A Practical Application of Monte Carlo Simulation for Options Pricing

Anubha Srivastava

Rythem Bajaj

Abstract

In this paper, an attempt has been made to describe a practical application of the Brownian-walk Monte Carlo simulation in option pricing. This simple Monte Carlo routine is useful in option pricing and forecasting productivity, installation rates, labour trends, etc. While Monte Carlo simulation is very useful and relevant to calculate the “P50 value” for contingency planning, the true strength of Monte Carlo simulation is in data extrapolation or forecasting. This paper throws light on some basic elements of Monte Carlo simulation approach for its application. The model can assist corporates to develop unique and accurate near-term market insights and trends in order to compete in the marketplace on analytics. Hence, in this paper, in particular, an attempt has been made to first study an options pricing (OP) model that produces an analytical solution, and then analyze two numerical options pricing models in terms of accuracy.

Key Words: *Monte Carlo simulation, options pricing, options pricing models, future development, Convergence test*

I - Introduction

1.1 Monte Carlo Simulation

Monte Carlo simulation (also known as the Monte Carlo Method) lets us see all the possible outcomes of our decisions and assess the impact of risk, allowing for better decision making under uncertainty. Monte Carlo simulation is a computerized mathematical technique that allows people to account for risk in quantitative analysis and decision making. Monte Carlo simulation furnishes the decision-maker with a range of possible outcomes and the probabilities that will occur for any choice of action. It shows the extreme possibilities—the outcomes of going for broke and for the most conservative decision—along with all possible consequences for middle-of-the-road decisions. The technique was first used by scientists working on the atom bomb; it was named after Monte Carlo, the Monaco resort town renowned for its casinos. Since its introduction in World War II, Monte Carlo simulation has been used to model a variety of physical and conceptual systems. Probability is a way to bracket the volatility of short-term forecasts (seemingly random data). Monte Carlo simulation is a specialized probability application that is no more than an equation where the variables have been replaced with a random number generator. In other words, Monte Carlo is another computer approximation routine or numerical method that replaces geometry, calculus, etc. A Monte Carlo simulation is a method for iteratively evaluating a deterministic model using sets of random numbers as inputs. This method is often used when the model is complex, non-linear, or involves more than just a couple uncertain parameters**.**

1.2 Monte Carlo Simulation Function - Monte Carlo simulation performs risk analysis by building models of possible results by substituting a range of values—a probability distribution—for any factor that has inherent uncertainty. It then calculates results over and over, each time using a different set of random values from the probability functions. Depending upon the number of uncertainties and the ranges specified for them, a Monte Carlo simulation could involve thousands or tens of thousands of recalculations before it is complete. Monte Carlo simulation produces distributions of possible outcome values. By using probability distributions, variables can have different probabilities of different outcomes occurring. Probability distributions are a much more realistic way of describing uncertainty in variables of a risk analysis. Common probability distributions include:

a. **Normal –**The user simply defines the mean or expected value and a standard deviation to describe the variation about the mean. Values in the middle near the mean are most likely to occur. It is symmetric and describes many natural phenomena such as people's heights. Examples of variables described by normal distributions include inflation rates and energy prices.

b. **Lognormal –** Values are positively skewed, not symmetric like a normal distribution. It is used to represent values that don't go below zero but have unlimited positive potential. Examples of variables described by lognormal distributions include real estate property values, stock prices, and oil reserves.

c. **Uniform –** All values have an equal chance of occurring, and the user simply defines the minimum and maximum. Examples of variables that could be uniformly distributed include manufacturing costs or future sales revenues for a new product.

d. **Triangular –** The user defines the minimum, most likely, and maximum values. Values around the most likely are more likely to occur. Variables that could be described by a triangular distribution include past sales history per unit of time and inventory levels.

e. **PERT–** The user defines the minimum, most likely, and maximum values, just like the triangular distribution. Values around the most likely are more likely to occur. However, values between the most likely and extremes are more likely to occur than the triangular; that is, the extremes are not as emphasized. An example of the use of a PERT distribution is to describe the duration of a task in a project management model.

f. **Discrete –** The user defines specific values that may occur and the likelihood of each. An example might be the results of a lawsuit: 20% chance of positive verdict, 30% change of negative verdict, 40% chance of settlement, and 10% chance of mistrial.

During a Monte Carlo simulation, values are sampled at random from the input probability distributions. Each set of samples is called iteration, and the resulting outcome from that sample is recorded. Monte Carlo simulation does this hundreds or thousands of times, and the result is a probability distribution of possible outcomes. In this way, Monte Carlo simulation provides a much more comprehensive view of what may happen. It tells us not only what could happen, but how likely it is to happen. Monte Carlo simulation provides a number of advantages over deterministic, or "single-point estimate" analysis:

· **Probabilistic Results.** Results show not only what could happen, but how likely each outcome is.

· **Graphical Results.** Because of the data a Monte Carlo simulation generates, it's easy to create graphs of different outcomes and their chances of occurrence. This is important for communicating findings to other stakeholders.

· **Sensitivity Analysis.** With just a few cases, deterministic analysis makes it difficult to see which variables impact the outcome the most. In Monte Carlo simulation, it's easy to see which inputs had the biggest effect on bottom-line results.

· **Scenario Analysis.** In deterministic models, it's very difficult to model different combinations of values for different inputs to see the effects of truly different scenarios. Using Monte Carlo simulation, analysts can see exactly which inputs had which values together when certain outcomes occurred. This is invaluable for pursuing further analysis.

· **Correlation of Inputs.** In Monte Carlo simulation, it's possible to model interdependent relationships between input variables. It's important for accuracy to represent how, in reality, when some factors go up, others go up or down accordingly.

