Module 3 Assignment
Asian Option Pricing
Cian O'Duffy

1 INTRODUCTION

This report details the valuation a number of asian options with a range of methods. Only continuous sampling of the stock price is considered and the following parameter values are used throughout:

Today's stock price S_0 = 100

Strike K = 100

Time to expiry $(T - t)$ = 1 year

Volatility σ = 20%

Constant risk -free interest rate r = 5%

Time varying parameters of r and σ are considered beyond the scope of this report. The coding was carried out in python. The data has been formatted into tables in excel intermediate calculations along with some intermediate calculations. Both an excel spreadsheet and python project have been submitted with this report.

2 SIMULATING THE UNDERLYING

The first step in the pricing of options by Monte Carlo simulations is the modelling of the underlying. In this assignment, we have been advised to use the simplest method for approximate simulation of stochastic differential equations, the Euler-Maruyama (EM) scheme. Consider the following equation:

$$dX_t = a(X_t, t)dt + b(X_t, t)dW_t \tag{1}$$

The EM scheme can solve this generally, however as the scope specifies constant values for *r*, and σ, these are considered constant and so the equation to be solved becomes the basic stochastic differential equation:

$$\frac{dS_t}{S_t} = rdt + \sigma dW_t, \tag{2}$$

which, when discretized gives the following EM scheme:

$$S_{t+1} = S_t(1 + r\delta t + \sigma\phi\sqrt{\delta t}) \quad , \tag{3}$$

where ϕ is a standard random normal variable, and δt is a small change in time. Note, while we are assuming that the parameters are constant, they do not need to be for the EM scheme to work, so the code I have written can easily be extended to include time-dependency and randomness.

As *r* and σ are constant there is also a closed form solution to (2):

$$S_{t+1} = S_t e^{\left(r-\frac{1}{2}\sigma^2\right)\delta t+ \sigma\phi\sqrt{\delta t}}$$

where:

$$e^{\left(r-\frac{1}{2}\sigma^2\right)\delta t+ \sigma\phi\sqrt{\delta t}} \sim 1 + r\delta t + \sigma\phi\sqrt{\delta t} + \frac{1}{2}\sigma^2(\phi^2 - 1)\delta\mathrm{t} + \cdots$$

The final term in this expression is called the Milstein correction. Consequently, I have simulated the underlying using both the EM scheme and the closed form solution to analyse the loss in accuracy for the EM scheme.

The code in the attached python project gives the following charts when viewed over a 10-year horizon:

