Monte Carlo Simulation in Option Pricing

Long Yun

(B.Sc. Peking University)

A THESIS SUBMITTED

FOR THE DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF STATISTICS AND APPLIED PROBABILITY

NATIONAL UNIVERSITY OF SINGAPORE

2010

Acknowledgements

I would like to take this opportunity to express my sincere gratitude to everyone who has provided me their support, advice and guidance throughout this thesis.

First of all, I would like to thank my supervisor, Assoc. Professor Xia Yingcun, for his guidance and assistance during my two-year graduate study and research. His ideas and expertise are crucial to the completion of this thesis. I would like to thank him for teaching me how to undertake researches and spending his valuable time revising this thesis.

I would also like to express my heartfelt gratitude to my girlfriend Zhao Yingjiao for her support and help in revising this thesis. Then I want to thank my friend Jiang Qian, Tran Ngoc Hieu, Lu Jun, Luo Shan, Liang Xuehua and my former colleagues Mohamed Lemsitef at Merrill Lynch Hong Kong, Zhu Yonglan at Barclays Capital for their help in completing this thesis.

Contents

Abstract

Along with the rapid development of derivatives market in the last several decades, option pricing technique becomes an extremely popular area in academic research, since Black, Scholes and Merton (1973) developed the first option pricing formula. A number of numerical methods can be applied in option valuation. However, they may encounter some difficulties when pricing relatively complicated options like path-dependent or American-style ones, which are quite common in the financial industry. In this thesis, the Least Squares Monte Carlo (LSM) approach to American option valuation by Longstaff and Schwartz (2001) is introduced. Moreover, the mathematical foundation, e.g. the convergence and the robustness of the simulation is provided. Furthermore, we improve this approach by applying the Quasi Monte Carlo, which can enhance the effectiveness, accuracy and computational speed of the simulation. The numerical results show that the improved algorithm works well in pricing American options and outperforms the original one in both effectiveness and accuracy. We have also discussed about the trade-off between the computational time and the precision of the price regarding number of

paths in simulation, number of possible exercise time points and different degrees of polynomials in the regression process.

Keywords: Option Pricing, American Options, Least Squares Monte Carlo

List of Tables

List of Figures

Chapter 1

Introduction

Option becomes a popular traded financial products both in Exchange and Over-the-counter (OTC) markets in the last four decades. There are also many other types of derivatives which are imbedded with an option or have the similar characteristics with options. It is well known that the value of an option generally depends on the strike price, the price of the underlying asset, the volatility of the underlying, dividends, interest rate and time to maturity. Though it is important for the traders to get the theoretical price of the option they trade, it still remains a challenge to price some types of option in the market, which drives option pricing technique as one of the most popular areas in both academic research and financial industry.

1.1 Literature Review on Monte Carlo Methods for Option Pricing

The most celebrated work in the research field of option pricing belongs to Fisher Black and Myron Scholes (1973), and Robert Merton (1973). Scholes and Merton were awarded the Nobel Prize for Economics in 1997 for their landmark contributions. We will provide more detailed information about their work, or Black-Scholes model, in Chapter 2.

Black-Scholes model for European option is one of the very few cases where the closed-form expressions for derivative prices exist. Analytical expressions for American options have been found in several simple cases as well, e.g. the formulas for American call options with discrete dividends provided by Mckean (1965), Roll (1977), Geske (1979), and Whaley (1981). However, in most cases, there is no analytical solution even in the simple framework of Black-Scholes model. In reality, this is a big problem as most options traded in the Chicago Board of Options Exchange (CBOE) are American ones.

Alternatively, one has to apply to numerical solutions to price these American options. The most famous numerical solutions for American options is the binomial model suggested by Cox, Ross, and Rubinstein (1979). Though binomial model is widely used in financial industry, a major problem with it as well as some other numerical methods is that the price of the underlying asset is the only stochastic factor involved in these models, while other determining factors are assumed to be constants. However, as we all know that this assumption does not hold, at least for the volatility smile, interest rate and dividends. Meanwhile, when it is used to handle several stochastic factors, binomial model becomes computationally infeasible because the number of binomial nodes in the model grows exponentially with number of factors, which is known as the curse of dimensionality. Therefore, this model is not flexible enough when dealing with multiple stochastic factors e.g. changing volatility, interest rate, dividend, or multiple underlying assets.

Another useful numerical method is Monte Carlo simulation, which was introduced to pricing option firstly by Boyle (1977). Unlike binomial trees, simulation technique is proven to be applicable in situations with multiple stochastic factors(Barraquand (1995)) and has been used to price European options for quite a long time. However, not until very recently, it is generally considered impossible to use simulation to price American options from a computational perspective(Campbell, Lo, and MacKinlay (1996) and Hull(1997)). The reason is that when pricing American options, one has to calculate the optimal early exercise policy recursively. This process would lead to biased results using simulation as there is only one future path any time time along. One of the early studies that try to propose solutions to price American options using simulation was conducted by Tilley in 1993. He suggested a simulation algorithm that mimics the standard lattice to determine the optimal early exercise strategy. Similarly, Barraquand and Martineau (1995) developed a method called Stratified State Aggregation along the

Payoff. Broadie and Glasserman (1997) also tried to use Monte Carlo Simulation in American option pricing, but their approach is more related the binomial model in essence.

Another approach to determine the optimal stopping times along the paths was proposed by Carriere (1996). He showed that pricing American options is equivalent to calculating numbers of conditional expectations using a backwards induction theorem. And it was possible to approximate the conditional expectations by combining simulation with advanced regression methods. Inspired by Carriere (1996), and Tsisiklis and Van Roy (1999), Longstaff and Schwartz (2001) put forth a simulation-based method so called Least Square Monte Carlo Algorithm (LSM) in a simpler way. They estimated the conditional expectations by letting the option alive at every exercise point from a simple least squares cross-sectional regression. They also show how to price different types of path dependent options, such as American put options, American-Bermuda-Asian options, cancelable index amortizing swaps, by using LSM. We will introduce the details of LSM and try to improve this technique in Chapter 4 of this thesis.

1.2 Organization of this Thesis

In this thesis, we will mainly focus on how to apply Monte Carlo simulation to price options especially American ones.

In the introduction, we have made a review on the related literature for option pricing, numerical methods for option pricing, and especially how to price American options by simulation. We can get a basic idea about this research area and how these ideas were generated and developed.

In the main chapter, we will start with the introduction of the finance background, in which we can find the definition of commonly used financial terms in this thesis and the characteristics of different options. We then turn our attention to the most famous model for option pricing - Black-Scholes Model. The assumption and basic idea under this model will be illustrated, and the pricing formula for European option is obtained after some PDE deriving work. While in practice, most option pricing is done by applying numerical methods or combining them with Black-Scholes model, we will introduce three basic numerical methods including binomial trees, finite difference and Monte Carlo simulation. After introducing how these numerical methods work, we compare the advantages and drawbacks of them.

In the next chapter, we will show how to price European options using Monte Carlo simulation. The framework of this method is discussed first. Although most softwares provide random number from normal distribution $N(0, 1)$ as a black-box function, we still give a brief introduction of generating pseudo-random numbers including the probability proof of converting random numbers sampling from $U[0, 1]$ to that from normal distribution $N(0, 1)$. After implementing the program, we will compare the results approximated from Monte Carlo simulation to the theoretical value, which is implied by Black-Scholes model.

Next, we will discuss how to apply Monte Carlo simulation to price American-style option. The motivation of pricing American option is discussed and the difficulty of pricing more complicated options using other numerical methods is explained. For example, they may encounter some difficulties when pricing relatively complicated options such as path-dependent, American-style or multiple stochastic factors, which are quite common in the financial markets. By introducing Longstaff and Schwartz's Least Square Monte Carlo Algorithm (LSM), we can apply Monte Carlo simulation to value American-style option successfully. Moreover, the mathematical foundation such as the convergence and the robustness of the simulation is provided. To check the feasibility and accuracy of LSM, we select several American options and compare the results from LSM to those from finite difference method, which is considered quite accurate and is widely used for pricing plain American option in industry. We will also pick up several options traded in CBOE and try to compare the market price with price calculated from LSM. As generally Monte Carlo simulation is quite time consuming, which may limit its usage in pricing complicated derivatives, we try to improve this approach by applying the quasi Monte Carlo, which can enhance the effectiveness, accuracy and computational speed. We will also discuss about the trade-off between the computational time and the precision of the price regarding numbers of paths in simulation, different basic functions in the regression process and numbers of possible exercise time points.

In the conclusion, major findings and contribution of this thesis will be presented and some ideas for future research will be proposed.

Chapter 2

Foundation

2.1 Finance Background

We will go through the main financial terms and concepts used in this thesis. A *derivative* can be defined as a financial instrument whose value depends on (or derives from) the values of other, more basic, underlying variables. Common types of derivatives securities are options, futures, forwards and swaps. We will focus on the options, as most of the methods developed here can be applied for other kinds of derivatives products too.

As mentioned by Hull (2006), an *option* is a derivative which gives the holder the right, but not the obligation to engage in some future transaction involving the underlying. A *call option* gives the holder the right to buy the underlying asset by a certain date for a certain price. A *put option* gives the holder the right to sell the underlying asset by a certain date for a certain price. The price in the contract is known as the *exercise price* or *strike price*; the date in the contract is known as the *expiration date* or *maturity*. There are two basic kinds of options. *European options* can be exercised only at the expiration date. *American options* can be exercised at any time up to the expiration date.

The value of an option contract generally depends on several parameters including strike price, the value of the underlying asset, the volatility of this asset, the amount of dividends paid on it, the interest rate and the time to maturity. The intuitive thinking indicates that the value of a call (put) option decreases (increases) as the strike price increase. The value of a call (put) option increases (decreases) as the underlying asset price increase. The value of any option increases as the volatility of the underlying asset increases. The value of a call (put) option increases (decreases) as the risk-free interest rate increase. The value of any option is also a function of the time to expiration, normally decreases as time decays. A summary of these factors' effect on option value is shown in the table 2.1 (Hull (2006)):

Table 2.1: Effect to option price when increase one variable

Variable	European call	European put	American call	American put
Current stock price	+	-	+	-
Strike price	-	+	-	+
Time to expiration	?	?	+	+
Volatility	+	+	+	+
Risk-free rate	+	-	+	-
Dividends	-	+	-	+

As they can only be exercised at the expiration date instead of any time, European options are generally easier to analyze than American options, and some of the properties of American options can be deduced from those of its European counterpart. Intuitively, an American option is always at least as valuable as the corresponding European option, since the holder of American option could always choose to act in the same manner as the holder of the European option by simply holding the option until the expiration date, when it becomes exactly the same as European option.

