{t \geq 0} \) if it satisfies the condition

\[
\{\tau \leq t\} \in \mathcal{F}_t \text{ for all } t \geq 0.
\]

(2.25)

To see that the Snell envelope indeed is an optimal strategy, we can consider the following example (where we follow Björk’s notations): Consider the tree stopping strategies:

**Strategy 1:** We use the optimal stopping strategy \( \hat{\tau}_n \).

**Strategy 2:** We stop immediately.

**Strategy 3:** We do not stop at time \( n \). Instead we wait until time \( n + 1 \), and from time \( n + 1 \) we behave optimally, i.e. we use the stopping time \( \hat{\tau}_n + 1 \)

Now compare the values of these strategies. The value of strategy 1 is, by definition, given by \( Z_n \). The value of strategy 2 is equal to \( X_n \). The value of strategy 3 is not as clear as the two others. We are assumed to behave optimally at time \( t = n + 1 \), so the value at time \( t = n + 1 \) must be equal to \( Z_{n+1} \). However, we are interested of the value at time \( t = n \) and this should be given by the conditional expectation \( E[Z_{n+1}|\mathcal{F}_n] \). Since strategy 1 is by definition optimal, we have the following inequalities

\[
Z_n \geq X_n, \quad Z_n \geq E[Z_{n+1}|\mathcal{F}_n].
\]

(2.26)
Since we at time $t = n$ we have only two options for the optimal stopping time $\hat{\tau}_n$: We can either stop immediately, in which case $\hat{\tau}_n = n$ and $Z_n = X_n$, or else it is optimal to keep going in which case $\hat{\tau}_n = \hat{\tau}_{n+1}$ and $Z_n = E[Z_{n+1} | \mathcal{F}_n]$. The optimal value at time $n$ should thus be given by

$$Z_n = \max\left( X_n, E[Z_{n+1} | \mathcal{F}_n] \right).$$ (2.27)

2.6 American Options

The holder of an American option has the option to either buy (call) or sell (put) the underlying asset $S$ at a certain price $K$ at an optional time $\tau \leq T$. The payoff function is

$$\Phi(S) = \begin{cases} (S_\tau - K)^+ & \text{(call)} \\ (K - S_\tau)^+ & \text{(put)} \end{cases}$$ (2.28)

where $(x)^+$ is short for $\max(x, 0)$. From this definition of the payoff function we see that the value of an American option is the value achieved by exercising optimally. Finding the optimal exercise rule is an optimal stopping problem. With the added feature of allowing for early exercise makes the valuation of American options complicated compared to European options. For each value of $S_t$, $t = T, \ldots, 0$ we have to determine the price of the option and check if it is favorable to exercise.

This problem is known as a free boundary problem. At each time $t$ we have two options: hold on to the option or exercise. With only two options given, there must exist a value of $S$ that marks the boundary between these two regions. This price, which we can denote $S_\tau$, is called the optimal exercise price. We do not know $S_\tau$ a priori and thus we do not know where to apply the boundary conditions, why this problem is referred to as a free boundary problem. Hence, the problem of pricing an American option is to find the optimal expected discounted payoff

$$\sup_{\tau \in \mathcal{T}} e^{r(T-\tau)} E[\Phi(S_\tau)],$$ (2.29)

where $\mathcal{T}$ is the class of admissible stopping times taking values in $[0, T]$.

Finding a fair value is not an easy task due to the complexity of the derivative. Unlike in the case of simple derivatives where we can derive the value in a closed-form solution, we have to turn to numerical procedures when we want to value an American option except in a case when the option is call option on non-paying dividend asset. To show this we follow the example given in Björk [2]. Consider an American call option on a non-dividend-paying underlying stock with strike price $K$ and maturity $T$. In this special case the value of the American option coincide with the value of an European call option with the same strike price and maturity on the same underlying stock. To see this, first note that

$$\Pi^A(t, s) \geq \Pi^E(t, s),$$ (2.30)

where $\Pi^A(t, s)$ denotes the pricing function of the American option and $\Pi^A(t, s)$ denotes the pricing function for the European option. The inequality (2.30) can be
made plausible if we note that the American option is like an European option but with option to exercise before maturity. Hence, a American option should be worth at least as much as its European counterpart.

Then we have the less obvious inequality

$$\Pi^E(t, s) \geq s - Ke^{-r(T-t)}.$$  \hspace{1cm} (2.31)

To see that this inequality holds, consider two portfolios $A$ and $B$. $A$ consist of a long position in the European option and $B$ consists of a long position in the underlying asset and a loan with maturity $T$ and face value $K$.

Now let $A_t$ and $B_t$ denote the value of the portfolios at time $t$. Regardless of the value at time $T$ we will have $A_T \geq B_T$ and to avoid arbitrage possibilities we must have $A_t \geq B_t$ for all $t \leq T$. The last inequality is exactly the same as (2.31).

We then get the trivial inequality (assuming we have positive rate of interest)

$$s - Ke^{-r(T-t)} > s - K, \forall t < T.$$  \hspace{1cm} (2.32)

We thus end up with

$$\Pi^A(t, s) > s - K, \forall t < T.$$  \hspace{1cm} (2.33)

Note that the right hand side of the inequality is the value of exercising the option at time $t < T$. Hence, it is never optimal to exercise the option before maturity. For the most cases we need to consider some numerical methods when we want to value an American option. The numerical methods can be divided in three categories

**Binomial trees.** Binomial tree method was one of the first numerical methods to price American options and was first proposed by Cox, Ross, and Rubinstein [15]. The binomial tree is a discrete-time and discrete-space approximation to the evolution of the driving process of the underlying asset. It assumes that from one time step to the next the price of the underlying asset can either go up to $Su$ with probability $p$ or decrease to $Sd$ with probability $1-p$ from its initial value $S$, where $u \geq 1$ and $0 < d < 1$ are constants.

Valuing option prices with binomial trees is achieved with backwards induction. At the terminal nodes (which corresponds to maturity date) the payoff is calculated, at any other node the payoff is determined by discounting the values at its successors nodes.