II - LITERATURE REVIEW

In the world of finance, mathematical models can be used as approximations to value complex real market derivatives. The modelling of financial options gained popularity when *Fisher Black and Myron Scholes, 1973* introduced the Black and Scholes (BS) model, which later became the foundation of the literature on options pricing where various studies are made on extending the model and developing alternative approaches to the valuation of options. A recent literature by *Broadie and Detemple, 2004* focuses on the trends and development of financial options modelling with emphasis on the development of models that depart from the assumptions of the classic BS model, since empirical evidence suggests that the BS prices tend to differ from the market prices of options due to the assumption that sharp changes in stock prices are negligible (*MacBeth and Merville, 1979; Vasile and Armeanu, 2009*). Several modifications of the model have been made to reduce discrepancies between these assumptions and the real world. Examples are the extension of the BS model with illiquidity (*Cetin et al., 2004*), the inclusion of transaction costs through adjusting the volatility (*Leland, 1985*), and also extensions to include jump-diffusion models and stochastic volatility models. Since financial markets undergo stochastic fluctuations, numerical methods such as Monte Carlo (MC) methods become useful tools to price options. Alternatively, binomial methods are discrete numerical approaches that can value options at any point in time until expiration. The literature has also expanded beyond the basics of these numerical methods, such as *Giles (2007) his* work on improving efficiency by introducing a multilevel approach to the MC method, and most recently, *Kyoung and Hong (2011)* presented an improved binomial method that uses cell averages of payoffs around each node in addition to the standard method. Essentially in this literature, the goal is to improve both the accuracy and efficiency in approximating values of options pricing (OP) models. For the convenience of further discussion, the notations used throughout the paper are summarized below:

S	price of underlying asset
K	strike or exercise price
C	value of the European call option
r	risk-free interest rate
t	time in years
T	maturity date
μ	volatility of returns of the underlying asset
σ	drift rate
ρ	a probability measure

Common abbreviations used are:

GBM	Geometric Brownian motion
OP	Options pricing
BS	Black-Scholes
MC	Monte Carlo
PDE	partial differential equation
SDE	stochastic differential equation

III-RESEARCH DESIGN

3.1-Options - An *option* is a derivative security that grants the buyer of the option the right, but not the obligation, to buy or sell an underlying asset, *S* (such as a stock, a bond or an index portfolio) on or before an expiration date, *T,* for an *exercise or strike price*, *K*. A *call option* is the right to buy, while a *put option* gives the right to sell. Let's take an example of a call option. Say a company holds 100 shares of a stock priced at $20 each. An investor believing the price will go up in a month's time enters into a contract with the company to buy the stock at, say $19 after one month. All the investor needed to pay is the premium of (stock - strike) = 20-19 = $1 per share. If the price did go up on the exercise date, the investor will exercise the option and gain the profit of buying cheap and selling high in the market. If the price goes down, the contract will expire and becomes worthless so he will only lose the premium price he paid to enter into the contract in the first place.

3.2-Option Styles

Exercising the options can be of several styles and some common ones are listed below. The first two are plain vanilla options. The third option is a non-vanilla option and the rest are exotic options.

- European options: Options that can only be exercised on the expiration date.
- American options: Options that can be exercised on or before the expiration date.
- Bermudan options: Options that can be exercised at any fixed period of time.
- Asian options: Options whose payoff depends on the average underlying asset over a certain period of time.
- Barrier options: Options either come into existence after a barrier is breached (up-and-in or down-and-in) or drop out of existence as a result of breaching the barrier (up-and-out or down-and-out).
- Look back options: Options that depend on the minimum (for call) or maximum (for put) value of the stock price over a certain period of time.
- Digital options: Options whose payoff is fixed after the underlying asset exceeds the exercise price.

3.2 (i) - The Put-Call Parity: The no-arbitrage assumption, which places a bound on the options, is important for this principle so that the same payoff is maintained for both the call and put options. The idea is that if a portfolio containing a call option has the same payoff at expiration as a portfolio containing a put option, then they must have the same value at any given time before the expiration. This is known as the put-call parity. Let C and P be the value of the call and put options at any time t respectively, and let T be the time at expiration, K the strike price and S the stock price at time t. Then, the payoffs at expiration are:

$C = \max(S - K; 0)$; and

$P = \max(K - S; 0)$:

The payoff at expiry is

$$C - P = \max(S - K; 0) - \max(K - S; 0)$$

$$\begin{cases} (S - K) - 0 \text{ if } S \geq K \\ (0 - (K - S) \text{ if } S \leq K \\ \qquad = S - K \end{cases}$$

Finally, discounting the value of the portfolio, the put-call parity is defined as:

$C - P = S - K.\, e^{-r\,(T-t)};$

where r is the discounted risk-free rate. The basic idea of the put-call parity can be applied to the Black-Scholes model that values call and put options independently, which we will derive later.

3.2 (ii) Risk-Neutral Valuation - A risk-neutral measure is a measure applied to arbitrage-free options valuation where the growth rate μ is replaced by the risk-free rate r. For example, for a continuous-time measure, we define a stochastic process, that is, a geometric Brownian motion (GBM) with the following stochastic differential equation (SDE):

$dS = \mu Sdt + \sigma SdW;$ (2.1)

where σ is the volatility and W is a Brownian motion. To make the equation risk-neutral, dW is redefined with a new measure so that we get:

$$dW = dW' + \frac{\mu - r}{\sigma}.\, dt$$

This equation is a result of applying the *Girsanov* theorem, which calculates the likelihood ratio of the original measure and the risk-neutral measure (refer to *Seydel (2005)*). Hence, Equation (2.1) yields:

$$\begin{aligned} dS &= \mu Sdt + \sigma S(dW' - \frac{\mu - r}{\sigma}.\, dt) \\ &= \mu Sdt - \ \mu - r \ Sdt + \sigma SdW'; \\ &= rSdt + \sigma SdW' \end{aligned}$$

3.2 (iii) The Black-Scholes Model - The BS model is a classic example of an options pricing model that was developed by *Fisher Black and Myron Scholes* in their seminal work in 1973. Their approach to options pricing problems is to solve a partial differential equation (PDE) with a final condition at t = T to obtain a unique solution. The fundamental idea is to find a closed-form solution to the Black-Scholes PDE by first using the Ito's calculus from Ito's lemma to obtain the BS equation, and then transform it to the heat equation to get the unique solution, and finally transform the solution back to find the corresponding solution of the Black-Scholes PDE.