Arbitrage is a trading strategy that takes advantage of two or more securities being mispriced relative to each other. In other words, arbitrage involves locking in a riskless profit by simultaneously entering into transactions in two or more markets. *Hedging* is a trading strategy that involves reducing the exposure to risk associated with holding one asset by holding other assets whose returns are usually correlated with the first. And under a compounded interest rate r, 1 unit investment today which earns continuous interest will worth e^{rt} after t years. In a landmark contribution to the field of option pricing, Black and Scholes (1973) developed a closed form solution for the price of European options under certain conditions. Their approach relies on the *No-Arbitrage Assumption*, which comes from observing that the return on an option can be perfectly replicated by continuously rebalancing a hedged portfolio consisting of shares of underlying asset and a risk free asset which earns continuously compounded interest at a rate of r. In the following section 2.2, we will give a more detailed introduction to the Black-Scholes model.

As a major extension of Black-Scholes model, Merton (1973) showed that if the underlying stock pays no dividends, the value of the American call is the same as the value of the European call yielded by the Black-Scholes model, as it is never optimal to exercise an American call option early. Except for this case, closed form solutions for pricing American options are rarely available. Moreover, since Black-Scholes Model is obtained under very strict assumptions, which may not be appropriate in real market, and loosing these assumptions normally leads to closed form solutions unavailable, the practicers must implement numerical methods to find the approximate values instead of the theoretical ones. This is why numerical techniques is widely employed in the financial industry and the academic research for numerical option pricing is so popular in recent years. In the following section

2.3, we will give a brief introduction to some basic numerical methods for option pricing.

2.2 Black-Scholes Model

In the early 1970s, Fischer Black, Myron Scholes, and Robert Merton made a major breakthrough in the field of option pricing, for which Myron Scholes, and Robert Merton were awarded Nobel prize for economics in 1997. Though ineligible for the prize because of his death in 1995, Black was mentioned as a contributor by the Swedish academy. Their work, which is known as Black-Scholes (or Black-Scholes-Merton) Model, has had a significant influence on the way that traders price and hedge options. It also leads the growth and success of financial engineering in the last 30 years. In this section, we will show the framework of the Black-Scholes Model, and how to derive the model for valuing European call and put option on a non-dividend-paying stock.

There are several explicit assumptions for deriving the Black-Scholes model:

- It is possible to borrow and lend cash at a known constant risk-free interest rate r.
- The price of the underlying follows a geometric Brownian motion with constant drift and volatility.
- There are no transaction costs.

- The stock pays no dividend.
- All securities are perfectly divisible (i.e. it is possible to buy any fraction of a share).
- There are no restrictions on short selling.
- There are no riskless arbitrage opportunities.
- Security trading is continuous.

Some of these assumptions can be relaxed and the Black-Scholes Model can be extended.

The main idea in the development of the model is that the return on an option can be perfectly replicated by continuously rebalancing a hedged portfolio consisting of shares of underlying asset and a risk free asset such as a government bond. As the return of this hedge portfolio is independent of the price movement of the stock, it only depends on the time and other known constant variants. This deterministic return cannot be greater than the return on the initial investment compounded at the risk free interest rate. Otherwise there exist arbitrage opportunities by borrowing at the risk free rate, using which to establish a position in the higher yielding hedge portfolio, which would in turn force the yield to the equilibrium risk free rate.

Next we will derive the Black-Scholes pricing formulas. Firstly, we will define some notations used in this section. We define:

S, the price of the stock

f, the price of a derivative as a function of time and stock price.

c, the price of a European call.

p the price of a European put option.

K, the strike of the option.

r, the annualized risk-free interest rate, continuously compounded.

μ, the drift rate of S, annualized.

σ, the volatility of the stock; this is the square root of the quadratic variation of the stock's price process.

t, a time in years; we generally use now = 0, expiry = T.

Π, the value of a portfolio.

R, the accumulated profit or loss following a delta-hedging trading strategy.

$N(x)$, denotes for the standard normal cumulative distribution function, $\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{x} e^{-\frac{z^2}{2}}dz$

$N'(z)$, denotes for the standard normal probability density function, $\frac{e^{-\frac{z^2}{2}}}{\sqrt{2\pi}}$

Wiener process, or sometimes referred to as Brownian motion, is a particular type of Markov stochastic process with a mean change of zero and a variance rate of 1.0 per year. A variable z follows a Wiener process if it has the following two properties:

Property 1. The change Δz during a small period of time Δt is

$$\Delta z = \epsilon\sqrt{\Delta t} \tag{2.1}$$

where ϵ is a standardized normal distribution $\phi(0,1)$.

Property 2. The value of Δz for any two different short intervals of time Δt is independent.

A generalized Wiener process for a variable x can be defined in terms of dz as

$$dx = adt + bdz \tag{2.2}$$

where a and b are constants.

A further type of stochastic process, known as an Itó process, is a generalized Wiener process in which the parameter a and b are functions of the value of x and t. i.e.

$$dx = a(x,t)dt + b(x,t)dz \tag{2.3}$$

Suppose a variable x follows the Itó process. Itó's lemma, which was discovered by the mathematician K. Itó in 1951, shows that a function G of x and t follows the process

$$dG = (\frac{\partial G}{\partial x}a + \frac{\partial G}{\partial t} + \frac{1}{2}\frac{\partial^2 G}{\partial x^2}b^2)dt + \frac{\partial G}{\partial x}bdz \tag{2.4}$$

As per the model assumptions, the stock price process follows a geometric Brownian motion. That is,

$$dS = \mu Sdt + \sigma Sdz \tag{2.5}$$

Suppose that f is the price of a call option or other derivative contingent on S. f must be some function of S and t. From Itó's lemma in (2.4), we have

$$df = (\frac{\partial f}{\partial S}\mu S + \frac{\partial f}{\partial t} + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\sigma^2 S^2)dt + \frac{\partial f}{\partial S}\sigma Sdz \tag{2.6}$$

The discrete versions of equations (2.5) and (2.6) are

$$\Delta S = \mu S \Delta t + \sigma S \Delta z \tag{2.7}$$

and

$$\Delta f = (\frac{\partial f}{\partial S}\mu S + \frac{\partial f}{\partial t} + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\sigma^2 S^2)\Delta t + \frac{\partial f}{\partial S}\sigma S \Delta z \tag{2.8}$$

where ΔS and Δf are the changes in S and f in a small time interval Δt. As S and f has the same underlying, the Wiener processes of them should be the same. In other words, the $\Delta z(=\epsilon\sqrt{\Delta t})$ in equations (2.7) and (2.8) are the same. Therefore, by choosing a portfolio of the stock S and the derivative f, the Wiener process can be eliminated.

The appropriate portfolio is

-1: derivative

$\frac{\partial f}{\partial S}$: shares

which means the portfolio is short 1 derivative and long $\frac{\partial f}{\partial S}$ shares of stock. Define Π as the value of this portfolio. We have

$$\Pi = -f + \frac{\partial f}{\partial S}S \tag{2.9}$$

The change $\Delta\Pi$ in the value of the portfolio in the time interval Δt is given by

$$\Delta\Pi = -\Delta f + \frac{\partial f}{\partial S}\Delta S \tag{2.10}$$

Substituting equations (2.7) and (2.8) into equation (2.10) yields

$$\Delta\Pi = (-\frac{\partial f}{\partial t} - \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\sigma^2 S^2)\Delta t \tag{2.11}$$

In this equation, it shows that the change $\Delta\Pi$ in the value of the portfolio in the time interval Δt does not related to the stochastic process Δz, which means this portfolio is riskless during the time Δt. By the No-Arbitrage Assumption, the portfolio must instantaneously earn the same rate of return as the risk-free securities. i.e.

$$\Delta\Pi = r\Pi\Delta t \tag{2.12}$$

where r is the risk-free interest rate. By substituting equations (2.8) and (2.11) into equation (2.12), we obtain

$$(\frac{\partial f}{\partial t} + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\sigma^2 S^2)\Delta t = r(f - \frac{\partial f}{\partial S}S)\Delta t$$

which is equivalent to

$$\frac{\partial f}{\partial t} + rS\frac{\partial f}{\partial S} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 f}{\partial S^2} = rf \tag{2.13}$$

We now show how to get the general Black-Scholes partial differential equations (PDE) to a specific valuation for an option. For the European call option c, we have the boundary conditions:

$c(0,t) = 0$ for all t

$c(S,t) \to S$ as $S \to \infty$

$c(S,T) =$max$(S-K,0)$

For the European put option p, we have similar boundary conditions too. After transforming the Black-Scholes PDE into a diffusion equation, we can solve the equation using standard methods. Thus we can get the value of the options as follows:

$$c = S_0N(d_1) - Ke^{-rT}N(d_2) \tag{2.14}$$

and

$$p = Ke^{-rT}N(-d_2) - S_0N(-d_1) \tag{2.15}$$

where

$$d_1 = \frac{ln(S_0/K + (r + \sigma^2/2)T}{\sigma\sqrt{T}}$$

$$d_2 = \frac{ln(S_0/K + (r - \sigma^2/2)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}$$

2.3 Basic Numerical Methods for Option Pricing

As is shown, Black-Scholes model is obtained under very strict assumptions, which may not be fully satisfied in the real market. There are also many extensions for Black-Scholes model by changing or loosing these assumptions. However, most of the extensions normally lead to closed form solutions unavailable. In this case, the practicers can implement numerical methods to find the approximate values instead of the theoretical ones. In this section, we will give a brief introduction to some basic numerical methods for option pricing, e.g. binomial trees, finite difference and Monte Carlo simulation. Generally, these different numerical methods have different application areas. Monte Carlo simulation is usually applied for derivatives where the payoff is dependent on the history of the underlying variable or where there are several underlying variables. Binomial trees and finite difference are usually used for American options and other derivatives in which the holder has the right to make early exercise decisions prior to maturity. In practice, these methods are able to handle most of the derivatives pricing. However, sometimes they have to be adapted to cope with particular situations. We will introduce these basic numerical methods for option pricing one by one.

2.3.1 Binomial Trees

Binomial trees derivatives pricing model was originally presented by Cox, Ross, and Rubinstein in 1979. As we have assumed, the underlying price follows a random walk. The binomial tree technique is a diagram representing different possible paths of the stock price over the life to maturity. In each time step, it has a certain probability of moving up in a certain percentage amount and a certain probability of moving down in a certain percentage amount. By taking smaller and smaller time step, the limit of the binomial tree leads to the lognormal assumption for stock price, the same as we assume in Black-Scholes model.

Consider a stock worth S_0 at time 0. At the end of the period, the price of the stock is uS_0 with the probability p or dS_0 with the probability $1-p$, where $u > 1 > d$.

Let f represent the current value of a call option on the stock which expires at the end of the period, having a strike price of K. We know that $f_u = \max(0, uS_0 - K)$ and $f_d = \max(0, dS_0 - K)$.