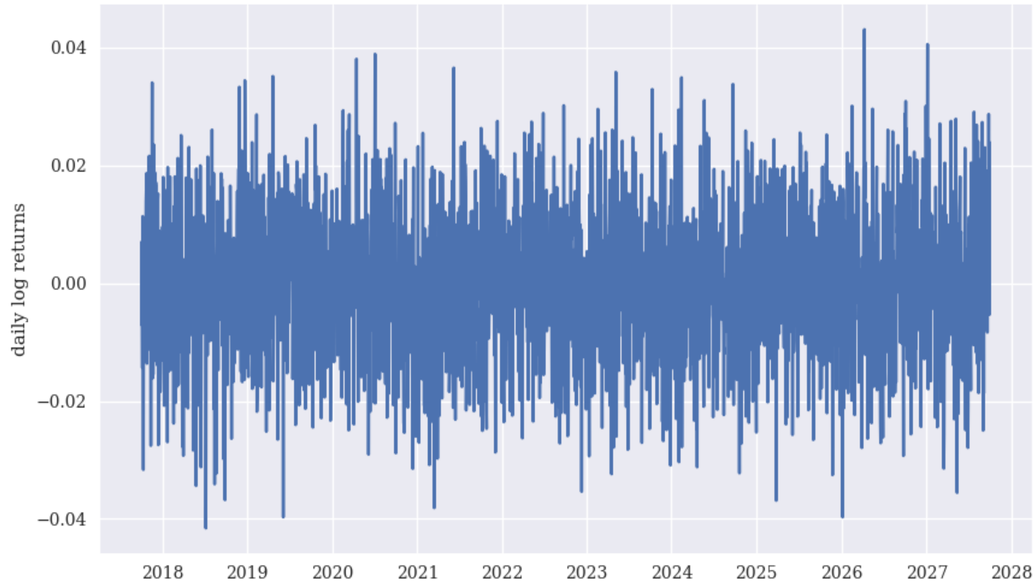

Figure 1: Daily log returns from sample path for EM Scheme

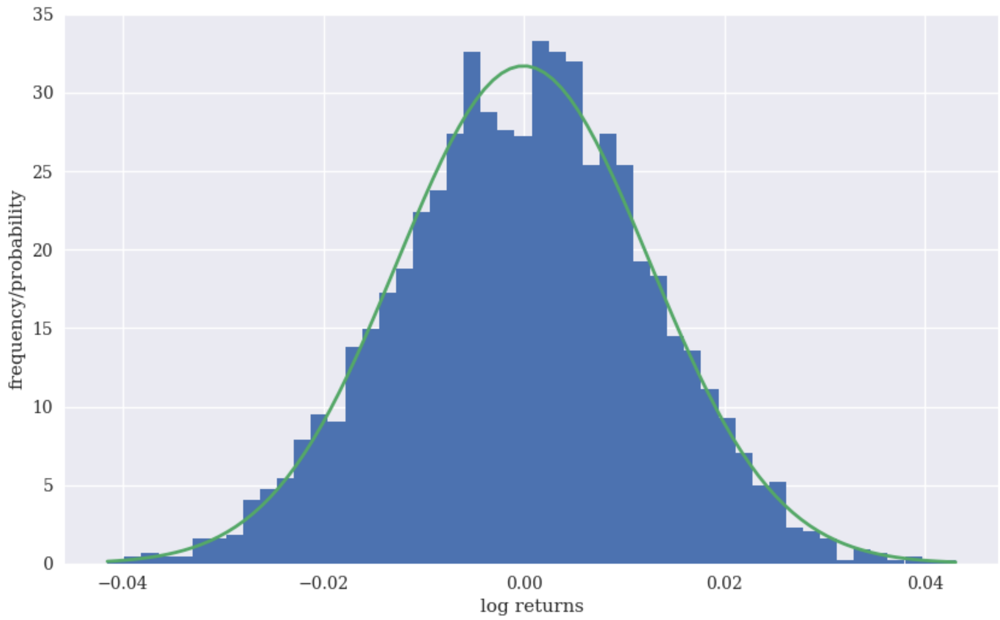

Figure 2: Histogram for log returns of sample path EM Scheme with normal pdf superimposed