The Black-Scholes PDE is an important part of the BS model. This PDE describes the option over time and is used to obtain the BS formula for pricing options. The underlying asset is assumed to follow the GBM with an SDE as defined in Equation (2.1). Itô's lemma states that for the SDE defined and any twice differential function, *C*, of *S* and *t*, we have:

$$dC = \left(\mu S\frac{\partial C}{\partial S} + \frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2}\right) dt + \sigma S\frac{\partial C}{\partial S}dW \quad (2.2)$$

The Wiener process dW is random so we want to eliminate this variable in order to obtain the PDE. This can be achieved by constructing a portfolio ∏ consisting of a long call for an option and short Δ shares of the underlying asset. A long call is the purchase of a call option while a short call is the selling of the underlying asset. Therefore, the portfolio is defined as:

$$\prod = C - \Delta S$$

A small change in the portfolio for a time period of [t; t + Δt] results in:

$$d\prod = dC - \Delta dS$$

Applying Equations (2.1) and (2.2) into the equation yields:

$$d\Pi = \left(\mu S\frac{\partial C}{\partial S} + \frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2}\right)dt + rS\frac{\partial C}{\partial S}dW - \Delta Sdt + \sigma SdW \quad (2.3)$$

To eliminate any risk of price movement, we apply delta hedging, which simply means that $\Delta = \frac{\partial C}{\partial S}$, to Equation (2.3) to get:

$$d\prod = \left(\frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2}\right)dt$$

The assumption of no-arbitrage defines the rate of return of the portfolio as d∏ = r∏dt. Therefore, the Black-Scholes PDE is given by:

$$\frac{\partial C}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 C}{\partial S^2} + \frac{\partial C}{\partial S}rS - rC = 0$$

3.3 Purpose of Study - The original aim was to develop and evaluate a computational tool for simulating the binomial OP model and the Monte Carlo (MC) simulation for the valuation of European options. However, as the work progressed and as more background materials were gathered, a need to steer the project in a different direction came into light for several reasons as summarized below.

1. Current research interests in the area of financial modelling are prominent in finding ways to improve options pricing models, by introducing alternative mathematical methods to price options or by modifying current models. A project related to evaluating methods to improve options pricing models would make a good contribution to this research area.
2. Developing a computational tool requires a programming language that can call a plotting library or a separate tool for plotting static graphs. This initial set up had already proven to be time-consuming for the author so there is a risk of not completing the project in time if

an attempt is made to build a tool to price options.

3.4 Objectives of the study -

- To understand the different models proposed in options pricing and additional extensions or modifications made to improve the models.
- To implement the algorithms for the binomial OP model and the MC simulation.
- To test and evaluate the accuracies of the binomial and MC simulation relative to the solution obtained from the BS formula.
- To implement the algorithm for the multilevel MC simulation.
- To apply the multilevel MC simulation to the European option.
- To evaluate the efficiency of the multilevel MC simulation for the European options.

3.5 Methodology

This paper is divided into small groups of sub-projects, where for each group, we investigate the behaviour of an options pricing model in terms of how well it approximates a solution or converges towards one. The study begins with background research on options pricing models, which includes the Black Scholes (BS) model, binomial OP model and the MC simulation. The background reading also includes current research in improving these models. The next step is to study the behaviour of binomial options pricing model. This includes implementing the model and providing a test case to test the model. The evaluation is then done by comparing the accuracy of this model relative to the BS model. The MC simulation is then investigated. Similarly, the model is implemented and tested with the same test case for consistency. Again, the evaluation is carried out by showing convergence of this MC value to the BS value. Two discretized methods are then introduced: Euler and Milstein schemes, to which implementation, testing and evaluation are carried out. Finally, an improved MC method in terms of efficiency is investigated. Similarly, the model is implemented. Testing and evaluation includes finding the computational costs and the root mean-square error of the model. As an extension, the Milstein scheme is introduced to this model to further improve the efficiency. The same methodology applies for this multilevel Monte Carlo simulation with the Milstein scheme.

3.6 Limitation of research -

Since the focus of the paper is to investigate the behaviour of existing options pricing models, this research limits only to it and does not focus on developing any model. Time factor has also been a great constraint. Due to time constraint, several problems that have been originally planned were not carried out.

3.7 Managerial implication-

This research aims to evaluate the efficiency and behaviour of the model and hence, can be applied in the real world in a better and improved form. Further, we could take a different direction and look at extensions to the models. For example, we can reduce the assumptions of the models and introduce more complex methods to price the options. Therefore, the models presented in this project are standard models that can be readily applied in the real world. The binomial options pricing model proves to converge faster to the BS model compared to the MC simulation, although it is much less flexible due to the assumption that there are only two possible price movements. The multilevel methods introduced to the MC simulation shows an increased efficiency, albeit not by much, but with promising results. The application of the Bermudan option would make for an interesting case for future work.

IV - DATA ANALYSIS

Implement the algorithms for the binomial OP model. Test and evaluate the accuracies of the binomial relative to the solution obtained from the BS formula.

4.1 Binomial Options Pricing Model

The binomial OP model is a lattice tree model that approximates a continuous random walk in discrete time with a fixed number of periods. A direct relationship of this model with the BS model may not be immediately evident but in the case of European options, the binomial value converges to the BS value as the number of periods increase. This model shares the same basic assumptions as the BS model and assumes an asset price path that follows a GBM.