If we let the number of time steps increase to infinity then the price given by the binomial price of a European option (on a non-dividend paying stock) will converge to the Black and Scholes price, since the model assumption will lead to the lognormal assumptions for stock prices that are assumed in the B&S mode when the number of steps increases.

---

$^3$Taking a long position in an asset means that you acquire the asset with the expectation that the value of the asset will rise. The opposite is going short, where you want to sell an asset you do not own (but may be borrowed). In short position you want the price of the asset to decrease.

$^4$Face value is the nominal value of a security stated by the issuer of e.g. a loan.
Although the binomial tree method is simple to implement and can be really fast compared to other methods, it is quite limited in practice since the computational requirement scales exponentially with the number of state variables, making it ineffective for high-dimensional problems.

**Finite difference methods.** Just like the binomial tree methods, finite difference methods falls into the category of lattice based methods. Finite difference method were first applied on American option pricing by Brennan and Schwartz [6]. Here we try to solve the differential equations (2.8) that the option must satisfy, by converting the differential equation into a set of difference equations that are solved iteratively. Due to the exponential growth of computational cost with the number of state variables, finite difference methods become impracticable for high-dimensional problems.

**Monte Carlo simulation.** The ideas behind the Monte Carlo method was developed in the 40’s by mathematician Stanislaw Ulam with help from John von Neumann. Their ideas were later used in the Manhattan project\(^5\).

In the 60’s the first financial applications of Monte Carlo methods was suggested by Hertz [13] and Boyle [5] later proposed that Monte Carlo methods could be used to value European style derivatives. It was not until Bossaerts [4] and Tilley [21] suggested that Monte Carlo methods could be used to price American style derivatives.

Pricing options using Monte Carlo methods includes simulation of paths of stochastic processes used to describe the evolution of underlying asset prices, calculation of the payoff at each paths and taking average of the discounted payoff. If we e.g. consider the problem of pricing a European option, we draw a sample \(s = (s_1, \ldots, s_n)\) from the distribution of \(S_T\) and let the Monte Carlo estimate of the price be given by

\[
\hat{\Pi}_0 = \frac{1}{n} \sum_{i=1}^{n} e^{-rT} \Phi(s_i).
\]

\(^5\)Manhattan Project is the name of the research and development project that produced the first atomic bombs during World War II

13
3 Monte Carlo Methods

In this chapter we are going to brush up on our knowledge of Monte Carlo methods. We start with reviewing the basics and some important theorems. Once the basics have been covered, we present the most common variance reduction techniques and an algorithm for stock paths simulation. When using Monte Carlo methods the main goal is to compute some expectation

\[ \tau := \mathbb{E}[\phi(X)] = \int_X \phi(x) f(x) \, dx, \quad (3.1) \]

where

- \( X \) is a random variable taking values in \( X \subseteq \mathbb{R}^d, \ d \in \mathbb{N}^* \),
- \( f : X \rightarrow \mathbb{R}_+ \) is the probability density of \( X \) (known as target density),
- \( \phi : X \rightarrow \mathbb{R}_+ \) is a function (known as the objective function) such that the above expectation is finite.

This may at a first glance seem to be a bit limited, but this framework actually covers a large set of fundamental problems in statistics, numerical analysis and other scientific disciplines. If we e.g. want to compute the following integral

\[ \alpha = \int_0^1 f(x) \, dx, \quad (3.2) \]

we can instead consider the expectation \( \mathbb{E}[f(U)] \), with \( U \) uniformly distributed between 0 and 1. Now suppose that we can generate points \( U_1, U_2, \ldots, U_n \) independently and uniformly over the interval \([0,1] \). Evaluating \( f \) at these points and taking the mean we get the Monte Carlo estimate

\[ \hat{\alpha} = \frac{1}{n} \sum_{i=1}^{n} f(U_i). \quad (3.3) \]

By the law of large numbers this estimate will converge to the true value.

**Law of Large Numbers.** Let \( X_1, \ldots, X_N \) be independent random variables with density function \( f \). Then, as \( N \) tends to infinity,

\[ \tau_N := \frac{1}{N} \sum_{i=1}^{N} \phi(X_i) \rightarrow \mathbb{E}[\phi(X)], \ (a.s.). \quad (3.4) \]

Another very important theorem is the central limit theorem.

---

*Almost surely, i.e. with probability 1*
Central Limit Theorem. Let $X_1, \ldots, X_N$ be independent random variables with $E[X_i^2] < \infty$ and $S_N = X_1 + \cdots + X_N$. Then, as $N$ tends to infinity,

$$
\mathbb{P} \left\{ \frac{S_N - E(S_N)}{\sqrt{V(S_N)}} \leq x \right\} \to \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} du, \quad x \in \mathbb{R}.
$$

(3.5)

The law of large numbers tells us that the expectation and the mean are equal in the limit. From the central limit theorem we can deduce that the distribution of the mean is normal, regardless of the initial distribution of $X$. Using the central limit theorem we can prove that Monte Carlo methods does not suffer from the curse of dimensionality, i.e. the convergence rate is independent on the dimensionality, $d$, of the problem. The central limit theorem implies, for large $N$, that

$$
N \times V[\tau_N - \tau] \approx V[\phi(X)],
$$

(3.6)

which implies

$$
D[\tau_N - \tau] \approx \frac{D[\phi(X)]}{\sqrt{N}}.
$$

(3.7)

We see that the rate of convergence $O(N^{-1/2})$ is independent of $d$.

Knowing that the distribution of a Monte Carlo estimator will be normal in the limit is useful when we want to draw conclusion about the variance of the estimator. Since the estimator $\tau_N$ is a consistent estimator\(^7\) we can construct an confidence interval of the estimator:

$$
I_{\tau} = \left[ \tau_N - \frac{\lambda_\alpha \sigma}{\sqrt{N}}, \tau_N + \frac{\lambda_\alpha \sigma}{\sqrt{N}} \right],
$$

(3.8)

where $\lambda_\alpha$ is the usual confidence limits of a standard normal variable with significance $2\alpha$. In practice the variance of $X$ is usually not known. In these cases we can use the approximate confidence interval with $\sigma$ replaced by the empirical variance,

$$
\sigma_N = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \tau_N)^2.
$$

(3.9)

3.1 Variance Reduction

It is obvious from the confidence interval of the estimator $\tau_N$ that if we want a better estimate we just need increase the number of simulations, $N$. We can however use various techniques to reduce the variance of the estimator. The idea behind variance reduction is to find another estimator with the same mean but smaller variance.