Suppose we have a portfolio consisting of a long position in Δ shares and a short position in one option. We will try to select Δ that makes this portfolio riskless. If the price moves up, the value of the portfolio is

$$S_0 u\Delta - f_u$$

If the price moves down, the value of the portfolio is

$$S_0 d\Delta - f_d$$

They are equal when

$$S_0 u\Delta - f_u = S_0 d\Delta - f_d$$

which implies,

$$\Delta = \frac{f_u - f_d}{S_0 u - S_0 d} \tag{2.16}$$

In this case, no matter the stock price moves up or down, the value of the portfolio is the same, which means, the portfolio is riskless. By the No-Arbitrage Assumption, the portfolio must instantaneously earn the risk-free rate r. i.e.

$$S_0\Delta - f = (S_0 u\Delta - f_u)e^{-rT}$$

or

$$f = S_0\Delta(1 - ue^{-rT}) + f_u e^{-rT} \tag{2.17}$$

Substituting equation (2.16) for Δ and simplifying, we can get f as

$$f = e^{-rT}[\tilde{p} f_u + (1 - \tilde{p}) f_d] \tag{2.18}$$

where

$$\tilde{p} = \frac{e^{rT} - d}{u - d} \tag{2.19}$$

From the illustration above, we know that the option pricing formula in (2.18) does not involve the probabilities of moving up or down in stock price. This characteristic also implies when we increase the steps to obtain a more accurate approximation to the real stock pricing moving. In practice, it is typically divided into 30 or more time steps. In all, there are 31 terminal stock prices and 2^{30}, or about 1 billion, possible stock price moving paths are considered. This can get a satisfying approximation for the value of the derivative. There are also many extensions for this basic model, such as the one developed by Hull and White (1990) by considering the dividend paying and multivariate valuation problems.

2.3.2 Finite Difference

Schwartz (1977) is the first to apply the finite difference technique to price options when closed form solutions are unavailable. Specially, he considered an American option on a stock which pays discrete dividends, which is quite common in real world. This method provides a practical numerical solution to the option pricing problem, and the optimal early exercise strategy as well.

Unlike the ideal case in Black-Scholes model, in practice, we need to consider the valuation of an option which pays discrete dividends and also allow for early exercise (American-style option). In this case, the partial differential equation (PDE) which determines the value of the options is the same as in the Black-Scholes Model, but the boundary conditions will be different due to the early exercise feature and the payment of dividend. More specifically, let f represent the value of an American call option on a Stock with the price of S, and expires at time T. The PDE is (equivalent to equation (2.13))

$$\frac{\partial f}{\partial t} = rf - rS\frac{\partial f}{\partial S} - \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 f}{\partial S^2} \tag{2.20}$$

subject to the boundary conditions:

$$\begin{aligned} & f(0,T) = 0 \\ & f(S,T) \to S, \text{ as } S \to \infty \\ & f(S,T) = \max(S-K,0) \\ & f(S,T^+) = \max(0, S-K, f(S-D,T^-)) \end{aligned} \tag{2.21}$$

where T^+ and T^- are the instants just before and just after the sock pays the discrete dividend D. The last condition reflects the fact that the stock price drops by D as the dividend is paid, and indicates that it is optimal to exercise the option just before the dividend is paid whenever the value $S-K$ is greater than the option value $f(S-D,T^-)$ right after the dividend is paid. Unfortunately, the equation (2.20) with the boundary conditions (2.21) has no closed form solution, but can be solved numerically by approximating the partial derivatives with finite differences.

We can estimate the derivative $\frac{\partial f}{\partial S}$ at the point (S,t) by

$$\frac{[f(S+\Delta,t) - f(S,t)] + [f(S,t) - f(S-\Delta,t)]}{2\Delta}$$

where Δ is a small change of S. The pricing algorithm approximates the partial derivatives at a lattice within the domains of price and time. For example, consider $n+1$ discrete values

$$S_i = ih, i = 0, \ldots, n$$

in the domain of S, and $m+1$ discrete values

$$t_j = jk, j = 0, \ldots, m$$

in the domain of time. After introducing the notation $f_{i,j} = f(S_i, t_j)$, we have

$$\frac{\partial f}{\partial S} = \frac{f_{i+1,j} - f_{i-1,j}}{2h} \tag{2.22}$$

Similarly,

$$\frac{\partial^2 f}{\partial S^2} = \frac{f_{i+1,j} - 2f_{i,j} + f_{i-1,j}}{h^2} \tag{2.23}$$

$$\frac{\partial f}{\partial t} = \frac{f_{i,j} - f_{i,j-1}}{k} \tag{2.24}$$

Substituting equations (2.22), (2.23) and (2.24) into (2.21) yields

$$\frac{f_{i,j} - f_{i,j-1}}{k} = rf_{i,j} - \frac{ri(f_{i+1,j} - f_{i-1,j})}{2} - \frac{\sigma^2 i^2 (f_{i+1,j} - 2f_{i,j} + f_{i-1,j})}{2} \tag{2.25}$$

Rearranging terms yields the system of $n-1$ equations in $n+1$ unknowns:

$$a_{1,i} f_{i-1,j} + a_{w,i} f_{i,j} + a_{3,i} f_{i+1,j}, for \quad i = 1, \ldots, n-1 \quad and \quad j = 0, \ldots, m \tag{2.26}$$

where $a_{1,i} = \frac{1}{2}rki - \frac{1}{2}\sigma^2 ki^2$, $a_{2,i} = (1 + rk) + \sigma^2 ki^2$ and $a_{3,i} = -\frac{1}{2}rki - \frac{1}{2}\sigma^2 ki^2$

We can also write the boundary as below

$$\begin{aligned} f_{0,j} &= 0, j = 0, \ldots, m \\ f_{n,j} - f_{n-1,j} &= h, j = 0, \ldots, m \end{aligned} \tag{2.27}$$

Equations (2.26) and (2.27) provide a solution for $f_{i,j}$ in terms of $f_{i,j-1}$. As $f_{i,0}$ is known, the entire series of $f_{i,j}$ can be generated by the iterating procedure, and any desired degree of accuracy can be obtained by choosing h and k small enough with the cost of computational time. As an extension, Hull and White (1990) suggested a modification to this algorithm to ensure its convergence to the true values and also extend it to deal with multivariate valuation problems.

2.3.3 Monte Carlo Simulation

Monte Carlo Simulation is very useful in calculating the value of an option with multiple factors of uncertainty or with complicated features. The term 'Monte Carlo method' was coined by Stanislaw Ulam in the 1940's and first applied in pricing European option by Phelim Boyle in 1977.

To understand the basic idea of the Monte Carlo simulation, we consider the problem of calculating the integral

$$\int_A g(x)f(x)dx = \bar{g} \tag{2.28}$$

where A is the integration range, $g(x)$ is an arbitrary function, and $f(x)$ is a probability density function. We can get a Monte Carlo estimation of $\bar{g}$ by generating an $i.i.d$ sample $\{x_1, \ldots, x_n\}$ from $f(x)$ and calculating

$$\widehat{g} = \frac{1}{n}\Sigma_{i=1}^{n} g(x_i) \tag{2.29}$$

Variance of $\widehat{g}$ is estimated by

$$\widehat{s}^2 = \frac{1}{n-1}\Sigma_{i=1}^{n}(g(x_i) - \widehat{g})^2 \tag{2.30}$$

We can compute confidence intervals using the fact that

$$\frac{\widehat{g} - \bar{g}}{\sqrt{\widehat{s}^2/n}} \to N(0,1) \tag{2.31}$$

when $n \to \infty$.

Boyle (1977) applied this reasoning to pricing a European option on a stock which follows a geometric Brownian motion. Recall from the assumption in equation (2.5) that

$$dS = \mu S dt + \sigma S dz \tag{2.32}$$

where dz is a Wiener process. To simulate the path of S, we can divide the life of the option into N short time intervals Δt and approximate the equation (2.32) by

$$S(t + \Delta t) - S(t) = \mu S(t) + \sigma S(t)\epsilon\sqrt{\Delta t} \tag{2.33}$$

where ϵ is a random sample from a normal distribution $N(0,1)$. After repeating this procedure we can construct a path for S by using N random samples from $N(0,1)$.

But in practice, usually it is more accurate to calculate $\ln S$ rather than S. Instead, we know that $\ln S$ follows the stochastic process below by Itó's lemma,

$$d\ln S = (\mu - \frac{\sigma^2}{2})dt + \sigma dz \tag{2.34}$$

For short time intervals Δt, we have

$$\ln S(t+\Delta t) - \ln S(t) = (\mu - \frac{\sigma^2}{2})\Delta t + \sigma\epsilon\sqrt{\Delta t} \tag{2.35}$$

From equation (2.35), it is easy to show that, for all T, we have

$$\ln S(T) - \ln S(0) = (\mu - \frac{\sigma^2}{2})T + \sigma\epsilon\sqrt{T} \tag{2.36}$$

or equivalently

$$S(T) = S(0)\exp[(\mu - \frac{\sigma^2}{2})T + \sigma\epsilon\sqrt{T}] \tag{2.37}$$

Equation (2.37) provides an straightforward way to estimate the value of the stock price at any time T and thus construct the path for the stock price.

One drawback of this kind of estimation is that the standard error is inversely proportional to $\sqrt{n}$. To overcome which, Boyle (1977) also introduces control variates and antithetic variates to improve the efficiency of the simulation. The key advantage of Monte Carlo simulation is that it can also handle the case when the payoff depends on both of the path followed by the underlying S, when the other two numerical methods may have some trouble in applying. One example of this advantage is that M. Broadie and P. Glasserman (1996) showed how to price Asian option by Monte Carlo simulation. Other major drawbacks of Monte Carlo simulation include that it is sometimes computationally time consuming and is hard to handle American-style options. In Chapter 4 of this thesis, we will discuss how to overcome these drawbacks and improve the useful Monte Carlo simulation.