4.2 Valuing the Options

In essence, the binomial OP model divides the time line into m equally-spaced intervals, where for each period δt = T/m, the price either goes up by an up-factor u or down by a down-factor d. Thus, if the current stock price is S, the stock price at the next period is either Su or Sd. For the next period, Su goes up to Suu or down to Sud and similarly, Sd goes to Sdu or Sdd. Notice that the stock price recombines at this stage since Sdu = Sud as per Figure I; therefore, this reduces the number of possible prices so that after m periods, there are only m +1 possible prices. We next define values for the parameters u and d. The Cox, Ross and Rubenstein (1979) (CRR)[10] method assumes that u and d are determined by the volatility σ, such that:

$$u = e^{\sigma\sqrt{\delta t}}, \text{ and } d = \frac{1}{u} = e^{-\sigma\sqrt{\delta t}}$$

Another important assumption is the risk-neutrality measure. Under this assumption, an investor's risk preferences are not taken into account so therefore, we assume that the return on the investment is a risk-free interest rate r. The steps involved in finding the option value are quite straight forward. We shall define the steps for finding a call option C. For a one-period binomial tree, the option will be Cu if the stock price goes to Su and Cd if the stock price goes to Sd.

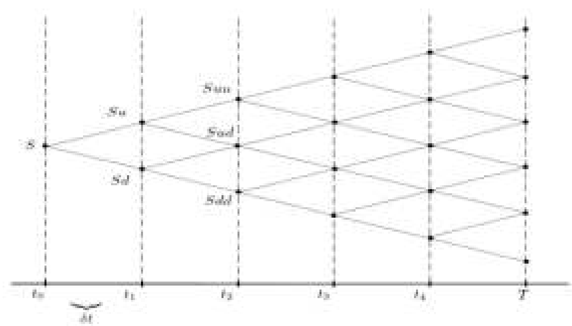


Figure I - A binomial tree with m = 5 of possible asset prices.

Hence, from the intrinsic value formula, we can define Cu and Cd as:

$$= \max(0; S_u - K);$$

$$C_d = (\max\ 0; S_d - K);$$

Suppose we build a portfolio that stores shares of a stock for investment. Let Δ be the number of shares and B be the price invested in the bonds of the stock. The portfolio payoff is thus ΔS + B. We can equate this to the option payoff, in this case the call option C, so that the up and down options become:

$$\Delta Su + e^{r\delta t}B = C_u, \text{ and} \quad (4.1)$$

$$\Delta Sd + e^{r\delta t}B = C_d. \quad (4.2)$$

Solving Equations (4.1) and (4.2), we find that:

$$\Delta = \frac{C_u - C_d}{(u-d)S} \text{ and } B = \frac{uC_d - dC_u}{(u-d)e^{r\delta t}}$$

Therefore, the call option is

$$C = \Delta S + B = \left[\frac{e^{r\delta t} - d}{u - d} C_u + \frac{u - e^{r\delta t}}{u - d} C_d\right] / e^{r\delta t}$$

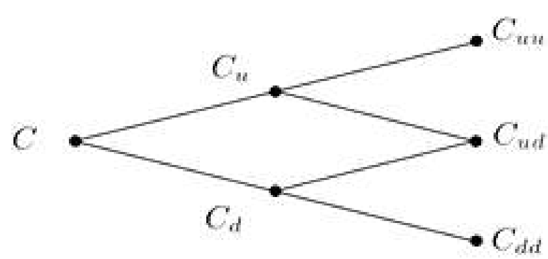


Figure II- Possible option prices for a 2-period binomial tree

To simplify the term, we let:

$$p = \frac{e^{r\delta t} - d}{u - d} \; and \; 1 - p = \frac{u - e^{r\delta t}}{u - d},$$

and hence, we can write the call option as:

$$C = [pC_u + (1-p)C_d]e^{-r\delta t}$$

Now we consider a call option with two periods. After the first period, C_u either goes up to C_{uu} or down to C_{ud} (Figure II). C_d is analogous. From the previous derivation, we find that:

$$C_u = [pC_{uu} + (1-p)C_{ud}]e^{-r\delta t}, \; and$$

$$C_d = [pC_{du} + (1-p)C_{dd}]e^{-r\delta t}.$$

Algorithm 4.2 : An algorithm for a binomial options pricing model.

```
function BinomialOPM (T, S , K, r , sigma , n) {
deltaT:= T/ n ;
up:=  exp( sigma * sqrt ( deltaT ) ) ;
down:=  1/up ;
cu:=  (up * exp(–r * deltaT) – exp(–q * deltaT ) )
       * up/(up^ 2 – 1 ) ;
cd:=  exp(–r * deltaT) – cu ;

for i := 0 to n { c ( i ):= S * up^ i * down^(n – I );
i f c ( i )< 0 then c ( i )=0;
}
for j := n–1 to 0 step – 1 {
        for i := 0 to j {
                c ( i ):= cu * c ( i )+ cd * c ( i +1);
    }
}
return c ( 0 ) ; }
```

Generally for m periods, the equation is given by:

$$C = \left[\sum_{j=a}^{m} \left(\frac{m!}{(m-j)!\,j!}\right) p^j (1-p)^{m-j} u^j d^{m-j} S - K\right] e^{-r\delta t} \quad (4.3)$$

where a is the minimum number of upward moves such that the strike price falls below the stock price upon expiry so that it can be exercised. In other words, we require that $u^a d^{m-a} S > K$.

In implementing the binomial model, the multi-period steps are computed recursively where the first step involves calculating the options at the terminal nodes and then working backwards to obtain the value of the first node. Algorithm 4.1 summarizes the steps in obtaining the binomial value of a call option. We can test the algorithm using the test case from the previous section where S = 250, K = 200, T = 1, r = 0:05 and σ = 0:2, with an additional parameter m for the number of periods. If we choose m to be 10 and run the algorithm, we obtain a binomial value of approximately 61:536162. Now that we have the binomial value, we want to verify that the result is correct so we compare it with the exact solution found from the BS formula. As it turns out, the binomial value converges to the BS value as the number of periods increase.