\(^7\)An estimator $\hat{\tau}_n$ is called consistent if $\hat{\tau}_n \xrightarrow{p} E[X]$
3.1.1 Antithetic Variates

This method is very simple, but do not always reduce the variance much. As we will see it is possible to reduce the variance without more simulation. First we need to consider how we simulate stock prices. Simulation of Brownian motions (e.g. stock prices) relies on some discretization scheme where the driver of the process is estimated with samples from $\mathcal{N}(0, 1)$. This means that we can express the terminal stock prices as a function of $z$, i.e. $S_T(z_i)$, $i = 1, \ldots, n$. Suppose we have a sample $z_1 = \{z_1, z_2, \ldots, z_n\}$, where each of the $z_i$'s are sampled from the standard normal distribution. Define $z_2$ by

$$z_2 = \{z_1, z_2, \ldots, z_n, -z_1, -z_2, \ldots, -z_n\}.$$  \hspace{1cm} (3.10)

We see that the sample $z_2$ has doubled the sample size of $z_1$ by simply adding the negative terms. Since $\mathcal{N}(0, 1)$ is symmetric around the origin we can consider the sample doubled. Using the sample $z_2$, our new estimator is

$$\tau_{av}^{av} = \frac{\sum \Phi(S_T(z_i)) + \Phi(S_T(-z_i))}{2n}.$$ \hspace{1cm} (3.11)

The expectation of this new estimator is the same as for the old, however the variance is not the same for the two estimators. Note that the sample $\Phi(S_T(z_2))$ is not independent anymore, $z_i$ and $-z_i$ have correlation -1. The variance of the new estimator is

$$V(\tau_{av}^{av}) = \frac{V[\Phi(S_T(z_i))](1 + \rho_{\pm})}{2n}.$$ \hspace{1cm} (3.12)

Now compare this variance with the variance of the basic estimator $\tau_N$,

$$V(\tau_N) = \frac{V[\Phi(S_T(z_i))]}{2n}.$$ \hspace{1cm} (3.13)

We see that we will have a lower variance of our new estimator if the correlation $\rho_{\pm}$ is negative.

3.1.2 Control Variates

The control variates technique is a bit more complicated than antithetic variates, but can lead to great variance reduction if done correctly. Assume we want to estimate the expectation $E[\phi(X)]$. Assume further that we have another random variable $Y$, which we will consider our control variate. The control variate should have the following properties

- $E[Y] = m$ is known,
- $\phi(X) - Y$ can be simulated at the same complexity as $\phi(X)$.

The idea is then to construct a new random variable, for some $b \in \mathbb{R}$,

$$Z = \Phi(X) + b(Y - m).$$ \hspace{1cm} (3.14)
with expectation
\[ E[Z] = E[\phi(X) + b(Y - m)] = \tau. \] (3.15)

We see that the expectation of \( Z \) is consistent with the basic estimator. The variance of \( Z \) is
\[ V[Z] = V[\phi(X) + b(Y - m)] = V[\phi(X)] + 2bC[\phi(X), Y] + b^2V[Y]. \] (3.16)

By differentiating with respect to \( b \) and minimizing yields
\[ b = b^* = -\frac{C[\phi(X), Y]}{V[Y]}. \] (3.17)

Plugging \( b^* \) into (3.16) gives the variance
\[ V[Z] = V[\phi(X)] (1 - \rho^2), \] (3.18)

where \( \rho \) denotes the correlation between \( \phi(X) \) and \( Y \). If we want to reduce the variance we should aim to find a random variable \( Y \) that is highly correlated with \( \rho(X) \) and is easy to simulate.

When applying control variate to the problem of pricing American options, a good candidate for \( Y \) is the price of a European option. The price of the European can be easily obtained by the B&\( S \) formula and is highly correlated with price of an corresponding American option. Glasserman and Broadie [8] uses, among others, European option as control variate to greatly reduce the variance of their mesh estimator.

3.1.3 Importance Sampling

The idea behind importance sampling is to reduce the variance by changing the probability measure from which paths are generated. By changing the probability measure, we can simulate (sample) from the important regions better. Consider the problem of estimating
\[ E(h(X)) = \int h(x)f(x)dx, \] (3.19)

where \( f \) is the density of \( X \). If we now multiply and divide with another density function \( g \) we get
\[ E(h(X)) = \int h(x)\frac{f(x)}{g(x)}g(x)dx = \hat{E}\left[ h(x)\frac{f(x)}{g(x)} \right], \] (3.20)

where \( \hat{E} \) indicates that the expectation is taken with \( X \) distributed according to \( g \). Our new importance sampling estimator is thus given by
\[ \tau^I_N = \frac{1}{N} \sum_{i=1}^N h(X_i)\frac{f(X_i)}{g(X_i)}. \] (3.21)
3.2 Simulation of Stock Prices

The use of Monte Carlo methods relies on that we are able to simulate the underlying process. When we consider the one-dimensional American option with constant interest rate, the underlying process is the stock price. If we assume \( r, \delta \) and \( \sigma \) are constants then the SDE 2.4 can be solved. Let \( S_0 \) be the initial value of the stock, then

\[
S_t = S_0 \exp \left( (r - \delta - \frac{\sigma^2}{2})t + \sigma W_t \right).
\]  

(3.22)

Since the increments of \( W \) are independent and normally distributed, we can simulate paths of the process \( S \) using a recursive procedure

\[
S_t = S_{t-1} \exp \left( (r - \delta - \frac{\sigma^2}{2})dt + \sigma \sqrt{dt} Z \right),
\]  

(3.23)

where we have assumed that time step are equidistant with step size \( dt \) and with \( Z \) distributed according to \( N(0, 1) \). With this recursive procedure we can generate sample paths \( S^i = \{S^i_1, S^i_2, \ldots, S^i_n\}, i = 1, \ldots, N \) using the following algorithm

**Algorithm:** Stock path simulation

1. \( S_0 = s; \)
2. for \( i = 1, \ldots, n \) do
   1. draw \( Z \sim N(0, 1); \)
   2. set \( S_t = S_{t-1} \exp \left( (r - \delta - \frac{\sigma^2}{2})dt + \sigma \sqrt{dt} Z \right); \)
3. end
4 Monte Carlo Methods for Pricing American Options

Most MC methods approaches the pricing of American option with the use of dynamic programming [11]. The theory of the dynamic programming algorithm was invented by Bellman [1] in the 50’s. Bellman described that the theory was created to 

treat the mathematical problems arising from the study of various multi-stage decision processes, ...