Chapter 3

Monte Carlo Simulation for Pricing European Options

3.1 Framework

As discussed in Section (2.3.3), we have

$$d\ln S = (\mu - \frac{\sigma^2}{2})dt + \sigma dz \tag{3.1}$$

and

$$S(T) = S(0)\exp[(\mu - \frac{\sigma^2}{2})T + \sigma\epsilon\sqrt{T}] \tag{3.2}$$

Let $f(S, t, K, T)$ denote the value of option at time t with an underlying worth S, a strike price of K, and expiring at time T. The value at time 0 should equal to the expected value at maturity discounted back at a interest rate of r, which means

$$f(S,0,K,T) = e^{-rT}\mathbb{E}g(S(0)e^{[(\mu-\frac{\sigma^2}{2})T+\sigma\epsilon\sqrt{T}]}) \tag{3.3}$$

where $g(x)$ is the payoff function. For a European call option,

$$g(x) = \max(S(T)-K, 0)$$

Let $\{y_1, \ldots, y_M\}$ denotes M $i.i.d$ samples generated from $N(0,1)$. By the Law of Large Numbers,

$$f(S,0,K,T) = \frac{1}{M}e^{-rT}\sum_{i=1}^{M} g(S(0)e^{[(\mu-\frac{\sigma^2}{2})T+\sigma y_i\sqrt{T}]}) \tag{3.4}$$

To generate a random sample from $N(0,1)$, we can generate a random sample from $[0,1]$ first. Statistical randomness does not necessarily imply 'true' randomness, i.e., objective unpredictability. Pseudorandomness is sufficient for many uses. Most software provide the function of generating a pseudo-random number, most of which are based on the linear congruential generator, which uses the recurrence

$$X_{n+1} = (aX_n + b) \mod m$$

to generate numbers.

Suppose we have a series of samples which is independent uniformly distributed between 0 and 1, we can transfer it into a series of samples sampling from the normal distribution $N(0,1)$ by the following theorem:

Theorem: If U_1 and U_2 are independent and uniformly distributed between 0 and 1, define

$$X_1 = \sqrt{-2\ln U_1}\cos(2\pi U_2), Y_1 = \sqrt{-2\ln U_1}\sin(2\pi U_2)$$

Then we have, X_1 and X_2 are independent and both have the distribution of $N(0,1)$.

Proof: Firstly, we need these lemmas:

Lemma 1: If (X_1, X_2) and (Y_1, Y_2) have the same distribution, $g(x_1, x_2)$ and $h(x_1, x_2)$ are the 2-dimension real functions, define:

$$\begin{cases} Z_1 = g(X_1, X_2) & , \\ Z_2 = h(X_1, X_2) & , \end{cases}$$

$$\begin{cases} W_1 = g(Y_1, Y_2) & , \\ W_2 = h(Y_1, Y_2) & , \end{cases}$$

Then, (Z_1, Z_2) and (W_1, W_2) will also have the same distribution.

Proof: We can prove this lemma assuming (X_1, X_2) and (Y_1, Y_2)has the joint distribution density function $f(x, y)$.

$$\begin{aligned} P(Z_1 \le z, Z_2 \le w) &= P(g(X_1, X_2) \le z, h(X_1, X_2) \le w) \\ &= \int_{R^2} I\{g(x,y) \le z, h(x,y) \le w\} f(x,y) dxdy \\ &= P(g(Y_1, Y_2) \le z, h(Y_1, Y_2) \le w) \\ &= P(W_1 \le z, W_2 \le w) \end{aligned} \tag{3.5}$$

Therefore, (Z_1, Z_2) and (W_1, W_2) have the same distribution.

Lemma 2: If X and Y is independent and have the distribution of $N(0,1)$,(R, Θ) is determined by this polar coordinates transformation:

$$\Delta : \begin{cases} X = R\cos(\Theta) & , \\ Y = R\sin(\Theta) & , \end{cases} \tag{3.6}$$

Then, R has the Rayleigh Distribution, and Θ has the $[0, 2\pi]$ uniformly distribution.

Proof: (X, Y) has the joint distribution density function $f(x, y) = \frac{1}{2\pi}\exp(-\frac{x^2+y^2}{2})$. The range for (R, Θ) is

$$\{(r, \theta)|r \geq 0, \theta \in [0, 2\pi)\}$$

Denote the set $D = \{(x, y)|\sqrt{x^2 + y^2} \leq r, \alpha \in [0, \theta)\}$, where α is the angle of amplitude for (x, y). It is easy to see that, under the transformation Δ, $\{R \leq r, \Theta \leq \theta\} = \{(X, Y) \in D\}$, we thus can have the joint distribution function for (R, Θ) as,

$$\begin{aligned}
G(r, \theta) &= P(R \leq r, \Theta \leq \theta) \\
&= P((X, Y) \in D) \\
&= \int_D \frac{1}{2\pi}\exp(-\frac{x^2 + y^2}{2})dxdy(\text{ let } x = t\cos\alpha, y = t\sin\alpha) \qquad (3.7) \\
&= \frac{1}{2\pi}\int_0^\theta d\alpha \int_0^r \exp(-\frac{t^2}{2})dt \\
&= \frac{\theta}{2\pi}\int_0^r \exp(-\frac{t^2}{2})tdt
\end{aligned}$$

As $G(r, \theta)$ is continuous and is derivable except for limit linear lines, we can get the joint distribution density function for (R, Θ) by the derivative:

$$g(r, \theta) = \frac{\partial^2}{\partial r \partial \theta}G(r, \theta) = \frac{1}{2\pi}\exp(-\frac{r^2}{2})$$

where $r \geq 0, \theta \in [0, 2\pi)$.

As the variables in $g(r, \theta)$ are divided,R and Θ are independent and have the following density function respectively:

$$g_R(r) = r\exp(-\frac{r^2}{2})I_{[0,\infty)} \qquad (3.8)$$

$$g_\Theta(\theta) = \frac{1}{2\pi} I_{[0,2\pi)} \tag{3.9}$$

From equations (3.8) and (3.9),R has the Rayleigh distribution, and Θ has the $[0, 2\pi)$ uniformly distribution.

Let $R_1 = \sqrt{-2\ln U_1}, \Theta_1 = 2\pi U_2$, the distribution function of R_1 is

$$\begin{aligned} F(r) &= P(R_1 \le r) \\ &= P(\sqrt{-2\ln u_1} \le r) \\ &= P(u_1 \ge e^{-\frac{r^2}{2}}) \\ &= \int_{e^{-\frac{r^2}{2}}}^{1} du_1 \\ &= 1 - e^{-\frac{r^2}{2}} (r \ge 0) \end{aligned} \tag{3.10}$$

The density function of R_1

$$f(r) = F'(r) = re^{-\frac{r^2}{2}} \tag{3.11}$$

According to equation (3.11), R_1 has the Rayleigh distribution too. Therefore, (R_1, Θ_1) and (R, Θ) in (3.6) have the same distribution. According to the lemma, (X_1, Y_1) and (X, Y) in (3.6) will have the same distribution too, i.e. X_1 and Y_1 are independent and have the normal distribution $N(0, 1)$.

3.2 Numerical Results

Earlier we have illustrated the framework of the Monte Carlo simulation in pricing European option. We will check the accuracy and efficiency of this method by some numerical results.

We will use the Black-Schole model as the benchmark and compare them with results from Monte Carlo simulation. We want to price an European put option on a stock, with the current stock price S varying from 36 to 44, the strike price $k = 40$, the volatility of returns $\sigma = 0.40$, the short-term interest rate $r = 0.06$, and expires at $T = 1$ year. For every S, we run 10,000 paths for 5 times in Excel and use the function NORMSINV(RAND()) to generate a random sample from a normal distribution $N(0,1)$. The results are in Table 3.1.

Table 3.1: Monte Carlo simulation for European option pricing

S	B-S	1	2	3	4	5	average	s.e.	ave bias	bias in %
36	6.711	6.747	6.654	6.684	6.634	6.713	6.686	0.045	-0.024	-0.36%
38	5.834	5.791	5.838	5.850	5.824	5.749	5.810	0.040	-0.023	-0.40%
40	5.060	5.116	5.159	5.117	4.897	5.080	5.073	0.102	0.013	0.27%
42	4.379	4.380	4.381	4.363	4.412	4.308	4.368	0.038	-0.010	-0.23%
44	3.783	3.791	3.836	3.713	3.733	3.786	3.771	0.049	-0.011	-0.29%

The above table shows that Monte Carlo simulation works well in pricing European option. In our example, the difference between the simulation value and the true value (calculated from Black-Scholes Model) is less than 1 or 2 cents, or less than 0.5% in percentage.

We have illustrated this approach using a simple option. Actually, it also allows for increasing complexity such as compounding in the uncertainty (currency option, or model correlation between the underlying sources of risk, or the impact of inflation or commodity on the underlying). It can also be applied to price options in which the payoff depends on several underlying assets such as a basket option or rainbow option, when the correlation is considered in the simulation.

However, Monte Carlo simulation is quite time consuming, which limits its usage if alternative method exists in pricing. We will discuss about how to improve Monte Carlo simulation and widen its use in financial industry.

Chapter 4

LSM Algorithm for Pricing American Options

In the previous chapter, we presented Monte Carlo simulation algorithm to price European options. We now turn our attention to how to value American options and more complicated options.

Binomial trees and finite difference techniques work well in valuing simple American options but become impractical when there are multiple factors. Monte Carlo simulation is able to handle valuing path-dependent and multiple factors very well but is not well suited to pricing American style options until the Least Square Monte Carlo algorithm (LSM) was developed by Longstaff and Schwartz in 2001. When using Monte Carlo simulation, the primary difference between valuing European and American options is that American ones require the entire simulated path, whereas European ones only need the terminal value of the path.

As LSM combines the advantage of several numerical methods, it applies to various derivatives under a fairly general class of price dynamics. Besides, it is simple and efficient to implement. As a result, LSM technique becomes increasingly popular among the financial markets.

4.1 The Least Square Monte Carlo Algorithm (LSM)

Unlike European options, American options can be exercised earlier to maturity. In order to value them, it is necessary to make a decision whether to continue or to exercise at every exercise point. The exercising value is very easy to determine (for call option is $\max(S-K,0)$). The key issue becomes how to calculate the value of continuing. Many ways of determining the continuing are discussed by researchers, among which the most successful one is the Least Square Monte Carlo Algorithm (LSM) introduced by Longstaff and Schwartz (2001).

The LSM algorithm involves using the least square analysis to determine the best-fit relationship between the value of continuing and the value of early exercise. This algorithm is iterative in nature and constructs the estimated expected value of an American-style option at a time, conditional on that the option has not been exercised before that time. This conditional expectation is estimated using linear regression. More specifically, we first generate a collection of sample paths under appropriate price dynamics. The continuous interval of possible optimal early exercise times in American pricing problems is approximated with a discrete set of time points, and discounted future realized payoffs are regressed on functions of the state variable at each time points. A complete estimated optimal early exercise strategy is obtained under an application of the dynamic programming principle, which implies that we should exercise the option at the first time when the option is both in-the-money and has an estimated conditional expectation of continuation less than the value with immediate exercise. The estimated value of the option is the discounted estimated expected payoff.