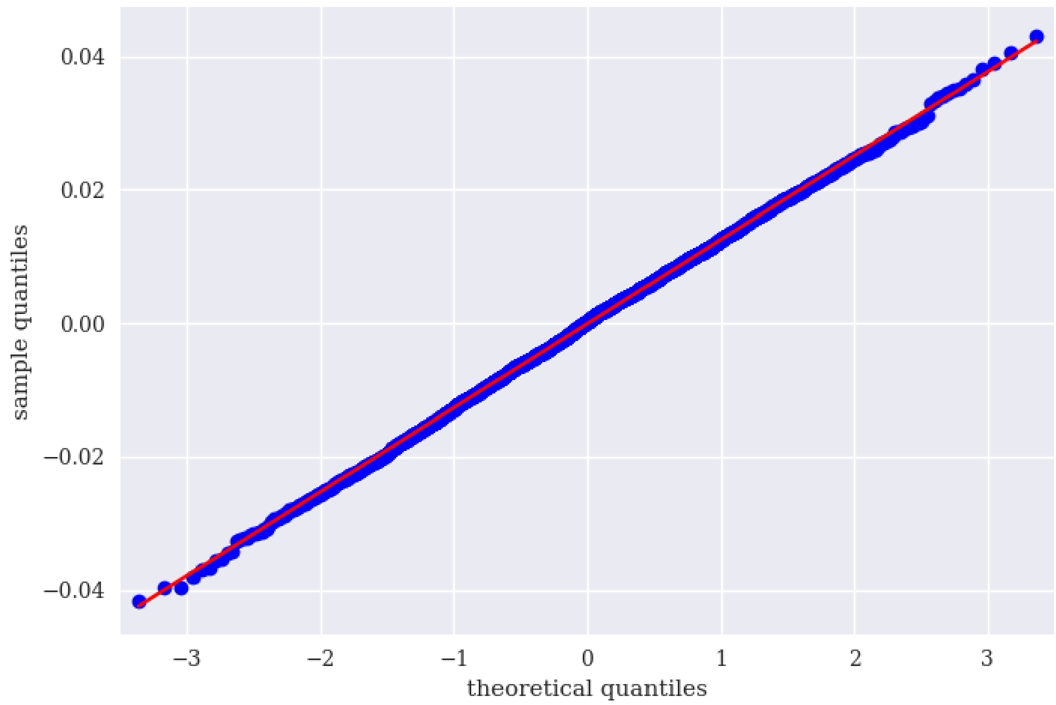

Figure 3: Q-Q Plot of log returns of EM Scheme sample path

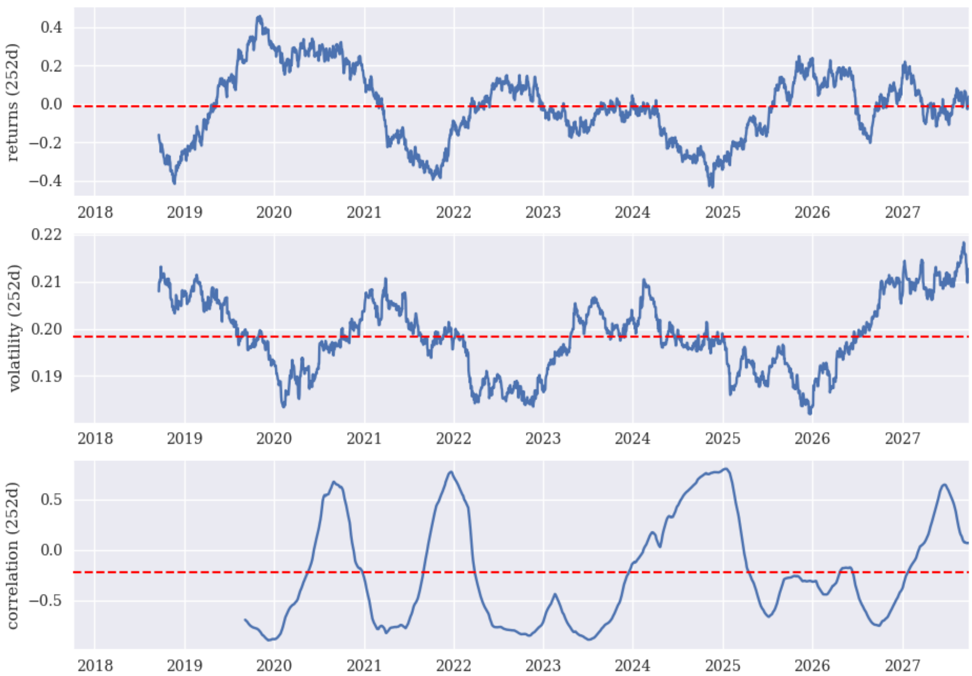

Figure 4: Annualised realized returns, annualized realized volatility an correlation between the two of sample path EM scheme.

