

- Final Project - European Call Options

Tim Jaschek

March 3, 2016

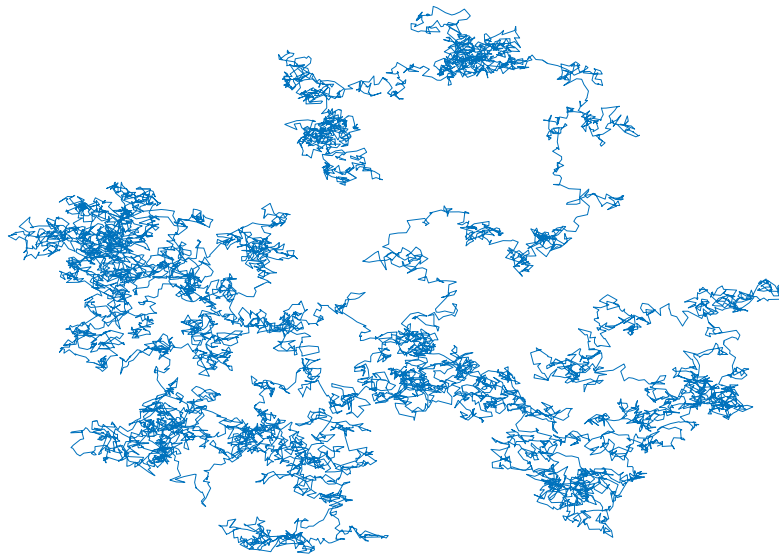


Figure 1: Simulation of a two dimensional Brownian Motion, Brownian Motion is a key element in the analysis of stochastic financial mathematics

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction and explanation of Task | 3 |
| 1.1 | Definition of european call options | 3 |
| 1.2 | Mathematical modelation | 3 |
| 2 | Preliminars about Monte Carlo method and variance reduction | 4 |
| 2.1 | Confidence intervals | 5 |
| 2.2 | Variance reduction methods | 6 |
| 2.2.1 | Control variates | 6 |
| 2.2.2 | Importance sampling | 6 |
| 2.2.3 | Stratified sampling | 7 |
| 3 | Simple plain Monte Carlo estimation | 7 |
| 3.1 | Using explicit formula for geometric Brownian motion | 7 |
| 3.2 | Using Euler scheme | 8 |
| 4 | Analytical solution | 8 |
| 5 | Control variates method | 9 |
| 6 | Importance sampling method | 9 |
| 6.1 | Estimation with the given value for μ | 10 |
| 6.2 | Find the optimal μ | 10 |
| 7 | Stratified sampling method | 11 |
| 7.1 | Estimation with 500 equiprobable strata | 11 |
| 7.2 | Standard derivation for different strata | 11 |
| 8 | Comparison of the manifold methods | 11 |

1 Introduction and explanation of Task

The aim of this thesis is to estimate the value of a special financial contract, called *European call option*. These products occur on the financial market for different assets like stocks, precious metals or raw materials. To give an estimation with the aim of a minimal variance I will introduce and use different mathematical methods. The thesis is part of the final evaluation in the course stochastic computations at Humboldt University Berlin in the Winter term of 2015/16 taught by Dr. Archef Bachouch.

1.1 Definition of european call options

A european call option is a financial contract granting its holder the right (without the obligation) to buy an agreed quantity of an underlying asset at a fixed price K (called the *strike price*) at a fixed future time T (called *expiry date* or *maturity*). In the following I will assume the underlying asset to be a stock.

1.2 Mathematical modelation

Regarding the underlying stock S of our call option, we can consider its price S_t at time $t \geq 0$, where the current time is $t_0 = 0$. If at time T , S_T is higher than K , then the holder exercises his call option for a profit $S_T - K$ (he buys a stock for the price K and sells it for a price of S_T). Whereas in the case that S_T is smaller than K the holder of the call option won't use it. So the profit of the holder is $(S_t - K)_+$. Obviously in times where we assume the stock prices to increase nobody would offer such a contract. Therefore a call option is connected with a premium C that one has to pay to buy this contract. We assume the following formula:

$$C := \theta = e^{-rT} \mathbb{E}[(S_t - K)_+]$$

where r denotes the *continuously compounded interest rate*.