Dynamic programming has proven to be a useful tool for many algorithms, not only in financial engineering. How do one apply dynamic dynamic programming to the problem of American option valuation? In the following part of this chapter we are going to answer that question and then give a few examples of algorithms that make use of dynamic programming.

Let $\Pi_i(s)$ denote the value of the option at $t_i$ given $S_i = s$, assuming that the option has not been exercised. Further assume that $\Phi(s)$ is a payoff function of an American-style contract. The goal is to compute the value of $\Pi_0(S_0)$. The value of $\Pi_0(S_0)$ can be obtained recursively by

$$
\Pi_T(s) = \Phi_T(s),
$$

(4.1)

$$
\Pi_i(s) = \max\{\Phi_i(s), C_i(s)\}, i = 0, \ldots, T - 1,
$$

(4.2)

where $C_i(s)$ is the continuation value defined by

$$
C_i(s) = E[\Phi(S_{i+1})|S_i = s].
$$

(4.3)

The dynamic programming formulation thus boils down to at each time step $t = 1, \ldots, T - 1$ compare the payoff with the expected value of the option if not exercised at this time. The challenge is that we do not know the continuation value and must be estimated. There are several proposed ways of how this expectation should be estimated.

4.1 Random Tree Method

Broadie and Glasserman [7] proposed a method were they value the option based on simulating a tree of paths of the underlying Markov chain $S_0, \ldots, S_n$. The tree is constructed by simulating $b \geq 2$ independent children states from each node with the initial node set to $S_0$. From each $S_{j}^0$, $j = 1, \ldots, b$ we simulate $b$ new children states and so on until we reach maturity ($t = T$). Each paths of the Markov chain can be denoted $S_0, S_1^j, S_2^{j_2}, \ldots, S_n^{j_2j_3\cdots j_n}$, where the superscript indicates that the terminal node $S_n^{j_2j_3\cdots j_n}$ is reached by following the $j_1$th branch out of $S_0$, the $j_2$th branch out of the next branch and so on. The construction of a random tree with $b = 2$ (or $b = 3$) children nodes can be seen as a binomial (or trinomial) tree with random placement of each node, whereas the placement of each node in the binomial tree is deterministic.

From the random tree we can calculate high and low estimators at each node by backwards induction. The high estimator is obtained by applying the dynamic programming
algorithm to the tree. At the terminal nodes we calculate the value of the high estimator as follows,

\[ \hat{\Pi}^{j_1 \cdots j_n} = \Phi(S_n^{j_1 \cdots j_n}). \]  

(4.4)

Working backwards, we set the continuation value to the average of the option value,

\[ C^{j_1 \cdots j_n} = \frac{1}{n} \sum_{j=1}^{n} \hat{\Pi}^{j_1 \cdots j_n}_{i+1}, \]  

(4.5)

and the option value at each node is thus given by

\[ \hat{\Pi}^{j_1 \cdots j_i} = \max \left\{ \Phi_i(S_i^{j_1 \cdots j_j}), \frac{1}{n} \sum_{j=1}^{n} \hat{\Pi}^{j_1 \cdots j_j}_{i+1} \right\}, \quad i = 1, \ldots, T - 1. \]  

(4.6)

This estimator is indeed biased high at each node, i.e. \( E[\hat{\Pi}^{j_1 \cdots j_i}|S_i^{j_1 \cdots j_i}] \geq \Pi^{j_1 \cdots j_i} \), which can be showed by applying Jensen’s inequality.

The low estimator is obtained by using the same dynamic programming approach but neglecting the branch leading to the highest value when estimating the continuation value at each node.

As the computational effort needed for generating the tree is increasing exponential with the number of steps \( n \), this method is only practicable with \( n \) relatively small.

4.2 Stochastic Mesh Method

Broadie and Glasserman [8] proposed a method which they gave the appropriate name the Stochastic Mesh method. Their idea was to use a mesh with constant number of nodes at each time step and connecting the nodes from one time step to another by assigning weights \( W \) between the nodes. The benefit of using a mesh rather than a tree is that computational effort will scale linear with the number of time steps \( n \). Two estimators of the option value were suggested, one biased high and one biased low. The high estimator is obtained by dynamic programming with the continuation value set to

\[ C_i(S_{ij}) = \frac{1}{n} \sum_{k=1}^{n} W_{jk}^{i} \hat{\Pi}_{i+1}(X_{i+1,k}), \]  

(4.7)

where \( X_{ij} \) denotes the \( j \)th node at time step \( i \) in the mesh. In the random tree approach we only consider the successors of the current node, but with the continuation value defined as above we make use of all nodes at step \( i+1 \). The weight \( W_{jk}^{i} \) connecting \( X_{ij} \) and \( X_{i+1,k} \) is defined by,

\[ W_{jk}^{i} = \frac{f_{i+1}(X_{ij}, X_{i+1,k})}{g_{i+1}(X_{i+1,k})}. \]  

(4.8)

The reason for defining the weights by this fraction is to overcome the fact that \( S_{i+1} \) given \( S_i = s \) has density \( f_i(x, \cdot) \) while the mesh points \( X_{i+1,j}, \ j = 1, \ldots, n \) were generated
from the density $g_{t+1}$. Broadie and Glasserman suggest the mesh density $g$:

$$g_t(u) = f_0(S_0, u), \quad t = 1, \quad (4.9)$$

$$g_t(u) = \frac{1}{n} \sum_{k=1}^{n} f_{t-1}(S_{t-1,k}, u), \quad t = 2, \ldots, T. \quad (4.10)$$

The advantage of this mesh density is that it will eliminate the risk of exponential growth of variance with increasing number of exercise dates. See Broadie and Glasserman [8] for further discussion. If we want to generate the mesh using the density $g$, we can proceed as follows: generate exactly one successor $X_{ij}$ from the underlying density $f_i(X_{i-1,j}, \cdot)$ from each of the mesh nodes $X_{i-1,j} j = 1, \ldots, n$. If we now choose one of the nodes \{x_{i1}, \ldots, x_{in}\} with uniform probability the value drawn till be distributed according to the mesh density $g_t(u)$.