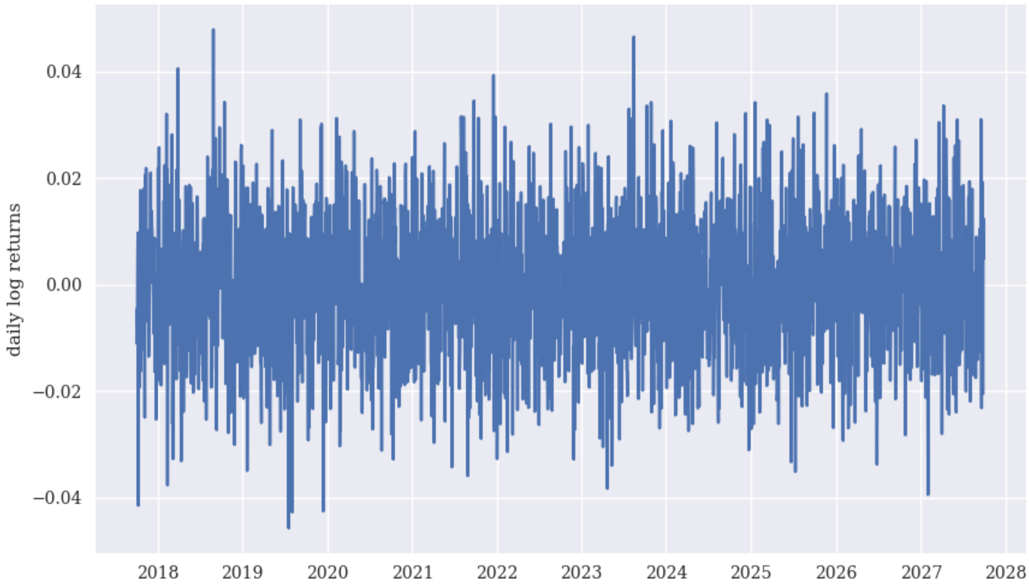

Figure 5: Daily log returns of sample path of closed form solution

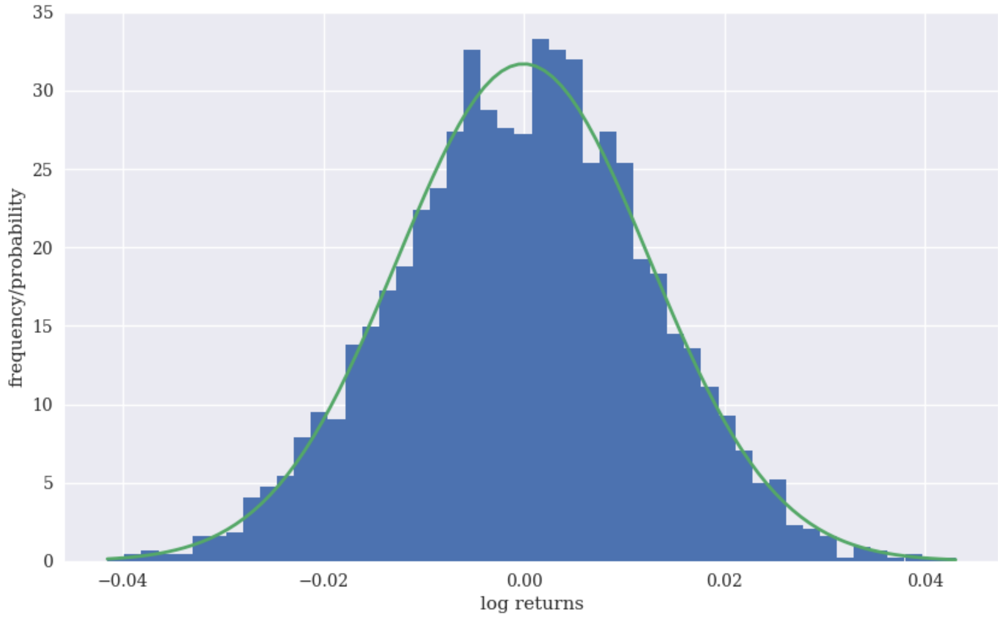

Figure 6: Histogram for log returns of sample path of closed form solution with normal pdf superimposed