Note that the above formula makes sense because we can consider S as a stochastic process and S_T as a real valued random variable. More assumptions concerning the distribution of S_T will be made later on. Now one can say more precisely that the aim of the thesis is to estimate θ with different methods.

In many parts of this thesis I will assume that we are in the context of the Black-Scholes model so that the stock price S is described by the geometric Brownian motion defined by:

$$S_t := S_0 \exp \left(\left(r - \frac{\sigma^2}{2} \right) t + \sigma W_t \right) \quad t \in [0, T] \quad (1)$$

where W is a standard Brownian motion and σ is the volatility of the asset.

If we assume this formula we can compute the premium of our European call option

analytically. Using equation 1 we have an explicit formula for S_T such that:

$$\begin{aligned} C &= e^{-rT} \mathbb{E} [(S_T - K)_+] \\ &= e^{-rT} \int_{\Omega} \left(S_0 e^{\left(r - \frac{\sigma^2}{2}\right)T + \sigma W_T(\omega)} - K \right)_+ \mathbb{P}(d\omega) \\ &= e^{-rT} \int_{\Omega} \left(S_0 e^{\left(r - \frac{\sigma^2}{2}\right)T + \sigma W_T(\omega)} - K \right) \mathbb{1}_{\{S_T \geq K\}}(\omega) \mathbb{P}(d\omega) \end{aligned}$$

Since W is a standard Brownian motion we have $W_T \sim \mathcal{N}(0, T)$. Knowing this density and considering the following inequality for the indicator function we will be able to get an explicit formula for C .

$$\begin{aligned} S_0 e^{\left(r - \frac{\sigma^2}{2}\right)T + \sigma x} &\geq K \\ r - \frac{\sigma^2}{2}T + \sigma x &\geq \log \frac{K}{S_0} \\ x &\leq \left(\log \frac{S_0}{K} + r - \frac{\sigma^2}{2}T \right) \sigma^{-1} \end{aligned}$$

Together with the above we get:

$$C = e^{-rT} \int_{-\infty}^{\left(\log \frac{S_0}{K} + r - \frac{\sigma^2}{2}T \right) \sigma^{-1}} \left(S_0 e^{\left(r - \frac{\sigma^2}{2}\right)T + \sigma W_T(\omega)} - K \right) \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} dx$$

A change of variable and some formations will now give us the formula:

$$C = S_0 \Phi(d_1) - K e^{-rT} \Phi(d_2) \quad (2)$$

where $d_1 = \frac{1}{\sigma\sqrt{t}} \left(\log \frac{S_0}{K} + \left(r + \frac{\sigma^2}{2}\right)T \right)$ and $d_2 = d_1 - \sigma\sqrt{T}$ and where Φ is the CDF of standard normal distribution.

2 Preliminars about Monte Carlo method and variance reduction

The main method in estimating an expectation of a random variable is the so called Monte Carlo method. Therefore one considers a random variable X and $(X_n)_{n \in \mathbb{N}}$ i.i.d. copies of this random variable. The great definition of the N-th Monte Carlo Estimator μ_N^{MC} of $\mu = \mathbb{E}[X]$ is $\mu_N^{MC} = \frac{1}{N} \sum_{n=1}^N X_n$. The rate of convergence of this method can be explained by the well known Central Limit Theorem (CLT) and by thinking about confidence intervals.