The low estimator is obtained by simulating a trajectory of the underlying price process $S_t$ until exercised according to the stopping time defined by

$$\hat{\tau} = \min \{t : \Phi_t(S_t) \geq C_t(S_t)\}. \quad (4.11)$$

The low biased path estimator is then given by

$$\hat{q} = \Phi(\hat{\tau}, S_{\hat{\tau}}). \quad (4.12)$$

### 4.3 Regression-Based Method

The regression-based approach was suggested by Tsitsiklis and Van Roy [22] and Longstaff and Schwartz [17] among others. The idea is to use regression (e.g. least-squares) of the option value $\Phi_{i+1}(x)$ to estimate the continuation value $C_{i}(x)$. Let $\psi_r, r = 1, \ldots, M$ be some basis functions (polynomials are often used), then for some constants $\beta_{ir}$ we can express the estimated continuation value by a linear combinations of $\psi$,

$$\hat{C}_i(x) = \sum_{r=1}^{M} \beta_{ir} \psi_r(x). \quad (4.13)$$

Using backwards induction we estimate the option value at each node,

$$\hat{\Pi}_i(S_{ij}) = \max \left\{ \Phi_i(S_{ij}), \hat{C}_i(S_{ij}) \right\}, \quad i = 1, \ldots, n. \quad (4.14)$$

The final estimate is given by

$$\hat{\Pi}_0 = \frac{\hat{\Pi}_1(S_{11}) + \cdots + \hat{\Pi}_1(S_{1n})}{n}. \quad (4.15)$$

Unlike the random tree and stochastic mesh methods, regression-based methods do not converge to the true option value. The accuracy of these methods is determined by the choice of basis functions.
4.4 Duality Method

Up until now we have considered the problem of American option valuation as an maximizing problem. In the duality approach, proposed by Hauh and Kogan [12] and Rogers [19], the valuation problem is represented as a minimizing problem. Let

\( M = \{ M_i, i = 0, \ldots, m \} \) be a martingale with \( M_0 = 0 \), then for any stopping time \( \tau \) taking values in \( \{1, \ldots, m\} \) we have

\[
E[\Phi_\tau(S_\tau)] = E[\Phi_\tau(S_\tau) - M_\tau] \leq E[\max_{k=1, \ldots, m} \{ \Phi_k(S_k) - M_k \}]. \tag{4.16}
\]

This inequality holds for every \( M \), we can therefore choose the martingale that minimizes the expression:

\[
E[\Phi_\tau(S_\tau)] \leq \inf_M E[\max_{k=1, \ldots, m} \{ \Phi_k(S_k) - M_k \}]. \tag{4.17}
\]

This inequality is true for all \( \tau \), it also holds for the supreum over \( \tau \):

\[
\Pi_0 = \sup_\tau E[\Phi_\tau(X_\tau)] \leq \inf_M E[\max_{k=1, \ldots, m} \{ \Phi_k(S_k) - M_k \}]. \tag{4.18}
\]

We define the martingale as a sum of martingale differences,

\[
M_i = \Delta_1 + \cdots + \Delta_i, \quad i = 1, \ldots, m, \tag{4.19}
\]

where

\[
\Delta_i = \Pi_i(S_i) - E[\Pi_i(S_i)|S_{i-1}]. \tag{4.20}
\]

Since we have \( E[\Delta_i|S_{i-1}] = 0 \), \( M \) is indeed an martingale.

4.5 Forward Backwards Method

In this part we will present a novel method for obtaining a high biased estimator of the true price \( Q(0, S_0) \) by the use of dynamic programming and where the continuation value is estimated with a single value, \( \hat{\Pi}(t + 1, B_{i+1}) \). The idea is to construct a grid \( B \) by backwards simulation conditional on the initial value \( S_0 \). With this choice of grid, the value \( \hat{\Pi}(t + 1, B_{i+1}) \) will be an unbiased estimator of the continuation value. The computational effort with this method is quadratic with number of grid points and linear with the number of exercise dates. Assume we have a backward grid \( B = \{ B_1^i, B_2^i, \ldots, B_T^i \}, \quad i = 1, \ldots, n \). The construction of the grid \( B \) will be explained shortly. We apply dynamic programming with

\[
\hat{\Pi}(t, B_t^i) = \begin{cases} 
\Phi(t, B_t^i) & \text{for } t = T, \\
\max \left( \Phi(t, B_t^i), \hat{\Pi}(t + 1, B_{t+1}^i) \right) & \text{for } t < T,
\end{cases} \tag{4.21}
\]

and we set as usual our final estimator to

\[
\hat{\Pi}(0, S_0) = \max \left( \Phi(0, S_0), \frac{1}{n} \sum_{i=1}^n \hat{\Pi}(1, B_1^i) \right), \tag{4.22}
\]

