Monte Carlo Simulation in European Call Option

Md Arafatur Rahman

Project Overview:

The primary objective of the project was to develop a Monte Carlo (MC) simulation model for European option pricing, a fundamental tool in financial engineering and risk management. This model has been used to estimate option prices, Greeks (delta and gamma), and implied volatility. The analysis of Monte Carlo estimation has been performed by drawing histogram, box plot and QQ plot etc. Additionally, the Black-Scholes (BS) model has been used to provide closed-form solutions for comparison with the Monte Carlo results.

Key Project Components:

1. Monte Carlo (MC) Simulation for Option Price:

The Monte Carlo method involves generating random paths for the underlying asset's price and applying option payoff functions at the option's maturity date. This allowed us to estimate option prices for various underlying asset scenarios.

1.1 Option pricing

The price of an option is the expected discounted payoff under the risk neutral measure [1]. The price V of a European option maturing at T is written as

$$V = e^{-rT}\mathrm{E}_r[\max(S_T - K, 0)] \tag{1}$$

Where S_T is the underlying asset price at the maturity T, K is a fixed price called as strike price, and the operator E_r denotes the expectation under the risk neutral measure. The log normal probability distribution for stock price model is given by

$$S_T = S_0 e^{(r-\frac{1}{2}\sigma^2)T+\sigma W_T}$$

Where, S_0 is the initial stock price, r is the risk-free interest rate, σ is the volatility of the stock, and W_T is the standard Brownian motion.

In our case study, we assumed the risk-free rate $r = 0$. Hence our stock price model is

$$S_T = S_0 e^{-\frac{1}{2}\sigma^2 T+\sigma W_T} \tag{2}$$

Now, equation (1) can be written as

$$V = \mathrm{E}_r[\mathrm{I}_{\{S_T>K\}}\,(S_T - K)] \tag{3}$$

Where $\mathrm{I}_{\{S_T>K\}}$ is the indicator function defined as $\mathrm{I}_{\{S_T>K\}} = \max(S_T - K) = \begin{cases} S_T - K & if\ S_T > K \\ 0 & otherwise \end{cases}$.

Equation (3) provides us with the theoretical option price. This formula can be employed in situations where we have a known distribution for the underlying asset and can compute the expectation in closed form. However, it's important to acknowledge that in many practical scenarios, obtaining a closed-form solution is not possible. Consequently, we apply the Monte Carlo Method. In that case we replace the expectation in equation (3) by the sample average as follows:

$$V = \frac{1}{N}\sum_{i=1}^{N} \mathrm{I}_{\{S_T^i > K\}}(S_T^i - K) \tag{4}$$

Where N is the number of samples.

The Monte Carlo simulation for option pricing follows the subsequent steps [3]:

- **Step 1:** Generate a large number of samples of S_T from the given model $S_T^i = S_0 e^{-\frac{1}{2}\sigma^2 T + \sigma\sqrt{T}Z^i}$ where Z^i are i.i.d standard normal i.e. $Z^i \sim N(0,1)$.
- **Step 2:** Calculate the payoff as $\mathrm{I}_{\{S_T^i > K\}}\left(S_T^i - K\right)$
- **Step 3:** Calculate the average payoff.
- **Step 4:** Apply the discounting (if any).

The values of the parameters in our simulation are $S_0 = 40, \sigma = 0.25, T = 1, K = 50$.

Typically, the Monte Carlo method is effective when we consider a large number of samples. In the results section, we have demonstrated the performance of our estimations based on different sample sizes.

2. Monte Carlo for Option Greeks (Delta and Gamma):

In addition to option pricing, option Greeks (delta and gamma) has been calculated. Delta measures the sensitivity of the option price to changes in the underlying asset price, while gamma measures the rate of change of delta with respect to the asset price [2].