2.1 Confidence intervals

Having computed a Monte Carlo estimator μ_N^{MC} without talking about confidence intervals is pretty useless. A confidence interval I_{C_α} is an symmetric interval around μ_N^{MC} such that the analytic solution is with a probability of α contained in this interval:

$$\mathbb{P}(\mathbb{E}[X] \in I_{C_\alpha}) = \mathbb{P}(\mathbb{E}[X] \in [\mu_N^{MC} - C_\alpha, \mu_N^{MC} + C_\alpha]) = \mathbb{P}(|\mathbb{E}[X] - \mu_N^{MC}| \leq C_\alpha) = \alpha$$

Here we will explain how the central limit theorem gives us a rate of convergence in the LLN. Since our given random variables $(X_i)_i$ are i.i.d. and square integrable with $\text{Var}[X_i] = \sigma^2 \quad \forall i \in \mathbb{N}$ the CLT yields and gives us:

$$\lim_{N \rightarrow \infty} \sqrt{N}(\mu_N^{MC} - \mathbb{E}[X]) \sim N(0, \sigma^2)$$

Hence for $a, b \in \mathbb{R}$:

$$\lim_{N \rightarrow \infty} P(a < \sqrt{N} \frac{(\mu_N^{MC} - \mathbb{E}[X])}{\sigma} < b) = P(a < Z < b)$$

for a random variable $Z \sim N(0, 1)$. Using this we can define Confidence Intervals:

$$\begin{aligned} P(-a < \sqrt{N} \frac{\mu_N^{MC} - \mathbb{E}[X]}{\sigma} < a) &= P\left(\mu_N^{MC} - \frac{a\sigma}{\sqrt{N}} < \mathbb{E}[X] < \mu_N^{MC} + \frac{a\sigma}{\sqrt{N}}\right) \\ &\approx P(-a < Z < a) \end{aligned}$$

If a fixed confidence level $\alpha \in [0, 1]$ is given, we will find $C_\alpha \in \mathbb{R}$ such that $P(|Z| \leq C_\alpha)$ and get the confidence Interval:

$$I_{\alpha, N} = \left[\mu_N^{MC} - \frac{C_\alpha \sigma}{\sqrt{N}}, \mu_N^{MC} + \frac{C_\alpha \sigma}{\sqrt{N}} \right]$$

Sending N to infinity will make this interval pretty small, so we can see the convergence of μ_N^{MC} to $\mathbb{E}[X]$. In the following I will use that for $\alpha = 0.95$ we get $C_\alpha = 1.96$. All confidence intervals that will be compute in the program are the 95% confidence intervals with the above C_α .

Being not always known in advantage, the variance of our random variable has to be estimated as well. Computation is possible by the numerical formula

$$\sigma \approx \sqrt{\frac{1}{(N-1)} \sum_{i=1}^N (X_i - \mu_N^{MC})^2}$$

known from the lecture of Dr. Archef Bachouch. The estimation error of the variance will not be considered. However, there must be an effect of this error to the confidence interval so one should always keep in mind that these are just estimations of the real confidence intervals.

2.2 Variance reduction methods

A simple Monte Carlo simulation can often converge relatively slowly. We can do some mathematical tricks to reduce the variance in special cases. Some methods will be explained here.

2.2.1 Control variates

The Idea of the control variates method is that if we have $\theta = \mathbb{E}[X] = \mathbb{E}[f(Z)]$ for a random variable Z that can be easily computed and where we know $\mathbb{E}[Z]$ we can compute θ in the following way:

- set $W_c := X + c(Z - \mathbb{E}[Z])$
- compute by Monte Carlo $\mathbb{E}[W_c]$

To see that this relay works just note that:

$$\mathbb{E}[W_c] = \mathbb{E}[X + c(Z - \mathbb{E}[Z])] = \mathbb{E}[X] + c\mathbb{E}[Z] - c\mathbb{E}[Z] = \mathbb{E}[X]$$

It can be shown easily that the optimal choice for c is $c^* = -\frac{\text{Cov}[X,Z]}{\text{Var}[Z]}$. Since this covariance and variance are often not known in advantage we have to use approximations for them. Therefore we do some plain Monte Carlo estimations but here we take just a small integer number of realisations.