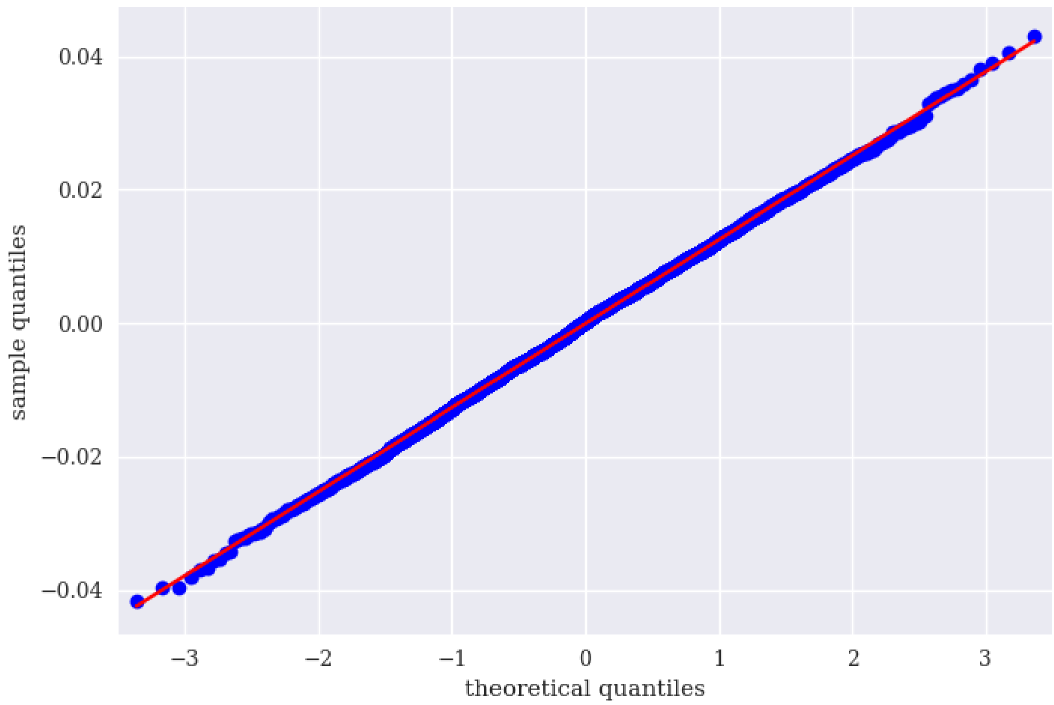

Figure 7: Q-Q Plot of log returns of closed form solution sample path

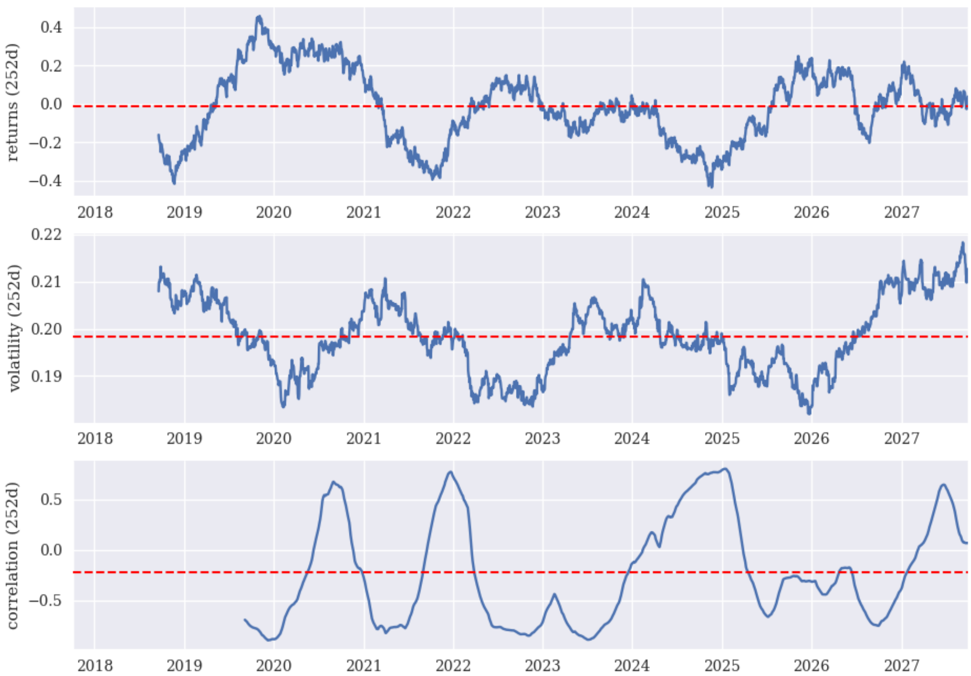

Figure 8: Annualised realized returns, annualized realized volatility an correlation between the two of sample path closed form solution.

Note that while the returns have been plotted over a 10-year horizon in the charts above, only a year horizon was used for option pricing as stated in the scope.

On the face of it there doesn’t seem much difference between the EM scheme and the exact solution. Both give normally distributed returns as expected from the stochastic differential equation (2).

3 SAMPLING FUNCTIONS

The payoff of Asian options is dependent on the average of the underlying spot price over the lifetime of the option. In this assignment, I have assumed continuous averaging (averaging over every time step). Additionally, I consider both arithmetic averaging and geometric averaging. There are two options for calculating the geometric average:

$$A = \left(\prod_{i=1}^{N} S(t_i)\right)^{\frac{1}{N}} \tag{4}$$

and

$$A = \exp\left(\frac{1}{N}\sum_{i=1}^{N} logS(t_i)\right) \tag{5}$$

Practically (5) is the only suitable method as when using (4), when the number of time steps is large enough (~150) the product become larger than the maximum value of a floating point number and so the result computers return is infinity. Therefore, I have used (5) when calculating the geometric average. When calculating the rolling geometric average, I have used the following schemes:

$$A_{t+1} = ((A_t)^{t+1}.S_{t+1})^{\frac{1}{t+2}} \tag{6}$$

$$A_{t+1} = \exp\left(\frac{1}{t+2}(logS_{t+1} + (t+1)logA_t)\right) \tag{7}$$

and the standard arithmetic equation given in (Ahmed, 2017).