We will illustrate a simple numerical example for better understanding of the intuition behind LSM algorithm. We want to price an American call option on a stock whose price is currently 100, and can be exercised at strike price 100 at the time 1 and 2. For simplicity, suppose the period between each exercise date is 1 year and the risk free interest rate is 5%. Assume that we have 10 simulated paths as in Table 4.1:

Table 4.1: Simulated paths

Path	t=0	t=1	t=2
1	100	110.2	111.1
2	100	106.6	101.4
3	100	89.9	84.9
4	100	119.6	107.9
5	100	83.1	105.0
6	100	94.4	86.3
7	100	100.8	91.8
8	100	106.7	109.7
9	100	92.4	93.1
10	100	75.5	72.4

The optimal strategy at time 2 is simply to exercise if the option is in-the-money, i.e. if $s(2) > 100$. Table 4.2 represents the realized cash flow following the optimal exercise strategy at time 2, conditional on the option not being exercised before time 2.

Table 4.2: Cash flow matrix at time 2

Path	t=0	t=1	t=2
1	-	-	11.1
2	-	-	1.4
3	-	-	0
4	-	-	7.9
5	-	-	5.0
6	-	-	0
7	-	-	0
8	-	-	9.7
9	-	-	0
10	-	-	0

Next, we will consider time 1. If the option is in-the-money at time 1, the option holder has to decide whether to continue or to exercise immediately. If the path is out-of money, we don't need to make the decision as it is not relevant to exercising. From the simulated path, there are 5 paths (path 1,2,4,7,8) are in-the-money at time 1. Let $\tilde{s}(1)$ be the in-the-money stock price at time 1 and $\tilde{y}(1)$ be the corresponding discounted cash flows at time 2 if continues. We have Table 4.3:

Table 4.3: Regression for time 1

Path	$\tilde{s}(1)$	$\tilde{y}(1)$
1	110.2	$11.1e^{-0.05}$
2	106.6	$1.4e^{-0.05}$
3	-	-
4	119.6	$7.9e^{-0.05}$
5	-	-
6	-	-
7	100.8	0
8	106.7	$9.7e^{-0.05}$
9	-	-
10	-	-

We can get the estimation of the conditional expectation of continuing at time 2 by regressing the discounted cash flow $\tilde{y}(1)$ on the current price $\tilde{s}(1)$ and a constant, which yields $E[Y|S(1) = s(1)] = -36.39+3.87s(1)$. The conditional expectation of continuing is less than the immediate exercise when $s(1) \geq 103.87$. This suggests that we should exercise at time 1 in path 1,2,4, and 8. The corresponding cash flow matrix is as as in Table 4.4:

Table 4.4: Cash flow matrix

Path	$t = 1$	$t = 2$
1	10.2	-
2	6.6	-
3	-	-
4	19.6	-
5	-	5.0
6	-	-
7	-	-
8	6.7	-
9	-	-
10	-	-

The value of the option is the average of the discounted value, i.e.

$$c = \frac{(10.2 + 6.6 + 19.6 + 6.7)e^{-0.05} + 5.0e^{-0.05*2}}{10} = 4.55$$

Technically, we assume a time horizon $[0, T]$ and an underlying probability space $(\Omega, \mathcal{F}, P)$. Ω is the set of all possible realizations of the stochastic economy on $[0, T]$ and has typical element ω. $\mathcal{F} = \mathcal{F}_T$ is the σ-field of distinguishable events through time T, and P is the probability measure defined on the sets in $\mathcal{F}$. Let $\mathbb{F} = (\mathcal{F}_t; t \in [0, T])$ represent the filtration generated by the associated price dynamic. Under the no-arbitrage paradigm, we can assume the existence of a risk neutral probability measure Q.

Our goal is to determine the value of American-style derivative securities which generate random cash flows during $[0, T]$. The appropriate value is the maximized expected value of the discounted cash flows from the option, where the maximum extends over all stopping times with respect to $\mathbb{F}$. Moreover, no-arbitrage pricing paradigm implies that the value of continuation is the expected value of remaining cash flows with respect to Q when following the optimal stopping rule. Suppose we can approximate the continuous interval of possible early exercise times with L discrete times $0 < t_1 \leq t_2 \leq \ldots \leq t_L = T$. We can write the expected value as

$$G(\omega; t_l) = E_Q[\sum_{j=l+1}^{L} e^{-\int_{t_1}^{t_j} r(\omega,u)du} CF(\omega, t_j; t_l, T)|\mathcal{F}_{t_l}] \tag{4.1}$$

where $CF(\omega, s; t_l, T)$ denotes the path of cash flows generated by the option, conditional on the option not being exercised at or prior to time t, and it is based on that the option holder is following the optimal exercise strategy at all time $s, t_l \leq s \leq T$, and $r(\omega, t)$ is the risk-free interest rate.

As the name implies, LSM uses least squares regression to estimate the conditional expectation function at each of the possible early exercise dates. More specifically, at time t_l, assume that the unknown functional form of $G(\omega; t_l)$ can be written as a countable linear combination of $\mathcal{F}_{t_l}$-measurable basis functions. As we are focusing on derivatives with finite-variance random payoffs, the conditional expectation $G(\omega; t_l)$ will be in the space of square integrable functions under the appropriate measure. This Hilbert space will have a countable orthonormal basis, which implies that $G(\omega; t_l)$ can be represented as a countable linear combination of elements of this basis. In practice, LSM uses $M < \infty$ elements from this basis to approximate. Denote it as $G_M(\omega; t_l)$. Weak assumptions about the existence of moments imply that the fitted value from the regression $\widehat{G}_M(\omega; t_l)$ converges in probability and in mean square to $G_M(\omega; t_l)$ as the number of sample paths tends to infinity.

As we have mentioned, the idea underlying the LSM algorithm is that the conditional expectation can be approximated by a least-squares regression for each exercise date. At time t_{L-1}, $G(\omega; t_{L-1})$ can be expressed as a linear combination of orthonormal basis functions $(p_j(X))$ such as Laguerre, Hermite, Chebyshev, Genbauer, Legendre or Jacobi polynomials. which is

$$G(\omega; t_{L-1}) = \sum_{j=0}^{\infty} a_j p_j(X), a_j \in \mathbb{R} \tag{4.2}$$

which is approximated by

$$G_M(\omega; t_{L-1}) = \sum_{j=0}^{M} a_j p_j(X), a_j \in \mathbb{R} \tag{4.3}$$

This procedure is repeated backwards until the first exercise date.

4.2 Convergence and Robustness of LSM

As we have discussed, LSM approximates the true value of American-style option with discrete exercise time points and regression on a selection of basic function. In this section, we will briefly discuss about the convergence of the algorithm with the discrete time N and robustness of the selection of basic function.

In Longstaff and Schwartz's paper, they provided some convergence results. The first one is about the bias between LSM with discrete exercise points and the true value of the continuously exercisable option.

Proposition 1. Let $V(X)$ denote the true value of the American option and $CF(w_i; M, K)$ denote the discounted cash flow resulting from the LSM algorithm. Then the following inequality holds almost surely.

$$V(X) \geq \lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} CF(w_i; M, K)$$

This proposition shows that the true value of the American-style option is bigger than or equal to any value resulting from the stopping rule implied by LSM as it is based on the stopping rule which maximizes its value.

A general convergence result of LSM needs to consider limits when the number of paths N, the number of basis functions M, and the number of discrete exercise points K. Longstaff and Schwartz also provided this following result:

Proposition 2. Assume the value of option $V(X)$ depends on a single state variable X following a Markov process with support on $(0, \infty)$. Also assume the option can only be exercised at time t_1 and t_2 (it is very easy to generalize to the case of K exercise points), and the conditional expectation function $G(w; t_1)$ is absolutely continuous and

$$\int_0^\infty e^{-X} G^2(w; t_1) dX < \infty$$

$$\int_0^\infty e^{-X} G_X^2(w; t_1) dX < \infty$$

Then for any $\varepsilon > 0$, there exists an $M < \infty$ such that

$$\lim_{N \to \infty} P[|V(X) - \frac{1}{N} \sum_{i=1}^{N} CF(w_i; M, K)| > \varepsilon] = 0$$

This result shows that by selecting M large enough and letting $N \to \infty$, the result from LSM converge to the true value in probability. Another important implication of this result is that the number of basis functions selected in the regression need not be infinite to get an accurate estimation of the true value V.

As an extension, Clement, Lamberton, and Protter(2002) established some other LSM convergence results. They prove that the LSM algorithm converges almost surely under fairly general conditions. They also determined the rate of convergence and showed that the normalized error is asymptotically Gaussian.

As the regression involves basic function, the question that whether the selection of difference will cause different result or not raises very naturally. There are also several papers regarding this robustness problem of LSM.

In 2003, Manuel Moreno and Javier Navas analyzed the impact of different basis functions on option prices. They applied the LSM algorithm to pricing an American put option, an American-Bermuda-Asian option, and a Bermuda call option on maximum of 5 assets. They calculated in- and out-of sample option prices and the standard errors for different types and numbers of basis functions.

Their work shows that LSM has different robustness characters in pricing different options. In pricing American put option, the LSM technique is very robust.

Using different polynomials, it produces very similar results and the standard errors are very low. And they also obtained very similar option prices using different polynomial degree (between 3 and 20). But for those complex options, the robustness is not guaranteed and the choice of basic function is not so clear. In this case, the choice of type and number of basic functions can slightly affect options prices in an acceptable range.

4.3 Improvement for LSM

LSM algorithm works well in pricing American options and other complex options. However, it is quite time consuming when the dimension is huge and the number of path is large enough to obtain an accurate estimation.

There are several papers on how to improve the computational speed of LSM. A.-S. Chen and P.-F. Shen (2003) provided computational complexity analysis of LSM. They broke down the algorithm into logical modules and analyzed the effect of adding or deleting logical modules on the algorithm. Finally, they found out that a truncated algorithm improves both the computational speed and the accuracy of the results. Another paper was proposed by A.R. Choudhury, A.King, S.Kumar, and Y.Sabharwal (2008) on trying to implement the parallelization techniques to LSM. They divided the whole algorithm into three phases and described how to parallelize each phase. Finally, they achieved some speed-up results.

Another way that we can improve LSM may rely on speeding up Monte Carlo simulation. As we need random number for Monte Carlo simulation and normally we will generate the pseudo-random number as stated. We can also consider another way in which is based on low-discrepancy sequences. This method is so called Quasi-Monte Carlo methods. The reason to try to improve with quasi-Monte Carlo methods is to generate random paths that is better distributed in the sample space, so that we may get a better estimate with a given accuracy using a smaller sample size.