2.2.2 Importance sampling

This method bases on a change of desity function. Let X be a randomvariable with desity function ψ_X and let ϕ be another density such that $\phi(x) \neq 0$ if $\psi_X \neq 0$. Let V be a random variable having ϕ as it's density. Then $\mathcal{L}(X) \ll \mathcal{L}(Y)$ and we get by the Radon Nikodym theorem the Radon Nikodym density $\frac{\psi_X}{\phi}$. Have the following equality:

$$\begin{aligned} \theta &= \mathbb{E}[f(X)] \\ &= \int f(x)\psi_X(x)dx \\ &= \int f(x)\frac{\psi_X(x)}{\phi(x)}\phi(x)dx \\ &= \mathbb{E}[h(Y)] \end{aligned}$$

where $h(x) = f(x)\frac{\psi_X(x)}{\phi(x)}$.

Like in the control variates method one is interested in the variance minimizing pdf ϕ^* . It can be shown that

$$\phi^*(x) = \frac{|f(x)|\psi_X(x)}{\int |f(z)|\psi_X(z)dz}$$

minimizes the variance.

This relation is known as the *Rubinstein Theorem*. Briefly one can say that the optimal density is a scaled version of the absolute part of our function multiplied with the density of the given random variable.

2.2.3 Stratified sampling

The last method I want to introduce is the stratified sampling. Consider the following: Let Y be a random variable. Again our aim is to estimate $\theta = \mathbb{E}[Y]$. Therefore let W be a real valued random variable and $(\Delta_i)_{1 \leq i \leq m}$ a finite partition of \mathbb{R} . If we set $I := \sum_{i=1}^m i \mathbb{1}_{\{W \in \Delta_i\}}$ we get:

$$\begin{aligned} \mathbb{E}[Y] &= \mathbb{E}[\mathbb{E}[Y | I]] \\ &= \sum_{i=1}^m \mathbb{P}(I = i) \mathbb{E}[Y | I = i] \\ &= \sum_{i=1}^m \mathbb{P}(W \in \Delta_i) \mathbb{E}[Y | W \in \Delta_i] \\ &= \sum_{i=1}^m p_i E_i \end{aligned}$$

where $p_i = \mathbb{P}(W \in \Delta_i)$ and $E_i = \mathbb{E}[Y | W \in \Delta_i]$.

Taking a partition such that the partitions are equiprobable the formula simplifies to:

$$\mathbb{E}[Y] = \frac{1}{m} \sum_{i=1}^m E_i$$

This expression can be computed by computing each summand with a plain Monte Carlo simulation. W is called the *stratification variable* and m is called *strata*.

3 Simple plain Monte Carlo estimation

3.1 Using explicit formula for geometric Brownian motion

Our first estimation for θ shall be a standard Monte Carlo simulation where we assume that we can compute S_T by formula 1. To implement this problem in Matlab I did the following steps:

- i) Generate 10^4 realizations of W_T . Note that since W is a Brownian Motion we have $W_T \sim \mathcal{N}(0, T)$.
- ii) Compute the via Black Schools model related S_T realizations .
- iii) Monte Carlo
- iv) Scaling results with e^{-rT} so that we now have an estimator for θ and the related confidence interval.

One gets the following computations:

```
Theta = 9.8373
ConfI = [9.5600, 10.1147]
length = 0.5547
```

3.2 Using Euler scheme

Here I used the euler scheme to simulate the realizations of S_T instead of direct computations by formula 1. In a previous Project we analysed how one can simulate a geometric Brownian motion G by the euler scheme. To compute S_T we have to do a little scaling in the following way:

$$\begin{aligned} S_T &= S_0 \exp \left(\left(r - \frac{\sigma^2}{2} \right) T + \sigma W_T \right) \\ &= S_0 e^{rT} \exp \left(\sigma W_T - t \frac{\sigma^2}{2} \right) \\ &= S_0 e^{rT} G_T \end{aligned}$$

where G is a geometric Brownian motion. Having generated the samples of S_T we continue analogously like above.

