

# MATH40082 (Computational Finance)

## Assignment No. 1: Monte Carlo Methods

Version 10134621

## 1 Background

### 1.1 Stock Options

Consider the equation for geometric Brownian motion, as used to model the path of an underlying asset paying proportional dividends at a continuous rate  $D_0$ :

$$dS = (\mu - D_0)Sdt + \sigma SdW, \quad (1)$$

where  $dW$  is the increment of a Wiener process (drawn from a Normal distribution with mean zero and standard deviation  $\sqrt{dt}$ ); we may then write that

$$dW = \phi\sqrt{dt}, \quad (2)$$

where  $\phi$  is a random variable drawn from a normalised Normal distribution.

Utilising (2) and risk neutrality, (1) can be integrated exactly over a timescale  $\delta t$  (NOT necessarily small) to yield (see also your lecture notes)

$$S(t + \delta t) = S(t) \exp \left( (r - D_0 - \frac{1}{2}\sigma^2)\delta t + \sigma\phi\sqrt{\delta t} \right). \quad (3)$$

Equation (3) then generates a random path. Since  $\delta t$  need not be small, in the case of European options, it is possible to generate a (random) value of  $S$  at expiry ( $t = T$ ) in just one step (i.e.  $\delta t = T$ ). From this value (say  $S(T)$ ), the payoff can then be easily calculated.

### European Options

Note that because the stock is paying dividends it makes the value of holding a share a little different since cash dividend payments are made to stock holders. We assume here that all contracts in the portfolio are options, so that **no** cash payments are received by the owner of the portfolio. Here we may price the options in the portfolio according to the following formula:-

- Assume that:

$$d_1 = \frac{\ln(S/X) + (r - D_0 + \sigma^2/2)(T - t)}{\sigma\sqrt{T - t}}, \quad d_2 = d_1 - \sigma\sqrt{T - t}.$$

- A Put Option  $P$  with terminal condition

$$P(S, T) = \max(X - S, 0)$$

has the analytic solution

$$P(S, t) = Xe^{-r(T-t)}N(-d_2) - Se^{-D_0(T-t)}N(-d_1)$$

- A Call Option  $C$  with terminal condition

$$C(S, T) = \max(S - X, 0)$$

has the analytic solution

$$C(S, t) = Se^{-D_0(T-t)}N(d_1) - Xe^{-r(T-t)}N(d_2),$$

- A Binary Put Option  $BP$  with terminal condition

$$BP(S, T) = \begin{cases} 1 & \text{if } S \leq X \\ 0 & \text{if } S > X \end{cases}$$

has the analytic solution

$$BP(S, t) = e^{-r(T-t)}N(-d_2)$$

- A Binary Call Option  $BC$  with terminal condition

$$BC(S, T) = \begin{cases} 0 & \text{if } S \leq X \\ 1 & \text{if } S > X \end{cases}$$

has the analytic solution

$$BC(S, t) = e^{-r(T-t)}N(d_2)$$

- If there is a payoff at maturity which is equal to the stock price, this is equivalent to the value of a call option with strike ( $X = 0$ ) so that

$$C(S, T; X = 0) = S$$

and

$$C(S, t; X = 0) = Se^{-D_0(T-t)} = e^{-r(T-t)}F_{t,T}$$

which is the discounted futures price  $F_{t,T}$ .

If this payoff of a portfolio is denoted as  $\Pi_i(t = T)$  (for the  $i$ th simulation), then the value of this payoff at  $t = 0$  is

$$\Pi_i(t = 0) = \Pi_i(t = T)e^{-rT} \quad (4)$$

If  $N$  simulations are performed, then (as described in the notes) we merely average out the  $\Pi_i(t = 0)$  to yield an approximation for the value of the portfolio, i.e.