4.3 Convergence of the binomial OP model to the BS model

First, we investigate the relationship between the binomial OP model and the BS model. Equation (4.3) with m periods can be rewritten as:

$$C = S\left[\sum_{j=a}^{m}\frac{m!}{(m-j)!\,j!}p^{j}(1-p)^{m-j}\frac{u^{j}d^{m-j}}{e^{r\hat{\delta}t}}\right] - Ke^{-r\hat{\delta}t}\left[\sum_{j=a}^{m}\frac{m!}{(m-j)!\,j!}p^{j}(1-p)^{m-j}\right]$$

Replacing the two parts in parentheses with functions Ø(a; m; p') and Ø (a; m; p) respectively, we obtain a simpler equation of the form:

$$C = S[\emptyset(a;m;p')] - Ke^{-r\hat{\delta}t}[\emptyset(a;m;p)],$$

where $p' = ue^{-r\delta t}p$. From Cox, Ross and Rubenstein (1979)'s work on the convergence of the binomial formula to the BS formula, as m tends to infinity,

$$\emptyset(a;m;p') \rightarrow N(d_1)\ and\ \emptyset(a;m;p) \rightarrow N(d_2)$$

Hence, the BS formula is a limiting case of the binomial OP model.

Next, we investigate the convergence of the binomial OP model to the BS model. This can be easily demonstrated with a plot of the number of periods m against the option values found using the binomial OP model (Figure III). Cox, Ross and Rubenstein (1979)[11] provided a proof for the convergence as m tends to infinity. Their proof uses a special case of the central limit theorem which imposes restrictions on u and d. However, the proof provided is too specific. Hsia (1983))[12] applied a more general proof for the convergence of the Binomial OP model to the BS formula without restricting u and d, using the DeMoivre-Laplace limit theorem with the only condition being mp→∞ as m→∞. Qu (2010))[13] further demonstrated that there is a direct proof of the Binomial OP model converging to BS formula as m tends to infinity with the use of direct approximation of binomial probability from the normal distribution.

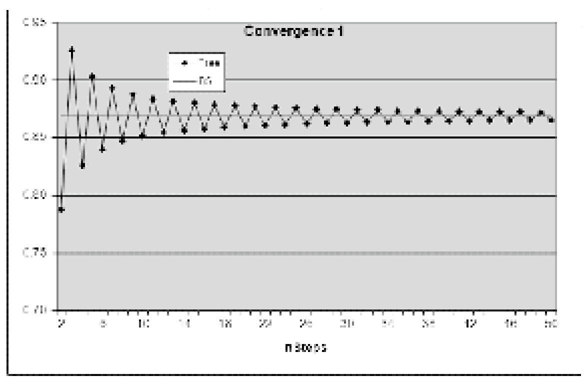


Figure III: A plot demonstrating the convergence of the binomial model to the BS model as m increases

Based on Chang and Palmer (2007)'s paper) [14], the rate of convergence from the Binomial OP model to the BS formula was found to be *1/m*. In the evaluation of the binomial OP model, we verify that this statement is true. Taking the same test case, we test the convergence with different values of *m*. Recall that the BS value was found to be 0.8689. For each value of *m*, we find the absolute error such that:

error = binomial value – BS value

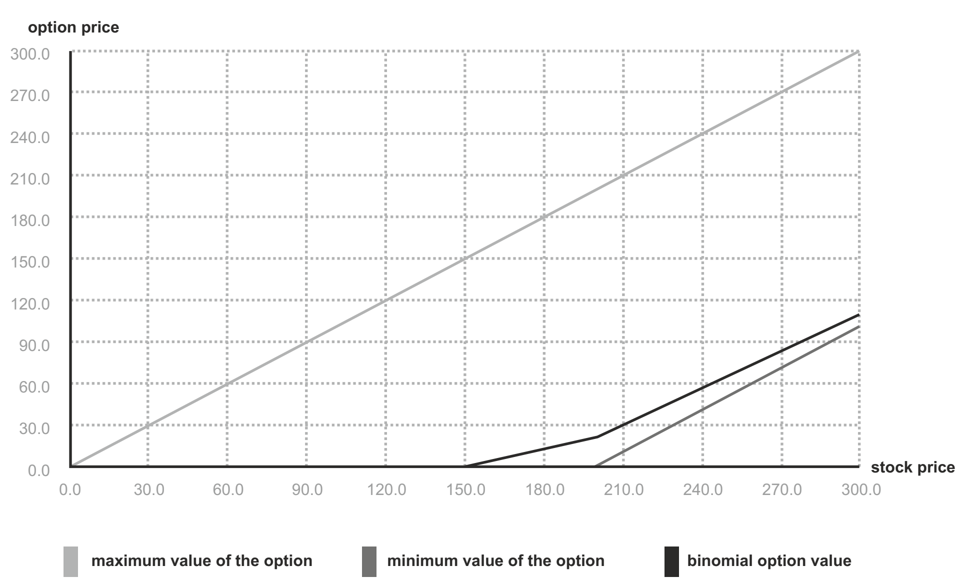
m	Binomial Value	|error|	1/m	Ratio=|error|/(1/m)
10	0.8513	0.017595	0.1	0.175947
50	0.8653	0.003559	0.02	0.17793
100	0.8671	0.001781	0.01	0.17809
200	0.8680	0.00089	0.005	0.178059
500	0.8685	0.000355	0.002	0.177693
1000	0.8685	0.000355	0.001	0.355386

Table I - Table of absolute errors for different m for a European call option.

The model is valid if its absolute error is proportional to the convergence rate, $|error| \propto \frac{1}{m}$

The result can be seen in Table I. Hence, the implementation of this model is correct.

Figure IV is a stock-option graph using the binomial model for n = 2 that illustrates how the option prices changes with different stock prices. Included in the graph are the maximum and minimum values. The minimum value of the option (intrinsic value) is the value at which a call option is in-the-money (i.e. the strike price is below the stock price). In other words, it is the actual value of the stock as opposed to the option value and is calculated by taking the difference between the strike price and the stock price. Option prices, on the other hand, are calculated using the equation $c = \left[\sum_{j=a}^{n} \binom{n}{j} p^j (1-p)^{n-j} (u^j d^{n-j} S - K)\right] e^{-rT}$. *where a is the smallest non-negative integer such that* $u^a d^{n-a} S>K$. The **time value** (extrinsic value) is the difference between the option price and the intrinsic value. As an option moves closer to maturity, the values of the options move closer to the intrinsic value, which means that the time value decays and eventually becomes worthless when it reaches maturity (Ugur, 2008))[15].