One gets the following computations:

```
Theta = 9.8273
ConfI = [9.5481, 10.1066]
length = 0.5585
```

4 Analytical solution

In this section I will give an analytical solution for θ . In the section about the mathematical model I introduced formula 2 which allows us to compute θ via the cumulative distribution function standard normal distribution. Luckily there is a build-in function available in MATLAB[®].

So using the theoretical thoughts we can immediately compute by putting our values in the formula:

One gets the following computations:

```
Theta = 9.9251
```

It can be seen pretty easy that my estimations from exercise 1 are very close to the theoretical value. Both are slightly smaller but the exact value is always contained in the confidence interval. One should keep in mind, that the approximation with normal distribution is always slightly bigger than the real values so this effect is no contradiction. The best estimation of the two is the first one (uses the explicit formula for geometric Brownian motion). We can also see that the confidence interval is slightly smaller. However, an average length of the confidence interval of round about 0.55 shows us that the variance of the two methods is relatively high. Hopefully the next computations with variance reduction methods well result in shorter confidence intervals.

5 Control variates method

Since we can compute $\mathbb{E}[S_T] = S_0 e^{rT}$ very easily and $\theta = \mathbb{E}[f(S_T)]$ with a function $f : \Omega \rightarrow \Omega$ defined by $f(\cdot) = e^{-rT}(\cdot - K)_+$ it makes sense to use the control variates method for an approximation of θ . To implement this problem in Matlab I did the following steps:

- i) First, I generated 10^4 realisations of a geometric Brownian motion and scaled them to our asset model and also the positive Parts of their difference with K .
- ii) By the explanation in the theoretical part in section 2.2.1 we have to compute $c^* = -\frac{\text{Cov}[X, Z]}{\text{Var}[Z]}$. To do this I computed expectations and variances of the above vectors by plain Monte Carlo with a small integer. I decided for $p = 400$.
- iii) Now I computed by plain Monte Carlo $\mathbb{E}[W_{c^*}]$.

One gets the following computations:

```
Theta = 9.8929
ConfI = [9.3655, 10.4202]
length = 1.0547
```

Unfortunately the length of the confidence interval is huge. There must be a mistake in the computation of the variance, but I can't find it. Since this method shall reduce the variance of our Monte Carlo estimation, we should get a length which is significantly smaller than the values of about 0.5 in exercise 1.

6 Importance sampling method

To given an estimation of θ by the importance sampling method it is admitted that

$$\theta = \mathbb{E}[\exp(-\mu W_T - \frac{\mu^2}{2}T)f(W_T + \mu T, K, S_0, \sigma, r, T)]$$

where $f(W_T, K, S_0, \sigma, r, T) := e^{-rT}(S_0 \exp((r - \frac{\sigma^2}{2})T + \sigma W_T) - K)_+$.

Sharp looking shows that f clearly is the standard function that we need to compute the premium because it holds:

$$\theta = \mathbb{E}[f(W_T, K, S_0, \sigma, r, T)]$$

Since K, S_0, σ, r and T are fixed throughout the project, I will simply write $f(W_T)$ in the following. By assumption we have:

$$\theta = \mathbb{E}[f(W_T)] = \mathbb{E}[\exp(-\mu W_T - \frac{\mu^2}{2}T)f(W_T + \mu T)]$$

and if we consider the presentation to the fourth project we have:

$$\exp\left(-\mu W_T - \frac{\mu^2}{2}T\right) = \frac{\phi_{W_T}(W_T + \mu T)}{\psi(W_T + \mu T)}$$

where ϕ_{W_T} is the density of $\mathcal{N}(0, T)$. We will reach our aim, to find the optimal μ if we use the Theorem of Rubinstein in a smart way. Unfortunately I didn't find a smart way here. The only possibility that remains is to use the program to generate the results for different μ and compare them.