2.1 Calculating option delta.

Mathematically, option delta is defined as the first order derivative of option price w.r.t the stock price, i.e. $delta = \frac{\partial V}{\partial S}$. The delta of an option at time zero is $delta = \frac{\partial V}{\partial S_0}$. To compute the option delta using the Monte Carlo method, we have two approaches: we can either utilize a finite difference method to approximate the partial derivative, or we can directly calculate the partial derivative. In this case study, I opted for the latter approach, directly computing the partial derivative. Equation (4) can be written as,

$$V = \frac{1}{N}\sum_{i=1}^{N} \mathrm{I}_{\{S_T^i > K\}}(S_0 e^{-\frac{1}{2}\sigma^2 T + \sigma\sqrt{T}Z^i} - K) \tag{5}$$

By taking the derivative of V w. r. t. S_0 we have,

$$\frac{\partial V}{\partial S_0} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{I}_{\{S_T^i > K\}} e^{-\frac{1}{2}\sigma^2 T + \sigma\sqrt{T} Z^i} \tag{6}$$

Now, we can employ Equation (6) as our Monte Carlo method for computing delta. It is important to note that to calculate delta, I interchanged the derivative operator inside the summation, which may not be applicable in a general case. This simplification assumes that we have a well-behaved function within the summation, allowing us to interchange the order of these operators. Further mathematical justification may be required for more complex cases.

2.2 Calculating Option Gamma

Gamma is the second order partial derivative of option price w. r. t. the stock price. Mathematically, it is written as $\text{gamma} = \frac{\partial^2 V}{\partial S_0^2}$. To compute gamma, we cannot apply derivative approaches as the second derivative becomes zero. A central difference formula on the option price V has been used to approximate gamma which is given as

$$\text{gamma} = \frac{\mathrm{V}(\mathrm{S}_0 + \Delta S) - 2V(S_0) + V(S_0 - \Delta S)}{\Delta S^2} \tag{7}$$

Where ΔS is a very small price change. As ΔS approaches to 0 we will get the exact value for gamma. It is essential to highlight that the randomness of MC simulation remained constant across all option price V (forward, backward, and central) in equation (7). Instead of writing gamma in terms of option price we can also approximate it by taking forward or backward differences on option delta.

3. Monte Carlo Method for Implied Volatility:

Implied volatility holds a crucial position in option pricing models. It represents the volatility of the option that aligns the estimated model price with the market price [4]. Calculating implied volatility necessitates finding the root of a non-linear equation. Various root-finding methods are available, but in this study, the Newton-Raphson method was employed due to its second-order rate of convergence. This approach entailed an iterative process to determine the implied volatility that equalizes the observed option price with the model's price. It is essential to highlight that the randomness of MC simulation remained constant across each Newton-Raphson iteration. This approach was crucial in maintaining stability during the iterative process.

3.1 Newton Method

Suppose we want to find the root of a non-linear equation $f(x) = 0$. The iterative formula for Newton-Raphson method is given by [5]

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)} \quad \text{with a initial guess } x_0 \tag{8}$$

We continue until $|x_{k+1} - x_k| \leq some\ given\ tolerance$. This method fails when $f'(x_k) = 0$.

3.2 Implied volatility by Newton method

The equation for implied volatility is given by

$$f(\sigma) = V(\sigma) - \bar{\bar{V}} = 0 \tag{9}$$

Where $\bar{\bar{V}}$ is the option market price which is a constant.

By taking the derivative w. r. t. σ we have

$$f'(\sigma) = V'(\sigma) = \frac{1}{N}\sum_{i=1}^{N} \mathrm{I}_{\{S_T^i > K\}} e^{-\frac{1}{2}\sigma^2 T + \sigma\sqrt{T}Z^i} \; (-\sigma T + \sqrt{T}Z^i) \tag{10}$$

The derivative of option price w. r. t. volatility is another Greeks known as Vega. Now we can use this derivative in Newton Raphson iterative formula to obtain our implied volatility as follows:

$$\sigma_{k+1} = \sigma_k - \frac{V(\sigma)}{V'(\sigma)} \tag{11}$$

4. Closed-Form Solutions

For comparison, the Black-Scholes (BS) model was also implemented to calculate option prices, delta, gamma, and implied volatility as the closed-form solutions. This allowed the Monte Carlo results to be validated against well-established theoretical models.