V - EMPIRICAL RESULTS

Implement the algorithms for the MC simulation.

Test and evaluate the accuracies of the MC simulation relative to the solution obtained from the BS formula.

5.1 **Monte Carlo Simulation** - Valuing of options is not limited to European and American options, which are the most basic styles of options. There are also exotic options with complicated features, such as the Asian option that takes the average underlying asset price over a predetermined period of time, and they cannot be easily valued using binomial OP model or the BS model due to their inflexibility in implementation. Therefore, in this chapter, we present another popular approach to valuing these options: **the Monte Carlo simulation**. This technique can easily simulate the stochastic process using random numbers and is flexible in terms of combining multiple sources of uncertainties.

Hence, it is practical for options that suffer from the curse of dimensionality, such as the real option. For the interest of this report, we will only apply the standard Monte Carlo simulation to European-style option to demonstrate its convergence to the BS model.

5.2 **Valuing the Options -** The MC simulation, which was first proposed by Boyle (1977), uses pseudo-random numbers to simulate price paths. It is a useful method to price options that has multiple uncertainties. We shall derive a sample path for the MC simulation. Recall in Equation (2.1) that the underlying asset is assumed to follow the GBM given by the SDE,

$$dS = \mu S dt + \sigma S dW,$$

where μ is the drift rate and σ is the volatility. Since the risk-neutrality assumption also applies here, we let μ = r, where r is the risk-free interest rate.

Algorithm 5.2: An algorithm for a standard Monte Carlo simulation.

```
function MonteCarlo {
        % m → number of time steps
        % n → number of simulation paths
        timestep := T/m;
        sum:= 0 ;
        for i := 1 to n {
                for j := 1 to m {
                        S = S * exp [ ( r – 0.5 * sigma
^2) * timestep + sigma * sqrt ( timestep ) * rand ] ;
                }
                        sum := sum + max(S–K, 0 ) ;
}
value := sum/n*exp(–r * timestep ) ;
return value ;
}
```

Using the properties of lognormal distribution, we let C = logS(t) and apply it to the Itô's formula to get:

$$d\log S(t) = \left(r - \frac{1}{2}\sigma^2\right)dt + \sigma dW$$

$$S(t) = S(t-1)\exp\left[\left(r - \frac{\sigma^2}{2}\right)\delta t + \sigma dW\right]$$

We generate the sample path for m periods by dividing the time period [0; T] into m intervals of δt to produce a sample path of:

$$S(t_j) = S(t_{j-1})\exp\left[\left(r - \frac{\sigma^2}{2}\right)\delta t + \sigma\sqrt{\delta t}Z_j\right], Z_j \sim N(0,1) \text{ and } j = 1, \ldots, m$$

The payoff, X(ω), for a European call option is max (S(t) – K; 0) for a sample path ω. To sample n asset price paths, we find the sample mean of the payoffs discounted to present using the risk-free rate, r, to obtain:

$$\bar{X} = \frac{1}{n}\sum_{i}^{n} X(\omega_i)e^{-rT}$$

(5.2)))This simple iteration can be seen in Algorithm 4.2. The pseudorandom number used for this implementation is a normally distributed value from the normal distribution N ~ (0, 1). (5.2)

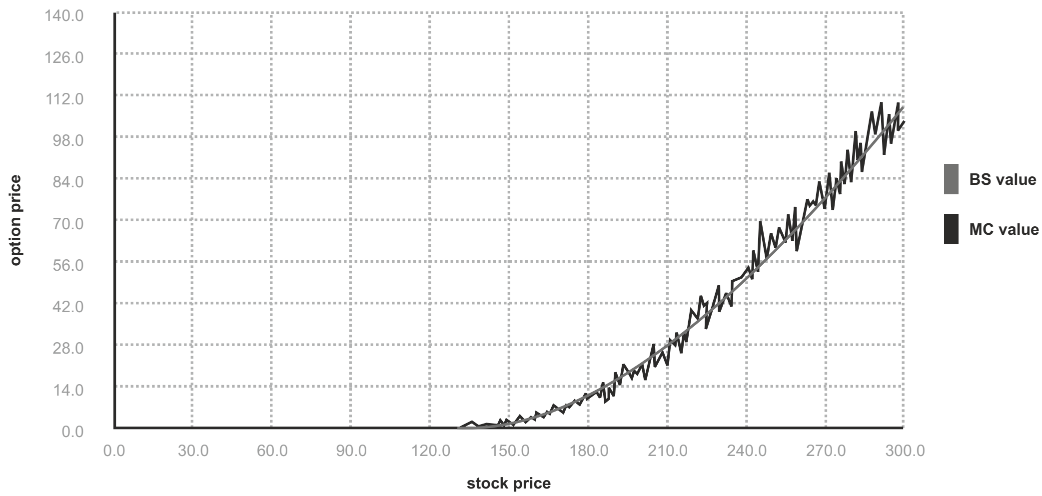


Figure V: A plot comparing the Monte Carlo simulation and the BS model for n = 100.

We begin our test with the test case we used in the earlier example. Setting the parameters S = 250, K = 200, r = 0:05 and σ = 0:2, we can generate a MC value with predetermined n and m. We demonstrate the results obtained from this simulation by plotting a graph of stock price against the option price. Figure 4.4 shows this result for n = 100 and m = 100, with comparison to the BS values.

6 Convergence Test - Convergence test suggests that the MC values tend to the BS closed-form solution when n increases. Although the convergence in Table II does not seem conclusive, there is an indication that the values are getting closer to the BS value of 61:4720918984744 as per Figure IV. For example, for $n=10^2$, the difference between the highest and the lowest MC values within the three seeds are approximately 9:494630, whereas for $n=10^6$, the difference is approximately 0:099661, indicating that there is a significant decrease in the standard deviation of the MC values as *n* increases.