$$\Pi = \frac{\sum_{i=1}^{i=N} \Pi_i(t = 0)}{N} \quad (5)$$

## 1.2 Path Dependent Options

Given a stochastic process that, as before, is governed by

$$dS = (\mu - D_0)Sdt + \sigma SdW.$$

Then the following options will depend on  $S(t_i)$  which are the share prices at  $K + 1$  equally spaced sampling times  $t_0, t_1, \dots, t_K$  with  $t_0 = 0$  and  $t_K = T$  (unlike part (a), the computation cannot proceed from  $t = 0$  to  $t = T$  in one step). Full details are given the the lecture notes - but the important point to note is that

$$S_{t_i}^n = S_{t_{i-1}}^n \exp[(r - D_0 - \frac{1}{2}\sigma^2)(t_i - t_{i-1}) + \sigma\sqrt{t_i - t_{i-1}}\phi_i]$$

to estimate the underlying asset values at each time, where each of the  $K$  increments  $dW_i$  involves drawing  $\phi_i$  from a Normal distribution.

## Asian Option

Assume that a discretely sampled Asian option has a payoff depending on the discretely sampled average given by

$$A = \frac{1}{K} \sum_{i=1}^K S(t_i).$$

Then we can write

$$V(S, A, t = T) = f(S, A),$$

where  $f$  is the payoff function depending the type of option.

There are different classes of Asian option, resulting in different payoff conditions. In this coursework we look at simple European style call or put options. A fixed strike call option will have the payoff

$$f(S, A) = \max(A - X, 0)$$

where  $X$  is the strike price and a floating strike call option would be

$$f(S, A) = \max(S - A, 0).$$

where  $A$  is sometimes called the average strike price.

A fixed strike put option will have the payoff

$$f(S, A) = \max(X - A, 0)$$

where  $X$  is the strike price and a floating strike put option would be

$$f(S, A) = \max(A - S, 0).$$

where  $A$  is the strike price.

## Lookback Option

The discretely sampled Lookback option has a payoff depending on the discretely sampled maximum or minimum given by

$$A = \max_i S(t_i),$$

or

$$A = \min_i S(t_i).$$

Then we can write

$$V(S, A, t = T) = f(S, A),$$

where  $f$  is the payoff function depending the type of option.

There are different classes of Lookback option, resulting in different payoff conditions. In this coursework we look at simple European style call or put options. We can either have a floating strike  $S$  or a fixed strike  $X$ . For example a floating strike Lookback call option would give

$$f(S, A) = \max(S - A, 0)$$

where  $A$  must be the minimum, and a floating strike Lookback put option would be

$$f(S, A) = \max(A - S, 0).$$

where  $A$  must be the maximum.

A fixed strike call option will have the payoff

$$f(S, A) = \max(A - X, 0)$$

where  $X$  is the strike price and  $A$  must be the maximum. and a fixed strike put option will have the payoff

$$f(S, A) = \max(X - A, 0)$$

where  $X$  is the strike price and  $A$  must be the minimum.

## Barrier Options

The discretely sampled knock-out barrier option will be knocked out (and return a value of zero) if the barrier asset price  $B$  is crossed before the maturity date.

The option will be an “up” option if the knock out condition is on  $S > B$ , or a “down” option if the condition is on  $S < B$ . So for example a up-and-out knockout barrier call option has the conditions

$$V(S, T) = \begin{cases} S - X & \text{if } S > X \\ 0 & \text{if } S < X \\ 0 & \text{if } S(t_i) > B \text{ for any } i = 1, 2, \dots, K \end{cases}$$