4.1 Black Scholes option price

The price of a European call option from BS model is given by:

$$V = N(d_1)S_0 - N(d_2)Ke^{-rT} \tag{12}$$

Where $N(\cdot)$ is the cumulative density function for normal distribution and d_1 & d_2 is given by

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

$$d_2 = d_1 - \sigma\sqrt{T}$$

4.2 Black Scholes option delta and gamma

The delta of a European call option from BS model is given as:

$$delta = N(d_1) \tag{13}$$

The gamma of a European call option from BS model is given as:

$gamma = \frac{N'(d_1)}{\sigma S_0 \sqrt{T}}$ where $N'(\cdot)$ is the pdf for the normal distribution (14)

5. Results

In this section, the results for all the tasks have been shown. The option price, delta, gamma, and implied volatility were estimated using the Monte Carlo method, and a comparison was made with the results obtained from the closed-form solution provided by the Black-Scholes model. To evaluate the accuracy and performance of the Monte Carlo estimates, a series of experiments were conducted, encompassing the use of box plots, histograms, and Q-Q plots. These graphical representations facilitated the visualization of the distribution and behavior of the estimated results under various scenarios.

5.1 Task 1(a): Stock Price simulation by Monte Carlo

The MC price path for a one-year period (252 days) was calibrated using the following discretized formula:

$$S_{t+1} = S_t e^{-\frac{1}{2}\sigma^2 dt + \sigma\sqrt{dt} Z^{t+1}} \text{ where } dt = \frac{T}{252} \tag{15}$$

The simulated price is shown in the following figure:

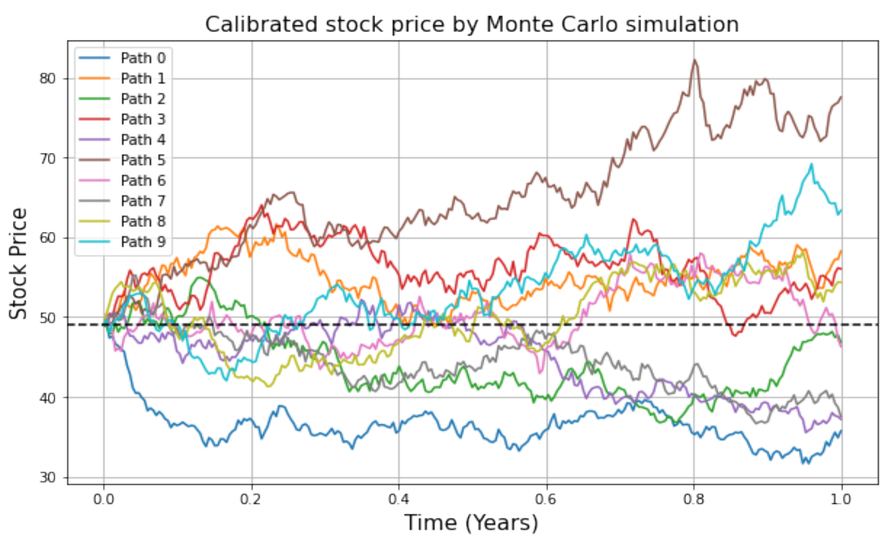

Fig 1: Stock price simulation by Monte Carlo method.

The price path displayed in Figure 1 corresponds to 10 sample paths. To enhance the efficiency of the implementation, the simulation was conducted in a vectorized form.

To verify that our simulation follows a log normal distribution, some additional experiments have been conducted. First, I simulated the terminal price S_T for $N = 10000$ samples.

Then I calculated the log return as follows:

$$\ln\left(\frac{S_T}{S_0}\right) = -\frac{1}{2}\sigma^2 T + \sigma W_T$$

Theoretically, this log return follows a normal distribution with a mean of $-\frac{1}{2}\sigma^2 T$ and variance $\sigma^2 T$.

The comparison of mean and variance between our simulation and Theoretical distribution is shown in the following table.