		MC value **2.2.1**	2.2.2
2.2.3 2.2.4 2.2.5 2.2.6	**2.2.7**	**2.2.8**	**2.2.9**
n	**Seed 1**	**Seed 2**	**Seed 3**
2.2.10	2.2.11	2.2.12	2.2.13
10^2	59.1854875330700	55.66077308258579	65.15540374696462
10^3	62.3879080724478	60.420893879950015	60.806343515915735
10^4	61.3979056884734	60.716972462469144	61.06141350337705
10^5	61.5693015528772	61.44086461566363	61.612622484545696
10^6	61.5366498823280	61.43698915939219	61.502473728570706

Table II: A table of mean MC values with m = 1000 and different values of n.

To find the approximation error, we first need to calculate the estimated variance. Let *a = E (X) and b² = VarX* be the expectation of X and the variance respectively. If we obtain n samples X_i for i = 1, 2, …, n, then the approximation of a is:

$$\hat{a} = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Therefore, the estimated variance is:

$$b^2 = \frac{\sum_{i=1}^{n}(X_i - \hat{a})^2}{n-1}$$

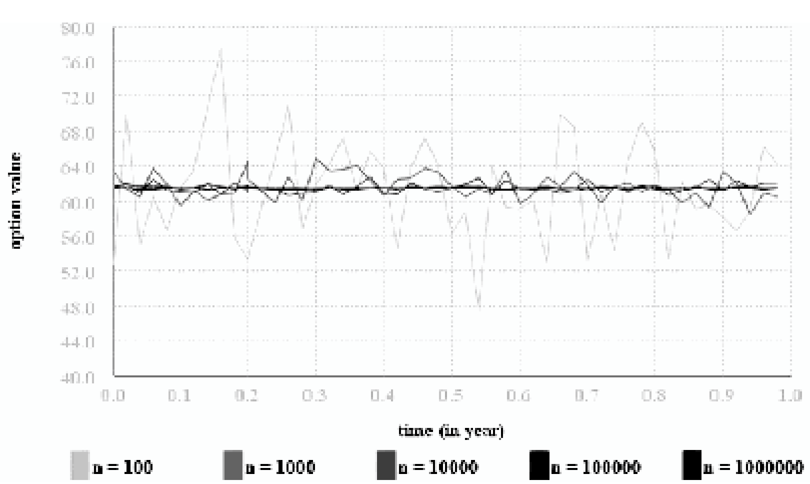


Figure VI: A plot of the Monte Carlo simulations for different values of n.

7 Simulation Process -

1. Create all necessary variables, including total number of simulations N, number of trading days each year n, expected return for asset using average of the five year realized annual return r, volatility sigma, asset value at day 0 (today) iniIndex, strike price Strike, risk free rate rf and all matrices we will need to store values.
2. Within each of the 252 steps for one simulation, we will generate a random number with normal distribution first. The variable "change" is calculated using the aforementioned equation.
3. In order for investors to be indifferent between holding the underlying asset and another asset generating 9.7% return as well, the probability for the underlying asset to go up is calculated as

   $$\frac{9.07\% - \text{down}}{\text{up} - \text{down}}$$

   in which up is defined as "change" when "change" is greater than 1 and down as "1/change", vice versa.
4. A random number is generated to determine if the index value will go up or down.
5. Repeat step 2-4 for the rest of the 251 steps.
6. Repeat step 2-5 for the rest of the N-1 simulations.
7. After finishing the simulation, we essentially have the performance of the asset in 10,000 different worlds. To determine when to exercise early, we will look at the index number at each step and calculate its future value. If the result is higher than the ending index number of that specific world, we call that step as an optimal early exercise time.
8. Divide the total number of optimal exercise opportunities by (n-1)*N to get the percentage amount of time investors will be better off by exercising their American call option early.

VI - EVALUATION

In this study, two types of evaluation have been discussed. They are: the evaluation of the models and the evaluation of the implementation. The first part gives a brief summary of all the evaluations of the models explained in earlier chapters. The next part explores the more technical side of the evaluation, giving insights on the choice of programming language used and then discussing the implementation of the algorithms.

6.1 Evaluation of Models -

Both the solutions obtained from the BS model and the binomial OP model can be easily verified by comparing them against option calculators. The MC simulation is much harder to compare since the model generates random values. Different test cases can simply be applied on our implementation of the BS model and that of the BS calculator to verify that the results are the same. Our result is accurate up to 15 decimal places. This calculator prints the entire binomial tree and hence, it is possible to verify that all values in our binomial tree are correct. Again, the result from this calculator is accurate up to 6 decimal places. However, a 6-decimal place accuracy of a result for comparison should provide enough information to verify that our model returns the correct result. Now, referring to the evaluation in earlier chapters, we verified that the binomial OP value converges to the BS value with an increase in the number of periods, as demonstrated. For the MC simulation, we rely on the BS model, which was found to produce correct results, to verify that the MC value is correct. The result is seen in Section 4.2, which explores the effect of increasing the number of simulation paths.

All implementation of the models are executed on the same machine to eliminate any error in the variation of the performances of different machines. The test cases are also kept consistent so that comparisons of results can be made between the models. For the choice of programming language, we do not require a fast language since we want to compare the performance of different models through observations of their execution times.

A round off issue may emerge especially in implementations where calculations are required in many runs. We rely on the precision of the double values used in the implementation, although round-off errors cannot be completely eliminated. The codes implemented in this project are not fully tested, each only having one test case since they are only used for analyzing. Nevertheless, the results from the test case are verified.