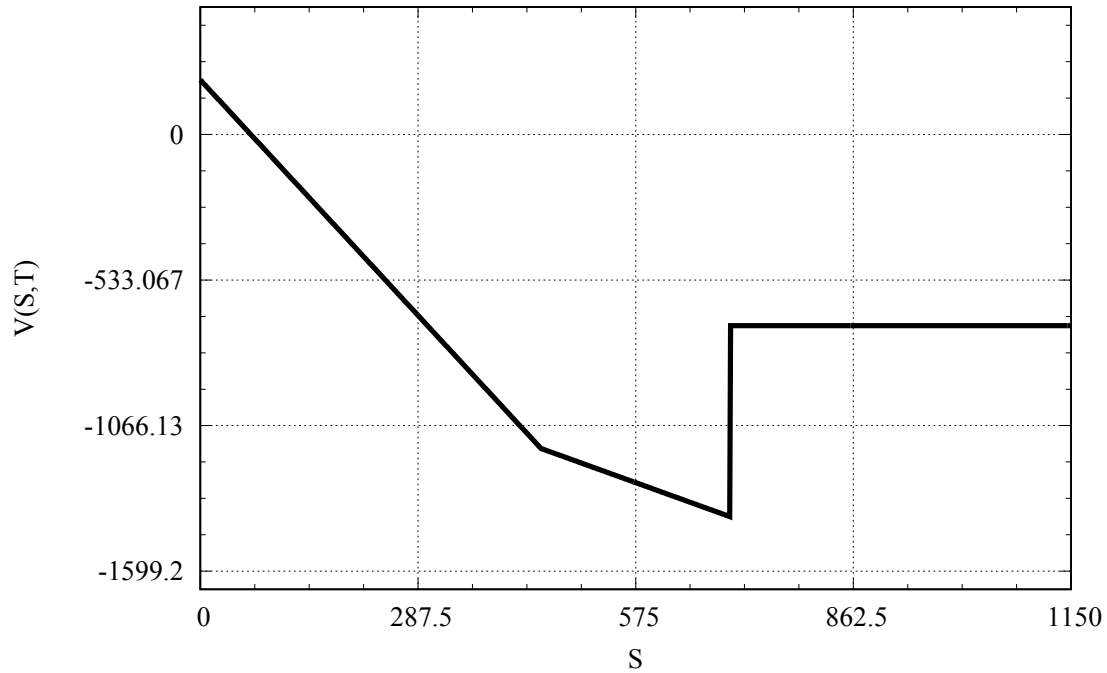
and a down-and-out knockout barrier put option will be

$$V(S, T) = \begin{cases} X - S & \text{if } S < X \\ 0 & \text{if } S > X \\ 0 & \text{if } S(t_i) < B \text{ for any } i = 1, 2, \dots, K \end{cases}$$

## 2 Tasks

### 2.1 Stock Options

You must value the portfolio comprising of long two put options with strike price  $X_1$ , long a call option with strike price  $X_2$ , short  $X_2$  binary put options with strike price  $X_2$  and unit payoff and short a call option with strike price equal to zero with the parameters  $T = 0.5$ ,  $\sigma = 0.25$ ,  $r = 0.03$ ,  $D_0 = 0.01$ ,  $X_1 = 450$  and  $X_2 = 700$ . The payoff of the portfolio at time  $T$  is shown below



- Write a program to calculate the value of the portfolio using the parameters given at  $t = 0$  using Monte Carlo simulation. You need only include the code in the appendix of your report. (Coding 3 marks)
- Plot out two figures for the value of the portfolio with  $t = 0$  at  $S_0 = X_1$  and  $S_0 = X_2$ , with increasing  $N$  ( $N = 1000, 2000, \dots, 50000$ , or more!) and compare the values you obtain with the exact values from the analytical formula. (Understanding 4 marks)

- Try to obtain a confidence interval for the option value at  $S_0 = X_1$ . Show how this interval changes as  $N$  is increased. Explain your results. (Understanding 4 marks)
- Obtain a confidence interval when using antithetic variables, and **compare** with the results using the basic method above. Explore how efficient using antithetic variables can be using different values of  $N$ . (Understanding 4 marks)
- Try an **appropriate** extension to the Monte-Carlo method and analyse its benefit. Comment on the efficiency of the method. (Originality/Initiative 5 marks)

## 2.2 Path Dependent Options

For this task you are required to value a discrete floating-strike Asian call option with the following parameters. The option matures at  $T = 1.25$ , the interest rate is  $r = 0.03$ , the dividend rate  $D_0 = 0.04$  and volatility is  $\sigma = 0.37$ . The stock price, currently at  $S_0 = 900$ , will be observed on  $K = 35$  plus one equally spaced dates throughout the lifetime of the option, where  $t_0 = 0$  and  $t_K = T$ . The floating strike price is the average stock price denoted by  $A$ . You should use Monte-Carlo simulation to value this option.