Quasi-Monte Carlo and Monte Carlo methods are stated in a similar way. Again, we consider the problem of calculating the integral of a function f as the mean of $f(x_i)$ on the set of points $x_1, x_2, \ldots, x_N$, i.e.

$$\int_{I^s} f(u)du \approx \frac{1}{N}\sum_{i=1}^{N} f(x_i)$$

where $I^s = [0,1] \times \ldots \times [0,1]$ is a s-dimensional unit cube. In a quasi-Monte Carlo method, the set $x_1, x_2, \ldots, x_N$ is a subsequence of a low-discrepancy sequence contrast to a subsequence of pseudorandom numbers in Monte Carlo method. Regarding the low-discrepancy sequence, Halton (1960), Faure (1982), Sobol (1967), and Niederreiter (1987), are the best known ones. To estimate the approximation error of this method, we will introduce Koksma-Hlawka inequality first.

Using Niederreiter's notation, the discrepancy of a sequence (X_i) is defined as

$$D_N(P) = \sup_{B \in J} \left| \frac{A(B;P)}{N} - \lambda_s(B) \right|$$

where P is the set $x_1, x_2, \ldots, x_N$, λ_s is the s-dimensional Lebesgue measure, $A(B;P)$ is the number of points in P which fall into the set B, and J is the set of s-

dimensional intervals in the form of $\prod_{i=1}^{s}[a_i, b_i) = \{x \in \mathbb{R}^s : a_i \leq x_i < b_i\}$, where $0 \leq a_i < b_i \leq 1$. The star-discrepancy $D_N^*(P)$ is similarly defined except that the supremum is taken over the set J^* in the form $\prod_{i=1}^{s}[0, u_i)$, where $u_i \in [0, 1)$. Let F have bounded variation $V(f)$ on I^s in the sense of Hardy and Krause. Then for any sequence $x_1, x_2, \ldots, x_N$ in I^s, we have

$$|\int_{I^s} f(u)du - \frac{1}{N}\sum_{i=1}^{N} f(x_i)| \leq V(f)D_N^*(x_1, x_2, \ldots, x_N)$$

The Koksma-Hlawka inequality states that for any point set $x_1, x_2, \ldots, x_N$ in I^s and $\varepsilon > 0$, there is a function F with bounded variation and $V(f) = 1$ such that

$$|\int_{I^s} f(u)du - \frac{1}{N}\sum_{i=1}^{N} f(x_i)| > D_N^*(x_1, x_2, \ldots, x_N) - \varepsilon$$

Therefore, the quality of this numerical technique depends only on the discrepancy of the sequence $D_N^*(x_1, x_2, \ldots, x_N)$.

The discrepancy of sequences usually used for the quasi-Monte Carlo method is bounded by $a\frac{(\log N)^s}{N}$ where a is a constant. But in Monte Carlo method, expected discrepancy of a uniform random sequence has a convergence order of $\sqrt{\frac{\log\log N}{2N}}$ by the law of the iterated logarithm. It shows that the accuracy of the quasi-Monte Carlo method is faster than that of Monte Carlo method.

We will mainly introduce the Faure sequence here, and for more details about the discrepancy and the low-discrepancy sequences, please refer to Glasserman (2004).

We begin by constructing a 1-dimensional Faure sequence first. Let p be a prime number, then for any integer m, there is a unique expansion in base p. We can express m as

$$m = \sum_{j=0}^{l} a_{j,p}^{(1)}(m)p^j$$

where l is the smallest integer to generate the expansion of m based on p.

A quasi-random number in $[0, 1]$ can be generated by mapping the expansion of m to a point in $[0, 1]$ with the radical inverse function θ_p. Thus, the quasi-random number can be expressed as

$$\theta_p^{(1)}(m) = \sum_{j=0}^{l} a_{j,p}^{(1)}(m)p^{-j-1}$$

Take $p = 3$ and $m = 7$ for example, we have the expansion of $7 = 2(3^1) + 1(3^0)$. The radical inverse function is $\theta_3^{(1)}(7) = 1(3^{-1}) + 2(3^{-2}) = \frac{5}{9}$. Quasi-random numbers in the sequence $\theta_3^{(1)}(i)$ for $i = 1, \ldots, 9$ are $(\frac{9}{27}, \frac{18}{27}, \frac{3}{27}, \frac{12}{27}, \frac{21}{27}, \frac{6}{27}, \frac{15}{27}, \frac{24}{27}, \frac{1}{27})$. Notice that the numbers added fill the gaps of the existing sequence, it is possible to terminate the simulation when we have obtained the desired level of accuracy.

Similarly, we can construct a d-dimensional Faure sequence. The coordinates of the first dimension is the same as shown above. The coordinates of other dimensions are constructed recursively, i.e.

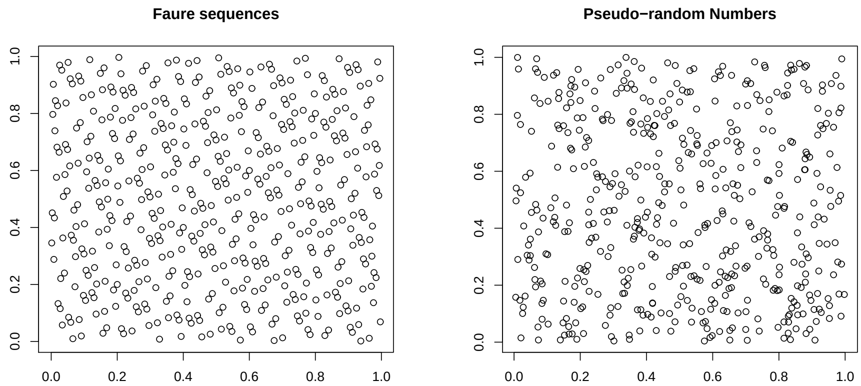
$$a_{j,p}^{(k)}(m) = [\sum_{i \geq j}^{l} [\frac{i! a_{j,p}^{(k-1)}(m)}{j!(i-j)!}]] \mod p,$$

$$\theta_p^{(k)}(m) = [\sum_{j=0}^{l} a_{j,p}^{(k)}(m)^{-j-1}, 2 \leq k \leq d.$$

For the d-dimensional Faure sequences, the coordinates of dimension other than 1 are simple permutations of the first dimension's coordinates. As a matter of fact, the i-th coordinates of any two dimensions cannot be the same.

For the comparison of pseudo-random numbers and Faure sequences in $[0,1] \times [0,1]$, please see Figure 4.1:

Figure 4.1: Comparison of Faure sequences and pseudo-random numbers



This picture shows the first 500 random numbers of each kind. It is clear that Faure sequences are better spaced in the sample space than the pseudo-random numbers, leading a possible more accurate result.

In our example, in order to construct N quasi random paths sampling from the normal distribution $N(0,1)$, we can evenly space out the quantile function of normal distribution and distribute these quasi random numbers in each path randomly. For example, if we want to simulate 1000 paths at time t, we can evenly divide the interval $[0,1]$ into 1000 intervals and select out the midpoint of each interval, i.e. $1/2000, 3/2000, \ldots, 1999/2000$. Then we will convert these midpoints to quantiles by applying the quantile function of normal distribution $\phi^{-1}(p)$ to each of the midpoints, yielding 100 quasi random numbers. We then get a random permutation from 1 to 1000 and distribute the corresponding quasi random number to each path. We can finally construct the quasi random paths by applying the same procedure to each time points. We will carry out some simulation studies in the next section and compare this improved LSM method with the original one.

4.4 Numerical Results

In chapter 3, we have shown how to price European options by Monte Carlo simulation. Here, we will extend its use to price American options. We will firstly check the LSM's pricing ability. We then will apply the improved LSM method and compare it with the original one. We will also discuss about the tradeoff between the computational time and the precision of the price regarding numbers of paths in simulation, different basic functions in the regression process and numbers of possible exercise time points. In consistency with European option and the original paper of Longstaff and Schwartz, we will use similar parameters as they selected.

4.4.1 LSM for Pricing American Options

We plan to price an American put option on a stock, with the current stock price S varying from 36 to 44, the strike price $K = 40$, the volatility of returns $\sigma = 0.40$, the short-term interest rate $r = 0.06$, and expires at $T = 1$ year. For every S, we run 100,000 (50,000 plus 50,000 antithetic) paths for 5 times in R and use the function rnorm() to generate a random sample from a normal distribution $N(0, 1)$. The detailed codes for R is included in the Appendix. The results are shown in Table 4.5:

Table 4.5: LSM for American option pricing

S	FD	1	2	3	4	5	average	s.e	ave bias	bias in %
36	7.101	7.145	7.124	7.089	7.140	7.151	7.130	0.025	0.029	0.40%
38	6.148	6.091	6.112	6.165	6.173	6.165	6.141	0.037	-0.007	-0.11%
40	5.312	5.293	5.307	5.278	5.318	5.305	5.300	0.015	-0.012	-0.22%
42	4.582	4.575	4.613	4.620	4.567	4.611	4.597	0.024	0.015	0.33%
44	3.948	3.909	3.940	3.965	3.942	3.956	3.943	0.021	-0.005	-0.14%

In this table, the column "FD" means the value implied by finite difference, which we use as a benchmark for our method. The above table shows that LSM works well in pricing American option. In our example, the difference between the simulation value and the benchmark value (calculated from finite difference) is less than 3 cents, or less than 0.5% in percentage.

We will also have a test on the market data. In CBOE, all options on individual stocks are American style and options on the major indexes are European except for those of S&P 100 Index (OEX). As for American options on individual stocks, we still need to consider dividend. For simplicity, we will take several options on OEX for testing purpose. We select the price of 2 American put options on OEX with the code 109764, the trading date is Jan 3, 2008, expiration date about one year after, the spot OEX index is 676, the strike price is 660 and 800 respectively, and implied volatility is 0.224 and 0.172 respectively. For each S, we run 100,000

(50,000 plus 50,000 antithetic) paths for 5 times in R with the interest rate $r = 0.02$. we will quote the bid-ask prices and the simulation results in the following table

Table 4.6: LSM on OEX options

K	Implied vol	1	2	3	4	5	average	Bid	Ask
660	0.224	42.9	44.9	47.9	44.3	41.9	44.4	44.4	46.4
800	0.172	126.3	126.5	126.4	125.8	125.7	126.2	126.0	128.0

This table shows that our simulation on these real market option works well too, as both of the simulation results fall into the bid ask spread.

From the comparison with finite difference and market price of traded option, LSM simulation results are quite accurate. This means that we have successfully applied Monte Carlo simulation to the valuation of American-style option, which used to be considered impractical before.