6.1 Estimation with the given value for μ

To implement the importance sampling I did the following steps:

- i) generation of 10^4 realizations of W_T
- ii) compute $f(W_T + \mu T)$ by the given formula for f and $\mu = 1.1$
- iii) plain Monte Carlo method for our expression

One gets the following computations for $\mu = 1.1$:

```
Theta = 9.9432  
ConfI = [9.8519, 10.0345]  
length = 0.1826
```

I am very impressed by the remarkable small confidence interval. This shows that I did a good job in variance reduction. To reach a comparable level of confidence with plain Monte Carlo simulations like in the first exercise one would need a number of realizations many times bigger than 10^4 .

6.2 Find the optimal μ

As I have explained above the strategy here should be to use Rubinstein Theorem. Unfortunately I didn't manage to get a value for μ her. I am very interested how this can be solved. Please notify jaschekt@math.hu-berlin.de if you find an explanation. Nevertheless, I really wanted a result her so I implemented a function that tests many different values for μ many times (I have to consider averages because of fluctuation). First, I did a rough case distinction with integer values for μ in the range of 0 to 10. The results always was 1 and so I refined my grid step by step. Finally, I considered possible values for μ in the range of 0.5 to 2 with steps of length 0.1 and 30 simulations for each value. The result is impressive.

```
mu optimal = 1.200  
length = 0.1821
```

So the optimal choice could be $\mu = 1 + \sigma$ or as well $\mu = 1 + \sigma T$.

7 Stratified sampling method

Finally I used the stratified sampling method with stratification variable $W_T \sim \mathcal{N}(0, T)$.

7.1 Estimation with 500 equiprobable strata

The difficulty in the method was to generate the equiproportionate intervals and the related values of W_T . I tackled this problem by using inversion sampling method. It was easy to divide the unit interval in equidistant partitions. Having done that one can use the inverse of the cdf of $\mathcal{N}(0, T)$ to get the values for W_T . The rest of this function is straight forward Monte Carlo simulation on each Interval.

One gets the following computations for $\mu = 1.1$:

Theta = 9.9351

ConfI = [9.9249, 9.9454]

length = 0.0205

Brilliant! The confidence Interval is extraordinary small. This is a great result for this task.

7.2 Standard derivation for different strata

Here I simply used the function from the previous section with different values for m_{st} and computed the standard derivation from the confidence intervals.

One gets the following computations

| | | | | | | | |
|-----|--------|--------|--------|--------|--------|--------|--------|
| mst | 20 | 50 | 100 | 200 | 500 | 1000 | 2000 |
| std | 0.4949 | 0.2976 | 0.3703 | 0.7783 | 0.6048 | 0.4418 | 0.5236 |

The best result for the standard derivation I got for $m_{st} = 50$.

8 Comparison of the manifold methods

First, I want to compare the methods with respect to the length of the confidence intervals. This criterion is the most important because it shows us, which method has the minimum variance. The control variates method I will not mention since there must be a mistake in the computation of the confidence interval. Without any further mathematical ideas I got with a plain Monte Carlo method an average length for the confidence interval of 0.5500. Importance sampling made it possible to reduce the length down to 0.1821. Stratified sampling made it possible to reach tiny confidence intervals with a length of just 0.0250 and thus is the most exact variance reduction methods for the computation of the premium of a European call option.

If I compare the methods to the difficulty of implementation I would like to point out, that importance sampling was very easy and fast to implement and gives us very good results. A simple plain Monte Carlo is also always a good idea if you want to get just a brief knowledge of what numbers you can expect in a broader sense.

During this project I learned a lot about variance reduction methods and understood why they are important. It was an first interesting experience in financial mathematics and the application of variance reduction methods. I am looking forward to a situation when I have to compute the premium of a European call option in future times and can use this program to do it with an incredible small variance.

Thank you for supervising, have a nice week.
T. Jaschek