- Code up the path dependent option. (Coding 2 marks)
- Using the given parameters at  $t = 0$ , produce at **most** 4 plots or tables to investigate the value of the path dependent option with different values of  $N$  and  $K$ . Try to spot and explain trends. (Understanding 8 marks)
- You are tasked with providing the most accurate value possible of **discrete** version of the option using the given parameters ( $K$  as stated above). Assume that you are only given 10 seconds of computation time to return a value of the option. State the most accurate value you can get in that time limit, how you verified it and any techniques used. (Originality/Initiative 5 marks)

## 3 Instructions

This assignment will account for 40% of your final mark in this module. The total number of marks in this assessment are 40, and they will be awarded as follows:

(i) **5 for working codes;**

Grade	Description
0-50%	Little or no attempt, codes not working
50%-70%	One or two bugs in the code are affecting the results
70%-100%	Results in 2.1 and 2.2 from the codes appear correct

(ii) **5 for the presentation of your written report;**

Grade	Description
0-50%	Poorly presented work. Significant amount of text unreferenced. Graphs and tables poorly labelled making it difficult to interpret them.
50%-70%	Good presentation. Text is readable. Graphs are ok, maybe missing labels and not always referenced correctly. Report is overly long and unnecessarily repeats the same (or similar) results.
70%-100%	Excellent presentation, well written and well referenced. Graphs are clear, tables used when appropriate. Report keeps within the page limit.

(iii) **20 for the understanding of the problems involved;**

Grade	Description
0-50%	Results are poorly presented or they are without supporting text. The methods are described but are not shown to be implemented through results. The student is unable to demonstrate they can correctly interpret results.
50%-70%	Demonstrates a good understanding of the standard methods. Is able to generate standard results and discuss them. Results are well presented.
70%-100%	Student is able to correctly interpret standard results and evaluate the efficiency of the standard methods.

(iv) **10 for originality/initiative.**

Grade	Description
0-50%	Little or no attempt at implementing any of the new methods. Those that have been implemented have poorly presented results or the student is unable to demonstrate they can correctly interpret results.
50%-70%	Demonstrates a good understanding of new or alternative methods. Is able to implement new methods, present results and discuss them. Results are well presented.
70%-100%	Has implemented difficult algorithms not detailed in the course. Presentation of the results is excellent. Student is able to correctly interpret results and compare methods in a coherent way.

Please see individual bullet points in the task section for a break down of the marks.

Reports should be prepared electronically using either MS Word, LaTeX, or similar, and must be submitted **without your name, but with your university ID number** online through the TurnItIn system. Please include the program files used to generate results for the report in an appendix as plain text. Your report should be written in continuous prose in the form of a technical report and should be approximately 8 - 10 pages long (excluding appendices). Any programming language may be used. The deadline for this assignment is 11am on **Monday 12th APRIL**.

**THIS DEADLINE MUST BE STRICTLY ADHERED TO – Reports handed in AFTER 11am MONDAY 12th APRIL will be docked 4 marks plus an additional 4 marks each day thereafter until a mark of zero is reached. Reports handed in after 5pm Friday 23rd April will be awarded a mark of zero and will not be marked.**

In order that your report conforms to the standards for a technical report, you should use the following structure:

- give a brief introduction stating the problem you are solving and the parameters you are using (from the model or method),
- present your results in the form of figures and tables, using the order of items in the bullet points as a guide as to the order of your document
- absolutely **NO** screenshots of running code need to be included,
- do not include overly long tables – a table should **never** cross over a page,
- present the results for any methods you have implemented, there is no credit for a discussion of a method that has not been shown to be implemented by you (through results) for your problem

- refer to and discuss each of your results in the text, part of the marks available in each bullet point are for interpreting the results
- try to keep to the page limit, removing any unnecessary results from the main text
- number and caption your figures and tables and refer to them by their number (not their position in the text),
- number any equations to which you refer,
- use consistent internal (and external) referencing.