4.4.2 Improved LSM vs Original LSM

As we have discussed, we will improve LSM by generating quasi random paths instead of pseudo-random ones. Similarly, we plan to price an American put option on a stock, with the current stock price S varying from 36 to 44, the strike price $K = 40$, the volatility of returns $\sigma = 0.40$, the short-term interest rate $r = 0.06$, and expires at $T = 1$ year. For every S, we run 100,000 (50,000 plus 50,000 antithetic) paths for 5 times in R. We use the function qnorm() to get the quantile value at each probability and get a random permutation from 1 to N using the function sample($1:N$). The detailed codes for R is included in the Appendix. The results are shown in Table 4.7:

Table 4.7: Improved LSM for American option pricing

S	FD	1	2	3	4	5	average	s.e	ave bias	bias in %
36	7.101	7.130	7.108	7.103	7.117	7.116	7.115	0.010	0.014	0.19%
38	6.148	6.147	6.163	6.145	6.154	6.160	6.154	0.008	0.006	0.10%
40	5.312	5.285	5.315	5.317	5.320	5.327	5.313	0.016	0.001	0.02%
42	4.582	4.591	4.600	4.595	4.560	4.621	4.593	0.022	0.011	0.25%
44	3.948	3.964	3.973	3.962	3.956	3.959	3.963	0.006	0.015	0.38%

In this table, the column "FD" means the value implied by finite difference, which we use as a benchmark for our method. The above table shows that improved LSM works well in pricing American option. In our example, the difference between the simulation value and the benchmark value (calculated from finite difference) is less than 1 or 2 cents, or less than 0.4% in percentage.

To compare the accuracy and stability of these two methods, we will compare the standard error and the bias in the simulation. The results are shown in Table 4.8.

Table 4.8: Improved LSM vs LSM in accuracy and stability

S	LSM s.e	LSM bias in %	improved s.e	improved bias in %
36	0.025	0.40%	0.010	0.19%
38	0.037	-0.11%	0.008	0.10%
40	0.015	-0.22%	0.016	0.02%
42	0.024	0.33%	0.022	0.25%
44	0.021	-0.14%	0.006	0.38%
average	0.025	0.24%	0.013	0.19%

It is shown that in the simulation for these 5 American options, the improved LSM reduces about 50% in standard error and reduce about 20% in average bias (to take the average of the absolute value of bias), which means that our method did improve the the original method in both accuracy and stability.

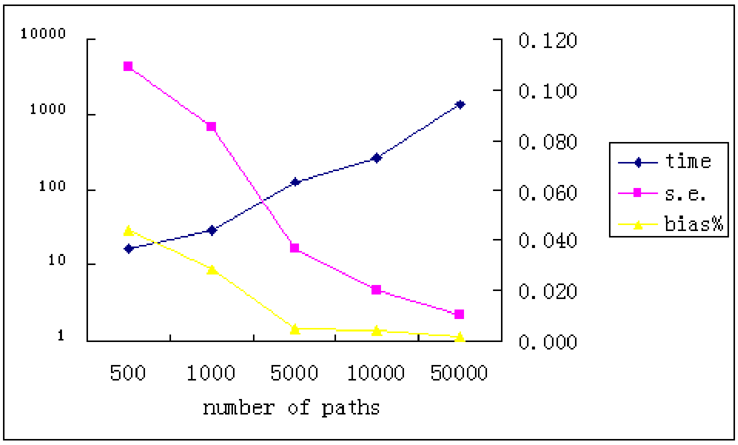
4.4.3 The Effect of Number of Paths

To check the impact of number of paths on the results, we run the simulation in R several times with different number of paths ranging from 500 to 50000 and compare the computational time, standard error, bias in percentage(all of the computational time in this thesis is based on processor of Intel(R) Core(TM)2 Duo CPU E6750 @ 2.66 GHz 2.67 GHz and memory of RAM 2.00 GB). We select the option with $S = 36$ and other parameters are kept as the same. The results are shown in Table 4.9 and Figure 4.2.

Table 4.9: The effect of number of paths

No. of paths	time(sec)	s.e.	bias in %
500	17	0.109	4.39%
1000	29	0.085	2.83%
5000	131	0.037	0.48%
10000	260	0.020	0.43%
50000	1376	0.010	0.19%

Figure 4.2: The effect of number of paths



It is shown that the computational time increases almost proportionally as number of paths increase, and s.e. and bias% decrease as paths increase. Considering the bid-ask spread and bias in other parameters in real market, our method can reach an acceptable of 0.5% bias as few as more than 5000 paths. To obtain a more accurate results and a less standard error, usually we will need about 50000 paths, which will take about 20 minutes to run the simulation.

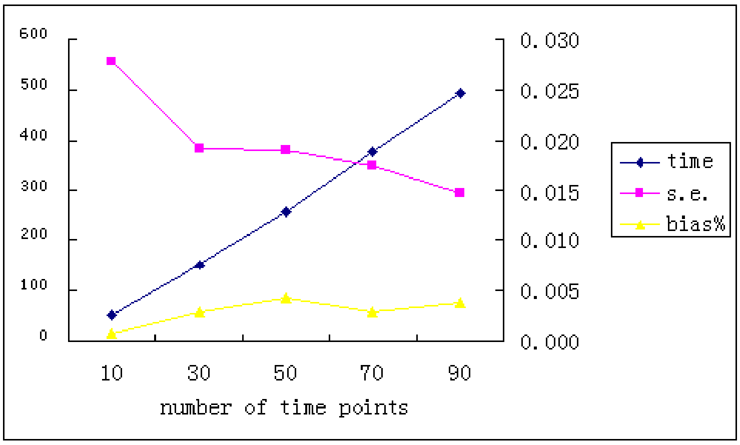
4.4.4 The Effect of Number of Exercise Time Points

As discussed in the convergence part, we can increase the number of exercise time points N to limit to the true value of an option which is exercisable at any time. But how large is N should be to obtain an accurate enough result? To check the impact of number of paths on the results, we run the simulation in R several times with different number of exercise time points ranging from 10 to 90 per year and compare the computational time, standard error, bias in percentage. As the memory of R is limited, we can only select up to 10000 paths in our simulation. Similarly, we select the option with $S = 36$ and keep other parameters the same. The results are shown in Table 4.10 and Figure 4.3.

Table 4.10: The effect of exercise time points

No. of time points	time(sec)	s.e.	bias in %
10	50	0.028	0.08%
30	150	0.019	0.28%
50	258	0.019	0.43%
70	375	0.017	0.28%
90	494	0.015	0.38%

Figure 4.3: The effect of exercise time points



It is shown that the computational time increases almost proportionally as number of exercise time points increases, and s.e. decreases as time points increase. For bias%, it does not change too much when the number of exercise time points is increased. Therefore, to obtain a s.e. less than 0.02, we usually need more than 30 time points.

4.4.5 The Effect of Polynomial Degrees in Regression

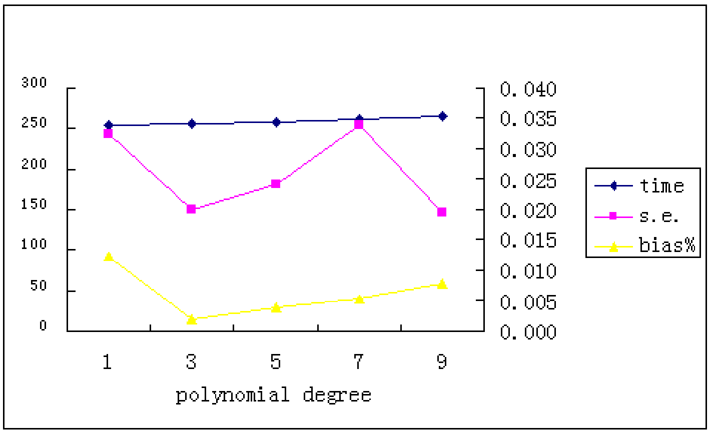
As discussed in the robustness part, in most cases, LSM is quite robust in selecting different type and number of basic functions. As our method is based on LSM and can obtain a less standard error under similar conditions with LSM, it should be at least as robust as LSM. To check the impact of number of paths on the results, we run the simulation in R several times with different polynomial degrees ranging from 1 to 9. Similarly, we select the option with $S = 36$, paths $N = 10000$ and keep other parameters the same. The results are shown in Table

4.11 and Figure 4.4.

Table 4.11: The effect of polynomial degrees

polynomial degrees	time(sec)	s.e.	bias in %
1	254	0.033	1.22%
3	256	0.020	0.20%
5	258	0.024	0.38%
7	261	0.034	0.52%
9	265	0.020	0.77%

Figure 4.4: The effect of polynomial degrees



It is shown that the computational time increases slowly as polynomial degree increases. Standard error and bias% do not change too much when polynomial degree increases except for the degree is 1, i.e. the regression is too simple as $y \sim x$. In all, our method is quite robust on the polynomial degree in the regression.

Chapter 5

Conclusion and Future Research

In this thesis, we have reviewed the research work on option pricing by both analytical way and numerical methods, and on how to apply Monte Carlo simulation to price both European and American options. To overcome the difficulties of pricing American options by Monte Carlo simulation, we have introduced the Least Square Monte Carlo algorithm (LSM) suggested by Longstaff and Schwartz (2001). We have also tried to improve this algorithm by applying quasi-Monte Carlo methods, which is able to enhance the accuracy and computational speed of LSM. To see whether these method are applicable or not, we carried out some simulation studies.

Simulation results show that our method do have improved the original LSM in both accuracy and robustness, by generating a less bias in percentage and a less standard error under similar conditions. Moreover, better accuracy and robustness are able to be obtained by increasing the number of paths in simulation, at the cost of computational time and the memory size needed in the program. We can also reduce the standard error by increasing the number of exercise time points in the simulation. In consistency with other earlier research, the improved LSM is quite robust on selecting different polynomial degree in the regression, if the degree is not too small of course.

Although we only carried out simulation studies on the plain American options, we can also handle with increasing complexity such as compounding in the uncertainty, and options in which the payoff depends on several underlying assets such as a basket option or rainbow option, when the correlation is considered in the simulation. Simulation has many important advantages in valuing and risk managing derivatives. After overcoming its drawback on pricing American-style options, the application of simulation in pricing derivatives becomes much more promising and broader. It becomes much easier to implement simulation to advanced models and more complicated derivatives in practice.

For future research, we may want to provide more throughout mathematical foundation for LSM. We can try to give more detailed convergence results of the LSM, the rate of convergence, the robustness of the algorithm when selecting different types and numbers of basic functions in different scenario.

As LSM uses ordinary least squares to obtain the estimation of conditional expectation, it may be more efficient to apply other regression techniques such as generalized least squares or weighted least squares, especially when the residuals of regression may be heteroskedastic in some cases.

Also, with its ability to handle multiple stochastic factors and American-style, we may be able to combine LSM with advanced models other than Black-Scholes model such as Jump-Diffusion models, Variance-Gamma models and Stochastic Volatility models, which reflect more in the real market and is widely used in the financial industry. We can also apply LSM to price more complicated derivatives such as accumulator, range accrual on multiple assets.