Figure 9 displays both the rolling geometric and arithmetic averages for a sample path. The inequality of arithmetic and geometric means states that the arithmetic mean of a list of non-negative real numbers is greater than or equal to the geometric mean of the same list, and that the two means are equal if and only if every number in the list is the same. For the simplest non-trivial case, by considering the real numbers x and y:

$$(x-y)^2 \geq 0$$

you can show that:

$$\frac{x+y}{2} \geq \sqrt{xy}.$$

Consequently, the arithmetic mean will be higher for every simulated payoff than the respective geometric mean.

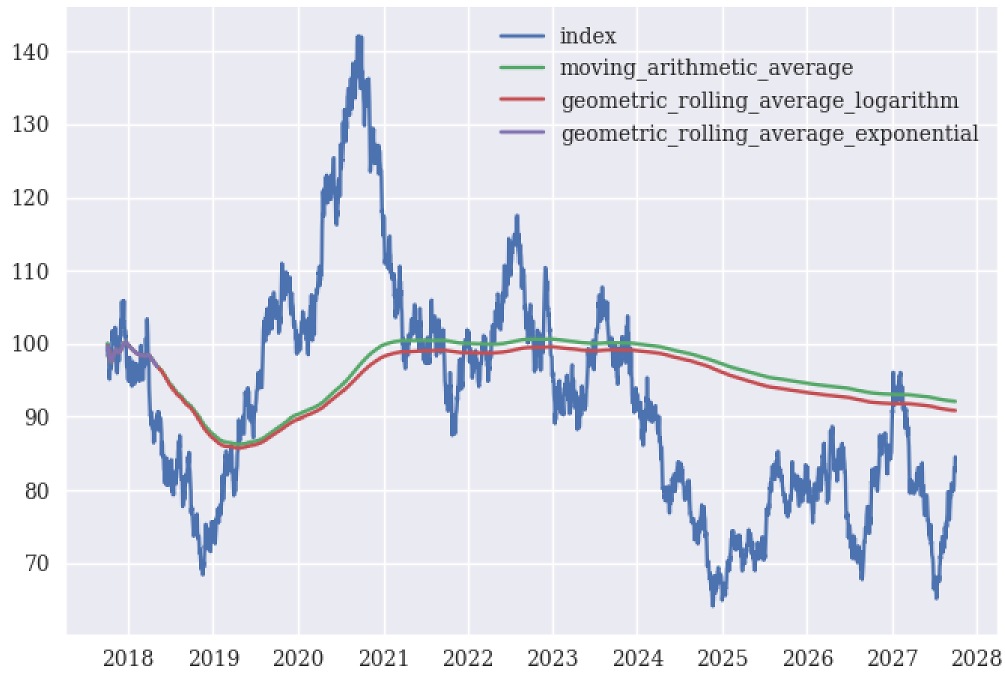

Figure 9: Rolling geometric and arithmetic averages for a sample path.

An additional consideration is whether the strike is floating (averaged) or fixed. Below is summarised the different payoffs considered in the assignment. Mean can refer to either geometric or arithmetic.

$$fixed\ strike\ call\ payoff = \max(mean - K, 0),$$

$$fixed\ strike\ put\ payoff = \max(K - mean, 0),$$

$$floating\ strike\ call\ payoff = \max(S_T - mean, 0),$$

$$floating\ strike\ put\ payoff = \max(mean - S_T, 0).$$

4 MONTE CARLO SIMULATION

Stock prices paths were generated for both the EM scheme and using the exact solution as illustrated in figure 10. The same random seed was used for the EM scheme and the exact solution so randomness can be eliminated when comparing the two methods. The distribution of the stock price in the final time step was used to determine the payoff distribution and the option price according to the fundamental asset pricing formula (FAPF).