VII-CONCLUSION

In this report, we first outline two standard options pricing models and evaluate them based on their accuracy and efficiency, and then apply the multilevel MC simulation as a method to improve the efficiency of the MC simulation. The result for the binomial options pricing model shows that the accuracy relative to the BS model can be achieved with a large number of periods, m. The convergence rate from this result is verified as 1=m, which is a theoretical convergence rate presented by Chang and Palmer (2007))[16]. The MC OP model is the next model to be investigated. We test the standard MC simulation by initially running a simulation of n paths for three seeds. These three seeds are taken to show how the result varies when they are run on different number of paths n. Due to its random nature, there is a sampling error associated with taking random variables to estimate the payoff, which we want to minimize. The result shows that increasing the number of paths will reduce the sampling error at a rate of 1=pn. However, to achieve the level of accuracy of a binomial OP model takes a much greater computational effort for the MC simulation.

Since discretized methods are applied in the multilevel MC simulation, a study of these methods is necessary. The convergence order takes into account the bias due to discretization.

5.1 Future Work

For future work, another method we could consider is the Runge-Kutta method that simplifies the calculation of the asset price by replacing the derivative term of the Milstein scheme with a simpler term while still keeping the same convergence order. Application of the multilevel MC simulation to the Bermudan option proved to be quite a challenge since this option style is path-dependent. In addition, the multilevel method adds to the complexity of pricing the American option so it is also wise to price the American option using the standard MC simulation. One method to price options of this style is to use the Longstaff-Schwartz's least square approach (Longstaff and Schwartz, 2001))[17]. For the multilevel MC simulation, application to other exotic option styles such as barrier and look back options can be tested to observe the behaviour of this method in estimating payoffs of different option styles. Here, we also present ideas on potential areas of options pricing for future work. We have seen two standard models that price options numerically. We may use another popular numerical method, finite-difference methods, to compare its performance with the other models.

References

1. Boyle, Phelim P. 1977. "Options: A Monte Carlo approach." *Journal of Financial Economics* 4(3):323-338.
2. Brandimarte, Paolo. 2002. *Numerical Methods in Finance: A MATLAB-Based Introduction.* Wiley Series in Probability and Statistics John Wiley & Sons, Inc.
3. Broadie, Mark and Jerome B. Detemple. 2004. "Option Pricing: Valuation Models and Applications." *Management Science* 50(9):pp. 1145-1177.
4. Cetin, U., R. Jarrow, P. Protter and M.Warachka. 2004. "Pricing Options in an Extended Black Scholes Economy with Illiquidity: Theory and Empirical Evidence." *Review of Financial Studies* 19(2):493.
5. Chang, Lo-Bin and Ken Palmer. 2007. "Smooth convergence in the binomial model." *Finance and Stochastics* 11:91-105(15).
6. Cox, John C., Stephen Ross and Mark Rubenstein. 1979. "Option pricing: A simplified approach." *Journal of Financial Economics* 7(3):229-263.
7. Giles, Michael B. 2007. Improved multilevel Monte Carlo convergence using the Milstein scheme. *In Monte Carlo and Quasi-Monte Carlo Methods.* Springer pp. 343-358.
8. Giles, Michael B. 2008. "Multi-level Monte Carlo path simulation." *Operations Research* 56(3):607-617.
9. Glasserman, Paul. 2004. *Monte Carlo Methods in Financial Engineering.* Springer. Hsia, Chi Cheng. 1983. "On Binomial Option Pricing." *The Journal of Financial Research* 6:41-46.
10. Jabbour, George M and Yi Kang Liu. 2005. "Option Pricing and Monte Carlo Simulations."*Journal of Business and Economics Research* 3(9).
11. Kyoung, Sook Moon and Joong Kim Hong. 2011. "An Improved Binomial Method using Cell Averages for Option Pricing." *IEMS* 10(2):170-177.
12. Lee, Cheng-Few and Carl Shu-Ming Lin. 2010. Two Alternative Binomial Option Pricing Model Approaches to Derive Black-Scholes Option Pricing Model. In *Handbook of Quantitative Finance and Risk Management.*
13. Leland, Hayne E. 1985. "Option Pricing and Replication with Transaction Costs." *The Journal of Finance* 40(5):1283-1301.
14. Longsta, Francis A. and Eduardo S. Schwartz. 2001. "Valuing American Options by Simulation: A Simple Least-Squares Approach." *The Review of Financial Studies* 14(1):113-147.
15. MacBeth, James D. and Larry J. Merville. 1979. "An Empirical Examination of the Black-Scholes Call Option Pricing Model." *The Journal of Finance* 34(5):1173.
16. Milshtein, G. N. 1976. "A method of Second-Order Accuracy integration of Stochastic Differential Equations." *Theory of Probablility and its Applications* 23(2):396-401.
17. Palczewski, Jan. 2009. "Milstein Scheme and Convergence, Computations in Finance: MATH5350."
18. URL:*http://www.maths.leeds.ac.uk/ jp/leeds/CIF/CIFLecture6.pdf*
19. Qu, Xianggui. 2010. "A Direct Justification of the Binomial Pricing Model as an Approach of the Black-Scholes Formula." *Pakistan Journal of Statistics* 26(1):187-193.
20. Schaffter, Thomas. 2010. "Numerical Integration of SDEs: A Short Tutorial." Swiss Federal Institute of Technology in Lausanne (EPFL), Switzerland, Unpublished manuscript.

21. Schwartz, Eduardo. 1977. “The Valuation of Warrants: Implementing a New Approach.” *Journal of Financial Economics* 4(1):79-93.
22. Vasile, Emilia and Dan Armeanu. 2009. “Empirical Study on the Performances of Black-Scholes Model for Evaluating European Options.” *Romanian Journal of Economic Forecasting* 10(1):48-62.
23. Wilmott, Paul, Sam Howinson and Jeff Dewynne. 1998. *The Mathematics of Financial Derivatives.* Cambridge University Press.

Dr Anubha Srivastava *is associated with the Amity Business School, Noida as Assistant Professor (Finance & Accounts) since the last seven years and holds Ph.d and UGC NET. She has written nine research papers in the finance domain. She can be reached at asrivastava5@amity.edu*

Rythem Bajaj *is a student of the Amity Business School pursuing the Masters in Business Administration (MBA) program. She can be reached at rythembajaj@hotmail.co.uk*