22
This estimator is biased high, which can be proven by using Jensen’s inequality.

If the grid $B$ is generated correctly, the value $\hat{\Pi}(t+1,B_{i,t+1})$ will be an unbiased estimator of the true option value $\Pi_{t+1}$. The construction of the backward grid starts with simulation of $n$ trajectories of the underlying asset: $S = \{S_{i1}^t, S_{i2}^t, \ldots, S_{iT}^t\}$, $i = 1, \ldots, n$. We can consider $S$ as our forward grid. Let $B = \{B_{11}, B_{12}, \ldots, B_{IT}\}$, $i = 1, \ldots, n$ denote the backward grid. The backward grid is constructed by setting

$$B_i^t = \begin{cases} S_T^t & \text{for } t = T \\ S_{i_t}^t & \text{for } t < T \end{cases}$$  \hspace{1cm} (4.23)$$

where $I_t^i$ is drawn from $\{1, 2, \ldots, n\}$ with probability

$$I_t^i \sim \left( \frac{p(S_{t+1} = B_{i+1}^t | S_t = S_t^i)}{\sum_{j=1}^n p(S_{t+1} = B_{i+1}^t | S_t = S_t^j)} \right)^{\ell_t},$$  \hspace{1cm} (4.24)$$

where $p$ is the density of $S$. The reason for choosing this probability is that when we are constructing the backwards grid we want to simulate $B_i^t$ conditional on $S_0 = s$ and $S_{i+1}^t = B_{i+1}^t$, i.e. with probability

$$p(S_t | S_{t+1}, S_0) = \frac{p(S_t, S_{t+1}, S_0)}{p(S_{t+1}, S_0)}$$

$$= \frac{p(S_t, S_{t+1}, S_0)}{\int p(S_t, S_{t+1}, S_0) ds_t}$$

$$= \frac{p(S_{t+1} | S_t, S_0)p(S_t | S_0)p(S_0)}{\int p(S_{t+1} | S_t, S_0)p(S_t | S_0)p(S_0) ds_t}. \hspace{1cm} (4.25)$$

Now if we plug in the empirical measure in the denominator in (4.25) and noting that the process $S$ is discrete Markov chain, equation (4.25) simplifies to

$$= \frac{p(S_{t+1} | S_t)}{\sum_{\ell} p(S_{t+1} | S_{t}^\ell)}. \hspace{1cm} (4.26)$$

Which is exactly the same as probability (4.25).
5 Numerical Results

In this section we are going to present two numerical examples. In the first subsection we compare the Stochastic Mesh method against our novel Forward Backwards method. Some questions we try to answer in the subsection are, \textit{Do the two algorithms converge to the same value? What can be said about the variance of the two algorithms? How do the two algorithms compare in speed?} We apply the two methods on American options (call and put) and omit discounting.

In the second subsection we test the practice viability of the two methods on a one-dimensional American put option. We include the Least Square method since the method performs very well for the one-dimensional case. We study the convergence of the tree methods, accuracy and speed.

5.1 Benchmark test for the Mesh algorithm and the Forward Backwards algorithm

Now we have come to a much interesting part of this thesis where we are going to put our novel Forward Backwards method against the well-established Mesh method. Note that we will not measure the accuracy of the two methods in this subsection. We only compare speed and variance of the two methods.

For the calculations in this section we used a geometric Brownian motion, $X$, with drift $\mu = 0.1$ and diffusion $\sigma = 0.3$, and initial values $X_0 = \{85, 100, 115\}$. An American call option with strike price $K = 100$ was used as derivative. No algorithms enhancements were used. To check the variance of the two methods we use of the mean estimator:

$$\text{Var}(\hat{Q}_M(N)) = \frac{1}{N} \sum_{i=1}^{N} \hat{Q}^{(i)}_M,$$

where $M$ subscripted indicates the Mesh method and $FB$ subscripted indicates the Forward Backwards method.

We have reason to believe that the Backward Forwards method will be significantly faster than the Mesh method. The work for both methods are linear in number of exercise opportunities and quadratic in the number of mesh points, $b$. However, with the Mesh method we have to, for each time step, calculate the transition density, calculate the weights, calculate the continuation value and finally the option value. The corresponding approach for the Forward Backwards method is to first calculate the transition density, choose randomly a grid point, calculate the continuation value and then compute the option value. In the latter approach we have no need to do the computationally demanding calculation of the weights and the calculation of continuation value is done more efficient which should make it faster compared to the Mesh method.

In table 1 we have computational times (in seconds) for the two methods with $b = \{100, 200, 400, 800\}$ grid points. With $b = 100$ grid points the computational time for the Mesh method is 80 times longer than the Forward Backwards method. When the number of grid points, $b$, are doubled, the computational time is increased by about a factor of 4 for the Mesh method and by about a factor of 2 for the Forward Backward
Table 1: Computational Time

<table>
<thead>
<tr>
<th>$b$</th>
<th>Mesh method</th>
<th>Forward Backwards method</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>8.00</td>
<td>0.10</td>
</tr>
<tr>
<td>200</td>
<td>28.46 (256%)</td>
<td>0.37 (270%)</td>
</tr>
<tr>
<td>400</td>
<td>115.04 (304%)</td>
<td>0.82 (122%)</td>
</tr>
<tr>
<td>800</td>
<td>490.55 (327%)</td>
<td>2.07 (152%)</td>
</tr>
</tbody>
</table>

CPU time in seconds. Percentage increase within the brackets.

Figure 1: Distribution of the estimates

(a) Mesh method
(b) Forward Backwards method

The histogram plots were created using 250 replications. The variance of the Mesh estimator replications is 16.97 and the variance of the Forward Backwards estimate replications is 5.69.

method. With $b = 2,000$ the computational effort for the Forward Backwards method is equal to computational effort for the Mesh method with $b = 100$. This allow us to use about 20 times more samples for the same computational effort with the Forward Backward method compared to the Mesh method. Note that computational times depend on computer hardware, implementation of the code and programming language among others. However, the results from table 1 clearly show that the Forward Backwards algorithm is faster and that the computational effort increase with a slower rate when the number of grid points increase compared to the Mesh method. The advantage in speed for the Forward Backwards method over the Mesh method could be expected. A bit more surprisingly, is that the replications of the Forward Backwards method have lower variance than the replications of the Mesh method. The variance of the Mesh mean estimator is almost 3 times greater than the Forward Grid mean estimator:

$$\text{Var}(\overline{Q}_M(N)) = 16.79,$$
$$\text{Var}(\overline{Q}_{FB}(N)) = 5.69,$$
with \( N = 250 \) replications for both estimators.