Appendix

Related R code

```
##main LSM

LSM<-function (s,strike,sig, riskFreeRate, realRate,n,m){

##s, stock price at the beginning

##sig, instantaneousstandard

deviation of GBM

##n, number of paths used in the simulation

##m,

number of exercise times(usually 50/year)


sMat<-method2(s,sig,realRate,n,m)

cashFlowMat<-as.matrix(ifelse(sMat[,m]<strike,strike-sMat[,m],0))

m<-m-1

cashFlowMat<-method1(strike,riskFreeRate,m,sMat,cashFlowMat)

counter<-apply(cashFlowMat,1,function(x)
```

```
ifelse(max(x)==0,1,which(x>0)[1]))
  discCashFlow<-cashFlowMat[cbind(1:n,counter)]*exp(-counter*riskFreeRate/50)
return(mean(discCashFlow)) }


##method1 -- LSM recursion
method1<-function(strike,
riskFreeRate,m,sMat,cashFlowMat)
{ if(m==0) {return(cashFlowMat)}
else{ numInMoney<-sum(sMat[,m]<strike)
sy<-numeric(2*numInMoney)
dim(sy)<-c(numInMoney,2)inMoney<-which(sMat[,m]<strike)
reducedCashFlowMat<-cashFlowMat[inMoney,]
reducedCashFlowMat<-as.matrix(reducedCashFlowMat)
sy[,1]<-sMat[inMoney,m]
counter<-apply(reducedCashFlowMat,1,function(x)
ifelse(max(x)==0,1,which(x>0)[1]))
sy[,2]<-reducedCashFlowMat[cbind(1:numInMoney,counter)]*exp(-counter*riskFreeRat
s<-sy[,1] sSq<-sy[,1]^2 sCu<-sy[,1]^3 sFo<-sy[,1]^4 sFi<-sy[,1]^5
sSi<-sy[,1]^6 sSe<-sy[,1]^7
lmFit<-lm(sy[,2]~s+sSq+sCu+sFo+sFi+sSi+sSe)
dfForLm<-data.frame(
```

```
s=sMat[,m], sSq=sMat[,m]^2, sCu=sMat[,m]^3, sFo=sMat[,m]^4,
sFi=sMat[,m]^5, sSi=sMat[,m]^6, sSe=sMat[,m]^7)
theFit<-predict(lmFit,dfForLm)
cashFlowNew<-ifelse((sMat[,m]<strike)&((strike-sMat[,m])>theFit),strike-sMat[,m]
cashFlowMat<-cbind(cashFlowNew,cashFlowMat)
return(method1(strike,riskFreeRate,m-1,sMat,cashFlowMat)) } }


##method2 -- Generate GBM paths
method2<-function(s,sig,realRate,n,m)
{ sMat<-numeric(n*(m+1))
dim(sMat)<-c(n,(m+1))
sMat[,1]<-s
for(j in 1:m){ a<-rnorm(n)
mu=0.06
sMat[,j+1]<-sMat[,j]*exp(((mu-(sig^2/2))/50)+(sig*sqrt(1/50)*a)) }
return(sMat[,2:(m+1)]) }



##method2 -- Generate Quasi random paths
method2<-function(s,sig,realRate,n,m)
{
```

```
sMat<-numeric(n*(m+1))

dim(sMat)<-c(n,(m+1))

sMat[,1]<-s

rp<-numeric(n*m)

dim(rp)<-c(n,m)

for(j in 1:m)

rp[,j]<-sample(1:n)

qn<-numeric(n)

dim(qn)<-c(1,n)

for

(i in 1:n)

qn[i]<-qnorm((2*i-1)/(2*n)) a<-numeric(n*m)

dim(a)<-c(n,m) f

or(i in 1:n) {

    for (j in 1:m)

    a[i,j]=qn[rp[i,j]]

}


for(j in 1:m)

{ mu=0.06

sMat[,j+1]<-sMat[,j]*exp(((mu-(sig^2/2))/50)+(sig*sqrt(1/50)*a[,j]))
```

```
}

return(sMat[,2:(m+1)]) }


##Generate Faure Sequences

FaureGen<-function(n,p,s){

dimNCR<-trunc(log(n,base=p))+1

nCRMat<-numeric((dimNCR)^2)

dim(nCRMat)<-c((dimNCR),

(dimNCR))

for(i in 1:dimNCR){

for(j in

1:dimNCR){

nCRMat[i,j]<-choose(j-1,i-1) } }

baseRep<-numeric(dimNCR*n)

dim(baseRep)<-c(n,dimNCR)

intList<-seq(1:n)

for(j in dimNCR:1){

baseRep[,j]<-trunc((intList/p^(j-1))%%p)

}

faureMat<-numeric(n*s) dim(faureMat)<-c(n,s)

for(i in 1:dimNCR){
```

```
faureMat[,1]<-faureMat[,1]+baseRep[,i]*p^(-i) }

for (i in 1:n){

fromBaseRep<-baseRep[i,]

for(j in 2:s){

nextBaseRep<-(nCRMat%*%fromBaseRep)%%p

for(k in 1:dimNCR){

faureMat[i,j]<-faureMat[i,j]+nextBaseRep[k]*p^(-k) }

fromBaseRep<-nextBaseRep } }


 return(faureMat) }


plot(FaureGen(500,5,2),main="Faure sequences",xlab="",ylab="")

plot(runif(500),runif(500),main="Pseudo-random

Numbers",xlab="",ylab="")
```

Bibliography

[1] Amin, K.I. (1991), On the Computation of Continuous Time Option Prices Using Discrete Approximations. *Joural of Financial and Quantitative Analysis* 26, 477-495.

[2] A.R. Choudhury, A.King, S.Kumar, and Y.Sabharwal. (2008) Optimizations in Financial Engineering The Least Squares Monte Method of Longstaff and Schwartz. Parallel and Distributed Processing, *IPDPS 2008.* IEEE International

[3] A.-S. Chen and P.-F. Shen, (2003), Computational Complexity Analysis of Least-squares Monte Carlo (LSM) for Pricing US Derivatives. *Applied Ecomomics Letters*, 10, 223-229

[4] Billingsley, P. (1995) *Probability and Measure, 3rd edition.* Wiley, New York.

[5] Black, F. and M. Scholes, (1973) The Pricing of Options and Corporate Liabilities. *Journal of Political Economy.* 81, 673-654

[6] Boyle, and Phelim P.,(1977) Options: A Monte Carlo Approach. *Journal of Financial Economics.* 4, 323-338

[7] Broadie, M. and P. Glasserman, (1997) Pricing American-Style Securities Using Simulation. *Journal of Economic Dynamics and Control.* 21, 1323-1352

[8] Broadie, M. and P. Glasserman,(1996) Estimating Security Price Derivatives Using Simulation, *Management Science*, 42, 269-285.

[9] Broadie, M. and Kaya, O. (2006) Exact Simulation of Stochastic Volatility and Other Affine Jump Diffusion Processes. *Operations Research* 54, 217-231.

[10] Broadie, M., and Jerome B. Detemple (2004), Option Pricing: Valuation Models and Applications, *Management Science*, 50, 1145-1177.

[11] Carriere, J. (1996) Valuation of Early-Exercise Price of Options Using Simulations and Nonparametric Regression. *Insurance: Mathematics and Economics.* 19, 19-30

[12] Clement, E., Lamberton, D., and Protter, P.(2002) An Analysis of a Least Squares Regression Method for American Option Pricing. *Finance and Stochastics*, 6, 449-471

[13] Cox, John C. and Ross, Stephen A., and Rubenstein, Mark (1979) Option Pricing: A Simplified Approach. *Journal of financial Economics* 7, 229-263.

[14] Egloff, D. (2005) Monte Carlo Algorithms for Optimal Stopping and Statistical Learning. *The Annals of Applied Probability* 15, 1396-1432.

[15] Emmanuelle Clement, Damien Lamberton, and Philip Protter. (2002) An Analysis of A Least Squares Regression Method for American Option Pricing. *Finance Stochast.* 6, 449-471

[16] Glasserman, P.(2004) *Monte Carlo Methods in Financial Engineering.* Springer

[17] Glasserman, P., and Bin Yu, (2004) Mumber of Paths Versus Number of Basis Functions in American Option Pricing, *The Annuals of Applied Probability,* 14, 2090-2119.

[18] Hull, J.C. (2006) *Options, Futures, and Other Derivatives*, 6th edition, Pearson Education.

[19] Hull, John and White, Alan. (1990) Valuing Derivative Securities Using the Explicit Finite Difference Method. *Journal of Financial and Quantitative Analysis* 25, 87-100.

[20] Joy, Corwin, Boyle, Phelim, and Tan, Ken Seng. (1996) Quasi-Monte Carlo Methods in Numerical Finance. *Management Science* 42, 926-938.

[21] Ibanez, A., and F. Zapatero, (1998), Monte Carlo Valuation of American Options through Computation of the Optimal Exercies Frontire. working paper, University of Southern California.

[22] Lars Stentoft (2004), Assessing the Least Squares Monte-Carlo Approach to American Option Valuation. *Review of Derivatives Research*, 7, 129.

[23] Lars Stentoft (2003), Convergence of the Least Squares Monte Carlo Approach to American Option Valuation, working paper.

[24] Longstaff F.A. and E.S. Schwartz, (2001), Valuing American options by simulation: a simple least squares approach, *The Review of Financial Studies* 14, 113-148

[25] Moreno, M. and Navas, J. (2003) On the Robustness of Least-Squares Monte Carlo (LSM) for Pricing American Derivatives. *Review of Derivatives Research* 6, 107-128.

[26] Morokoff, and W.j., R.E.Caflishch, (1994), Quasi-Random Sequences and Their Discrepancies. *SIAM Journal of Computing*. 15, 1251-1279.

[27] Morokoff, and W.j., R.E.Caflishch, (1995), Quasi-Monte Carlo integration. *Journal of Computational Physics*. 122, 218-230

[28] Nelson Areal, Artur Rodrigues, and Manuel R. Armada, (2008) On Improving the Least Squares Monte Carlo Option Valuation Method, *Review of Derivatives Research*, 11, 119-151.

[29] Neiderreiter, H. (1992) *Random Number Generation and Quasi-Monte Carlo Methods.* SIAM, Philadelphia, Pa.

[30] Schwartz, E. S. (1977) The Valuation of Warrants: Implementing a New Approach. *Journal of Financial Economics* 4, 79-93.

[31] Scott B. Nickleach (2008) Numerical Algorithms for Stock Option Valuation, Ph.D. dissertation, University of Pittsburgh.

[32] Steinbrecher, G., and Shaw, W.T. (2008). Quantile mechanics. *European Journal of Applied Mathematics* 19 (2): 87-112.

[33] Tilley, J.A, (1993), Valuing American Options in a Path Simulation Model. *Transactions of the Society of Actuaries.* 45, 83-104