Table1: Mean and variance of the simulated and theoretical distribution.

Parameters	Theoretical	Simulated
Mean	-0.03125	-0.02806
Variance	0.0625	0.0627

The simulated and theoretical parameters exhibit a close alignment, suggesting a good match. Further improvement can be achieved by increasing the number of samples, N.

Furthermore, a histogram and a QQ plot were created for visual inspection, as depicted in Fig 2.

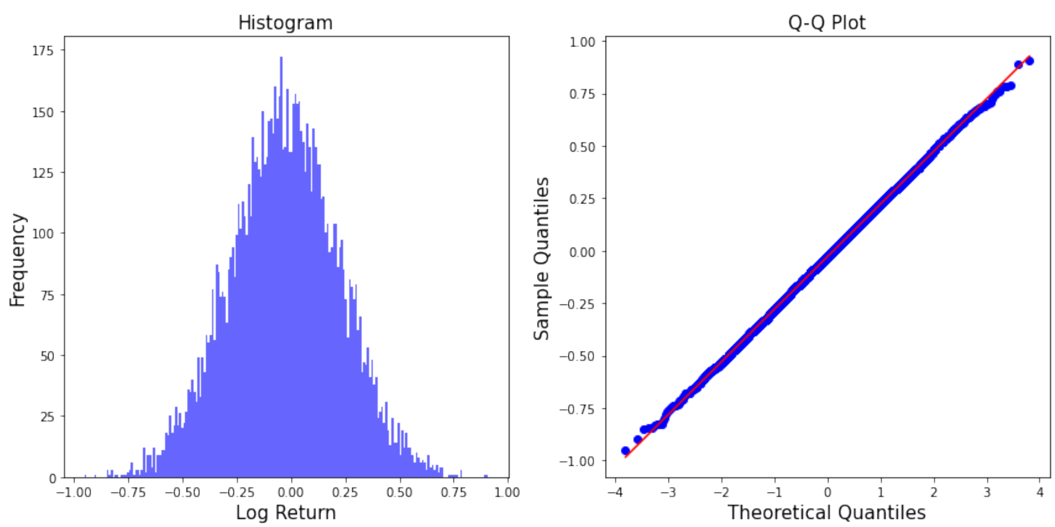

Fig 2: A histogram (on left) and QQ plot (on right) for the log return $\ln\left(\frac{S_T}{S_0}\right)$.

From the histogram and QQ plot we can say that our simulated stock price follows log normal distribution.

5.2 Task 1(b): Option Pricing by Monte Carlo

By following the steps in sec 1.1, the MC option price has been estimated with the parameters $S_0 = 49, K = 50, r = 0, T = 1, \sigma = 0.25$. It's noteworthy that for pricing, there's no need to generate the entire sample path. Instead, we can simply generate the stock price at maturity, denoted as S_T, which enhances the efficiency of the implementation.

The estimated results for a sample size of $N = 10^8$ are as follows:

Monte Carlo estimated price : 4.4403

Exact price from the Black Scholes : 4.44007

The computational time for Monte Carlo price was 3.93 sec. The Python code was run on a local machine with the following configuration: Processor -Intel(R) Core(TM) i7-10700 CPU @ 2.90GHz 2.90 GHz; Ram - 16.0 GB (15.8 GB usable).

It has been observed that the Monte Carlo estimation performance improves as the number of samples increases. Additionally, it's important to recognize that each estimate is a random value, as it is based on calculating the sample mean. To explore the relationship between sample sizes and estimation accuracy, an experiment was designed. In this experiment, $M = 50$ estimates were calculated for different numbers of samples, and the mean of these 50 estimates was determined. The results are presented in Table 2 and Fig 3.

Table2: Monte Carlo estimated price with different sample sizes.

Num of Samples N	**Estimated Price (Mean of 50 $Estimates$)**	**Running time**
10^6	4.43889 (10^{-3})	2.001 Sec
10^7	4.44044 ($3.7 * 10^{-4}$)	20.57 sec
10^8	4.44015 ($8 * 10^{-5}$)	199.88 sec

The numbers in the parentheses represent the errors (|MC estimated price - BS price|) for the estimates. It is evident that as the sample size increases, the Monte Carlo estimates improve in accuracy. However, it's important to note that this improvement comes at the cost of increased computational time.