Now we are going to check if the two methods converge to the same value. To test this we use both American call and put options with three different strike values \( K = \{85, 100, 115\} \). Since the computational effort increases exponential for the Mesh estimator we keep the number of grid points relatively low and instead we use more replications. For the Forward Backward method we us\( e 20 \) times more grid points than the Mesh method for the same computational effort with the same number of replications. Table 2 show results from a one-dimensional asset option of American type with maturity \( T = 1 \) and \( n = 5 \) exercise dates (including time zero). The estimated values, \( \overline{Q}_M(N) \) and \( \overline{Q}_{FB}(N) \), are obtained using \( N = 250 \) replications with \( (b_M, b_{FB}) = (200, 200) \) grid points. Looking at the results from table 2 we make the unpleasant discovery that the two methods differ in the estimated values. For the call option (first three rows) our FB estimator underestimates the Mesh estimator for all initial values of \( X_0 \). For the put option (last three rows) our FB estimator overestimates the Mesh estimator for initial value \( X_0 = 85 \) and underestimates for initial values \( X_0 = \{100, 115\} \).

### 5.2 American option

The following numerical comparison will be done using a one-dimensional American put option as a benchmark. We will estimate the American option with an Bermudan option with five exercise dates (including time \( t = 0 \)). The option specifications are

- Initial value of the stock, \( S_0 = \{38, 40, 42\} \),
- Strike price, \( K = 40 \),
- Interest rate, \( r = 0.06 \),
- Volatility of the stock, \( \sigma = 0.2 \),
• Dividend, \( \delta = 0 \),

• Maturity (in years), \( T = 1 \).

Three methods are compared: the Least Squares method, the Stochastic Mesh method and the Forward Backwards method. No algorithm enhancements was used. For each of the methods we compute the estimate, standard deviation (std) of the estimate, 95%-confidence interval and absolute error. The absolute error is obtained by comparing the estimate with the true values given by the binomial model. The true values given by the binomial model are

\[
\Pi(0, S_0) = [3.260, 2.323, 1.623],
\]

where \( S_0 = \{38, 40, 42\} \). The estimates was obtained using \( N = 250 \) Monte Carlo replications for the Stochastic Mesh method and the Forward Backwards method and \( N = 10 \) for the least square estimates.

<table>
<thead>
<tr>
<th>( S_0 )</th>
<th>( b )</th>
<th>Estimate</th>
<th>Std</th>
<th>Confidence interval</th>
<th>Error</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>32</td>
<td>3.642</td>
<td>0.642</td>
<td>[3.561, 3.723]</td>
<td>0.426</td>
<td>12 s</td>
</tr>
<tr>
<td>38</td>
<td>64</td>
<td>3.501</td>
<td>0.472</td>
<td>[3.452, 3.550]</td>
<td>0.314</td>
<td>90 s</td>
</tr>
<tr>
<td>38</td>
<td>128</td>
<td>3.466</td>
<td>0.299</td>
<td>[3.431, 3.502]</td>
<td>0.240</td>
<td>356 s</td>
</tr>
<tr>
<td>40</td>
<td>32</td>
<td>2.509</td>
<td>0.563</td>
<td>[2.441, 2.576]</td>
<td>0.228</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>64</td>
<td>2.445</td>
<td>0.367</td>
<td>[2.396, 2.494]</td>
<td>0.132</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>128</td>
<td>2.429</td>
<td>0.249</td>
<td>[2.395, 2.462]</td>
<td>0.082</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>32</td>
<td>1.694</td>
<td>0.528</td>
<td>[1.631, 1.756]</td>
<td>0.108</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>64</td>
<td>1.627</td>
<td>0.314</td>
<td>[1.591, 1.663]</td>
<td>0.012</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>128</td>
<td>1.611</td>
<td>0.218</td>
<td>[1.582, 1.640]</td>
<td>0.018</td>
<td></td>
</tr>
</tbody>
</table>

The put option parameters are \( r = 6\% \), \( \delta = 0 \), \( \sigma = 20\% \), \( K = 40 \), with maturity \( T = 1 \) years. Five exercise dates (including \( t = 0 \)) were used. Errors are absolute error with the true option values given by the binomial model are \((3.260, 2.323, 1.623)\). A single estimate was obtained using \( N = 250 \) replications.

From table 3 we see that the mesh estimator is converging to the true value as \( b \) increases. The computational time is increasing exponential as expected. Since the computational is increasing exponential with number of grid points it may be a good idea to keep the number of grid point fairly low and instead use with more Monte Carlo replications.

Looking at table 4 and more closely at the error of the backward grid estimator we can conclude that the estimator is not consistent. Even with twice as many grid points \( (b = 2560) \) the estimator is not converging to the true value. We have tried experimenting with more exercise dates, but with no luck. Adding exercise dates seem to only increase the variance of the estimator.

In table 5 we see that the Least Square estimator is not converging to the real value as number of grid points are increasing. This is expected since the accuracy of
Table 4: Forward Backwards Estimate of American Put Option

<table>
<thead>
<tr>
<th>$S_0$</th>
<th>$b$</th>
<th>Estimate</th>
<th>Std</th>
<th>Confidence interval</th>
<th>Error</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>320</td>
<td>4.569</td>
<td>0.201</td>
<td>[4.529, 4.609]</td>
<td>1.309</td>
<td>1.5 s</td>
</tr>
<tr>
<td>38</td>
<td>640</td>
<td>4.544</td>
<td>0.143</td>
<td>[4.516, 4.572]</td>
<td>1.284</td>
<td>3.8 s</td>
</tr>
<tr>
<td>38</td>
<td>1280</td>
<td>4.559</td>
<td>0.107</td>
<td>[4.537, 4.580]</td>
<td>1.299</td>
<td>11 s</td>
</tr>
<tr>
<td>40</td>
<td>320</td>
<td>2.787</td>
<td>0.190</td>
<td>[2.749, 2.825]</td>
<td>0.464</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>640</td>
<td>2.797</td>
<td>0.169</td>
<td>[2.763, 2.830]</td>
<td>0.473</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1280</td>
<td>2.802</td>
<td>0.108</td>
<td>[2.780, 2.823]</td>
<td>0.478</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>320</td>
<td>1.838</td>
<td>0.169</td>
<td>[1.804, 1.871]</td>
<td>0.214</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>640</td>
<td>1.864</td>
<td>0.128</td>
<td>[1.838, 1.889]</td>
<td>0.240</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1280</td>
<td>1.854</td>
<td>0.098</td>
<td>[1.834, 1.873]</td>
<td>0.230</td>
<td></td>
</tr>
</tbody>
</table>

The put option parameters are $r = 6\%$, $\delta = 0$, $\sigma = 20\%$, $K = 40$, with maturity $T = 1$ years. Five exercise dates (including $t = 0$) were used. Errors are absolute error with the true option values given by the binomial model are (3.260, 2.323, 1.623). A single estimate was obtained using $N = 250$ replications.