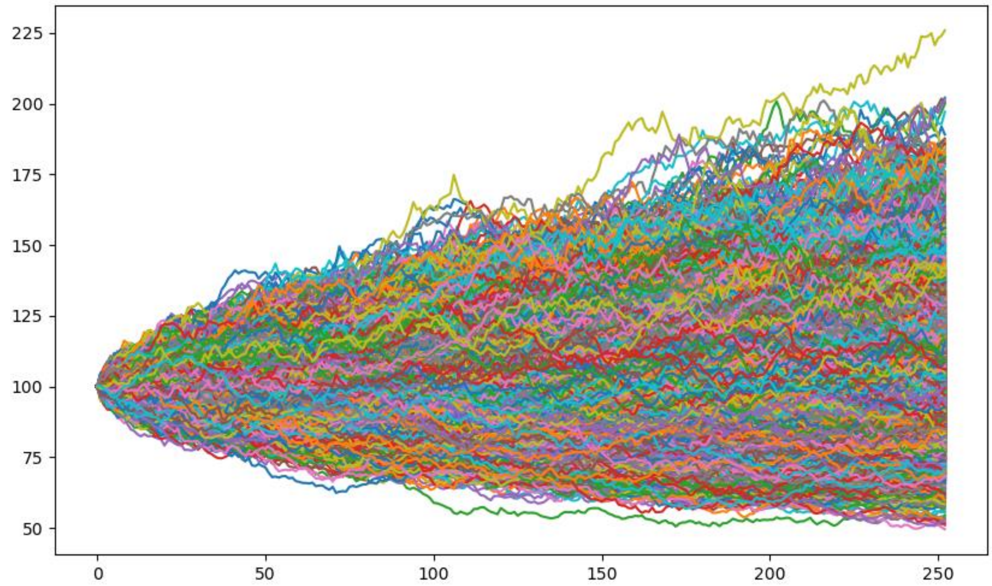

Figure 10: Stock price simulations against time-step

The sensitivity to number of simulations and timestep was considered by varying both of these parameters. The results are summarized in figure 11 to figure 15. The standard error of the mean of the payoff is:

$$SE_{\bar{x}} = \frac{SD}{\sqrt{N}},$$

where SD is the sample standard deviation and N is the number of simulations. Therefore, due to the FAPF the sample error of the option price is:

$$SE_{Price} = e^{-rT}\,\frac{SD_{Payoff}}{\sqrt{N}}.$$

These error bars are displayed in figures 11 and 12.

To validate the standard error calculation, six runs were compared for each time step to show that the standard error was equal to the standard deviation prices for each run. These results are summarized for an arithmetic Asian call with fixed strike in table 1, however the rest are the results are available in the attached excel spreadsheet.

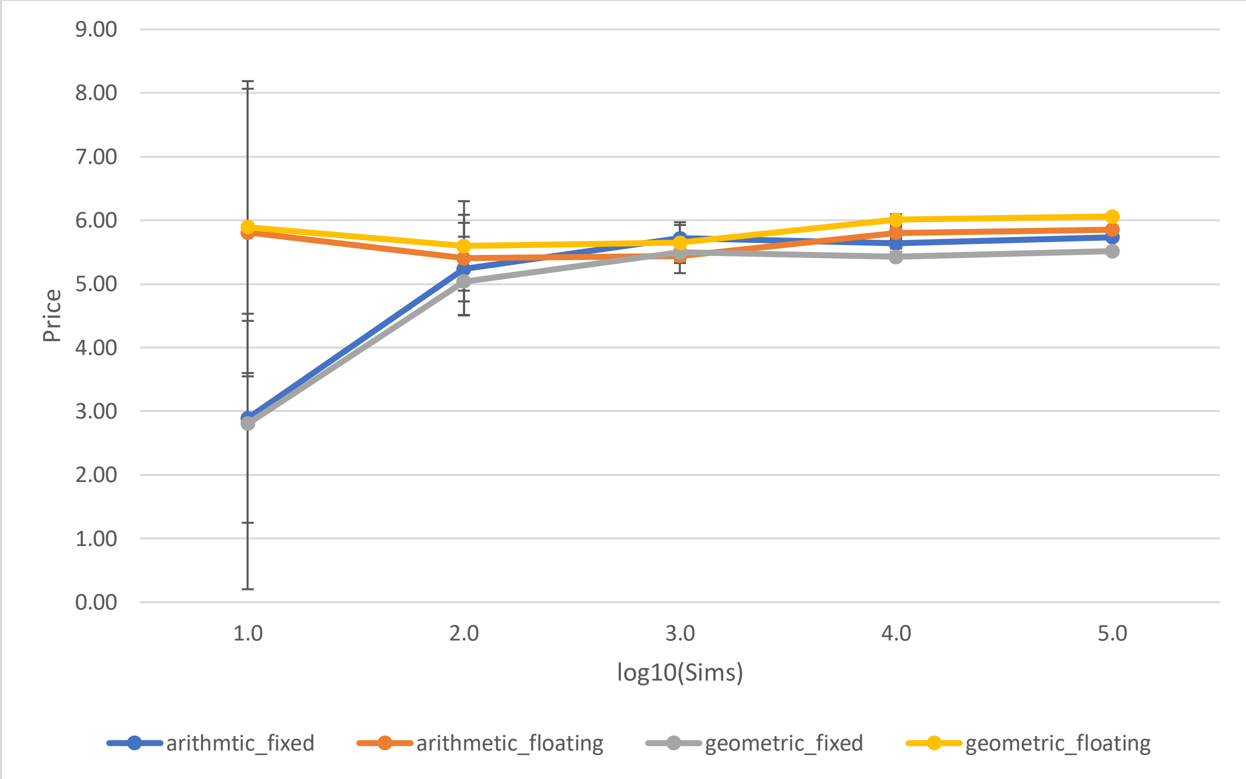

Figure 11: Asian Call Values for varying numbers of simulations. Error bars price standard error.