To better understand our estimated results, a box pot has been presented in Fig 3.

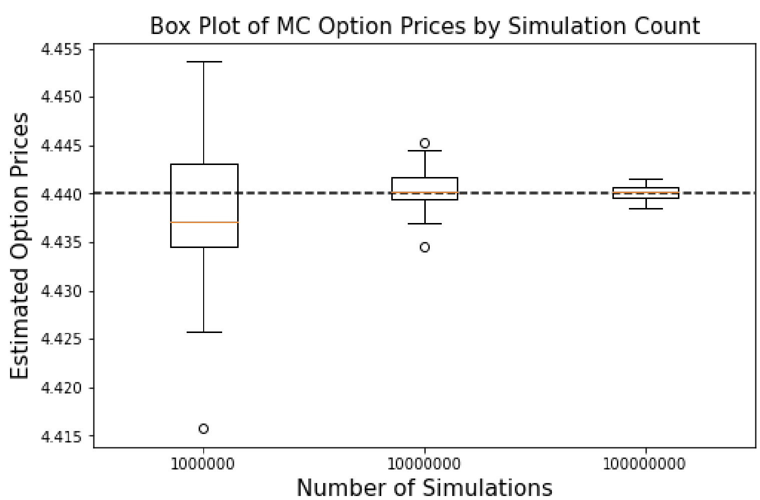

Fig 3: Box plot for estimated option price.

It's evident that as the number of simulations or sample size, N, increases, our estimated results converge towards the exact solution. The narrowing of the box in the box plots indicates reduced deviations from the estimated mean, further illustrating the improved accuracy of the estimates.

5.3 Task 2: Option sensitivities (Delta and Gamma)

The results for estimated delta of a sample size of 10^8 are as follows:

Monte Carlo delta	0.51758
Black Scholes delta	0.5176

Gamma has been estimated by a central difference formula with different price increments. The results are shown in Table 3.

Table 3: Estimated gamma with different price increments.

Price Increments ΔS	MC Estimated gamma	BS Gamma
0.5	0.03254	0.03253
1.0	0.03252	
2.0	0.03247	

We can see that the finite difference method has a better approximation when ΔS is smaller.

5.4 Task 3: Implied volatility

The implied volatility for the Monte Carlo and Black Scholes is shown in Table 4. The number of samples we considered were $N = 10^8$ and the tolerance for Newton method was 10^{-8}.

Models	Implied Volatility
Monte Carlo	0.33011
Black Scholes	0.33004

For both Monte Carlo and Black Scholes, the number of iterations needed for the Newton method to converge were 2 which indicates faster rate of convergences.

6. Key Findings and Conclusions:

The project was successful in achieving the following key outcomes:

- The Monte Carlo simulation provided reliable estimates of European option prices, consistent with the Black-Scholes closed-form results.
- Delta and gamma calculations were accurate and consistent with closed form results.
- The implied volatility estimation using the Monte Carlo was consistent with Black Scholes results. The Newton-Raphson method proved effective in determining the numerical solutions for finding root of a non-linear equations.
- The experiments, including box plots, histograms, and Q-Q plots, demonstrated the robustness and reliability of the Monte Carlo simulation for option pricing.
- Monte Carlo estimation gets better as the number of samples increases.

References:

1. Hull, J. Options, Futures, and Other Derivatives. Pearson, Jan 2017.
2. Shreve, S. Stochastic Calculus for Finance II: Continuous-Time Models. Springer, Dec 2010.
3. Glasserman, P. Monte Carlo Methods in Financial Engineering. Springer, 2003.
4. Investopedia, NY City, 1999, accessed on Nov 4, 2023, <www.investopedia.com>
5. Free Encyclopedia, Accessed on Nov 4, 2023, <www.wikipedia.com>