This method is determined by the basis functions. For low-dimensional problems Least Squares methods are very efficient. But for higher dimension problems it may be hard to find suitable basis function to obtain good accuracy of the estimator.

6 Conclusions

For low-dimensional problems it is a relatively easy task to obtain the value of an American-style option using e.g. the binomial model or some least squares method. However, for American-style options that depends on multiple assets or on multiple state variables we do not have any obvious way of obtaining the price of the option. The Stochastic Mesh method discussed in this thesis is a solid method for pricing high-dimensional options, but even with recent refinements [16] the computing time grows quadratically with the number of sampled points. Our novel Forward Backwards method has proven to be a very efficient algorithm when compared to the Stochastic Mesh method. The variance of our Forward Backward estimator is lower than the Stochastic Mesh estimator, even tough the computing time for our method is only a fraction of the computing time needed for the Stochastic Mesh method.

The theory of our novel method do say that it is a consistent estimator of the true option value, but the numerical results show that the estimator do not converge to the true option value. This is indeed a disappointing result, however we think that it is well worth to keep trying find a way of making the Forward Backwards estimator converge. If the estimator is consistent, the method will be a highly competent method for, in particular, pricing high-dimensional American-style options and in general approximate the Snell envelope.
The put option parameters are $r = 6\%$, $\delta = 0$, $\sigma = 20\%$, $K = 40$, with maturity $T = 1$ years. Five exercise dates (including $t = 0$) were used. Errors are absolute error with the true option values given by the binomial model are (3.2600, 2.3233, 1.6231). A single estimate was obtained using $N = 10$ replications.

<table>
<thead>
<tr>
<th>$S_0$</th>
<th>$b$</th>
<th>Estimate</th>
<th>Std</th>
<th>Confidence interval</th>
<th>Error</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>38</td>
<td>100</td>
<td>3.680</td>
<td>0.0403</td>
<td>[3.614, 3.746]</td>
<td>0.129</td>
<td>3 s</td>
</tr>
<tr>
<td>38</td>
<td>1,000</td>
<td>3.286</td>
<td>0.0277</td>
<td>[3.240, 3.331]</td>
<td>0.0079</td>
<td>10 s</td>
</tr>
<tr>
<td>38</td>
<td>10,000</td>
<td>3.199</td>
<td>0.0104</td>
<td>[3.183, 3.217]</td>
<td>0.0185</td>
<td>63 s</td>
</tr>
<tr>
<td>40</td>
<td>100</td>
<td>2.830</td>
<td>0.120</td>
<td>[2.632, 3.027]</td>
<td>0.218</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>1,000</td>
<td>2.314</td>
<td>0.0294</td>
<td>[2.266, 2.363]</td>
<td>0.0039</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>10,000</td>
<td>2.282</td>
<td>0.0094</td>
<td>[2.266, 2.297]</td>
<td>0.0180</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>100</td>
<td>1.984</td>
<td>0.0630</td>
<td>[1.880, 2.087]</td>
<td>0.222</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>1,000</td>
<td>1.634</td>
<td>0.0242</td>
<td>[1.594, 1.674]</td>
<td>0.0065</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>10,000</td>
<td>1.586</td>
<td>0.0063</td>
<td>[1.576, 1.597]</td>
<td>0.0227</td>
<td></td>
</tr>
</tbody>
</table>
References


Approximation of Option Prices Using Computer Simulations

A challenging task in the world of financial mathematics is to price American options. An American option is a derivative which derives its value from an underlying asset, e.g. a stock. The holder of an American stock option has the choice to either sell or buy a certain quantity of a company’s stock by a certain date for a certain price. The price in the contract is known as the strike price and the date is called expiration date or maturity. A put option gives the holder right to sell the underlying asset, and a call option gives the holder right to buy the underlying asset.

Unlike the European option, that may only be exercised at maturity, an American option may be exercised at any time until the expiration of the option. This additional feature of American options make them much more complicated to price compared to European options. The price of American options must be approximated, whereas the price of European options easily can be obtained analytically.

Once a day (if we assume that the owner of the option checks the market once a day) the owner of an American option has to make the decision to either keep the option at least one more day or to exercise the option and receive the payoff. The payoff is determined by the payoff function. The payoff function is \( \Phi(S_{\tau}) = \max(0, S_{\tau} - K) \) for the call option and \( \Phi(S_{\tau}) = \max(0, K - S_{\tau}) \) for the put option, where \( K \) is the strike price and \( S_{\tau} \) is the stock price.

The owner should of course make the decision which he or she believes will be the most profitable. In order to make the right decision the owner must compare today’s payoff with the expected payoff received if exercised tomorrow. When we, by simulation, try to price an American option we are basically doing the same thing. The problem is that we do not know tomorrow’s expected payoff - we must estimate it. There are several suggested approaches on how this expectation, often referred to as the continuation value, should be estimated. Overall, the main difference between the methods is how the continuation value is estimated. In this thesis I have tried to cover as many methods as possible.

Apart from the established methods for pricing American options, we also present a novel forward backwards simulation-based method for solving the task. The method relies on Monte Carlo simulation of the stock price movements. In short, Monte Carlo methods are when one tries to solve a problem by simulating the underlying process and then calculate the average of some function of the simulated process. In our setup, the underlying process is of course the price movements of the stock and the function is the continuation value.

According to the theory behind our method the estimator should be high biased and converge to the true option price. Unfortunately, the numerical results show that our estimator do not converge to the true option value. This was an unexpected and disappointing result. However, the benchmark test show that our method is really fast. If one could figure out a way to make the estimator converge, the method would be a highly competent alternative for pricing American options.