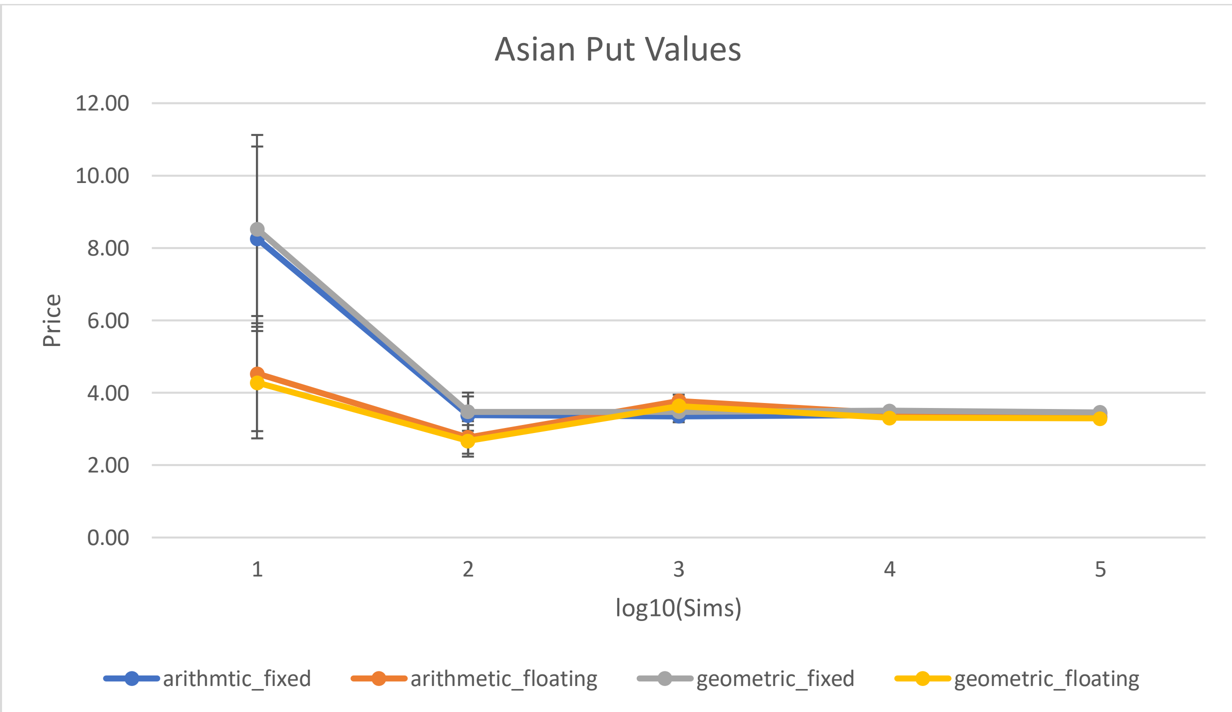

Figure 12: Asian Put Values for varying number of simulations. Error bars price standard error

Table 1: Arithmetic Fixed Strike Asian Call Prices for multiple runs

Sims	Price						Std of Price of all runs	Standard Error of price	Difference
	Run 1	Run 2	Run 3	Run 4	Run 5	Run 6			
10	2.89	5.97	3.59	6.16	4.71	5.00	1.29	1.64	-0.35
100	5.23	5.25	5.76	5.85	5.94	6.00	0.34	0.73	-0.38
1,000	5.71	5.40	5.75	5.69	5.28	6.16	0.31	0.25	0.05
10,000	5.63	5.75	5.68	5.75	5.65	5.85	0.08	0.08	0.00
100,000	5.73	5.81	5.70	5.81	5.78	5.78	0.04	0.03	0.02

The statistics do agree when considering a small number of runs are compared (six). Additionally, the variation in price between each run decreases with increasing number of simulations as you would expect.

Varying the number of timesteps used did not have a great impact on the option values although it appeared to have a greater impact on the put values than the call values.

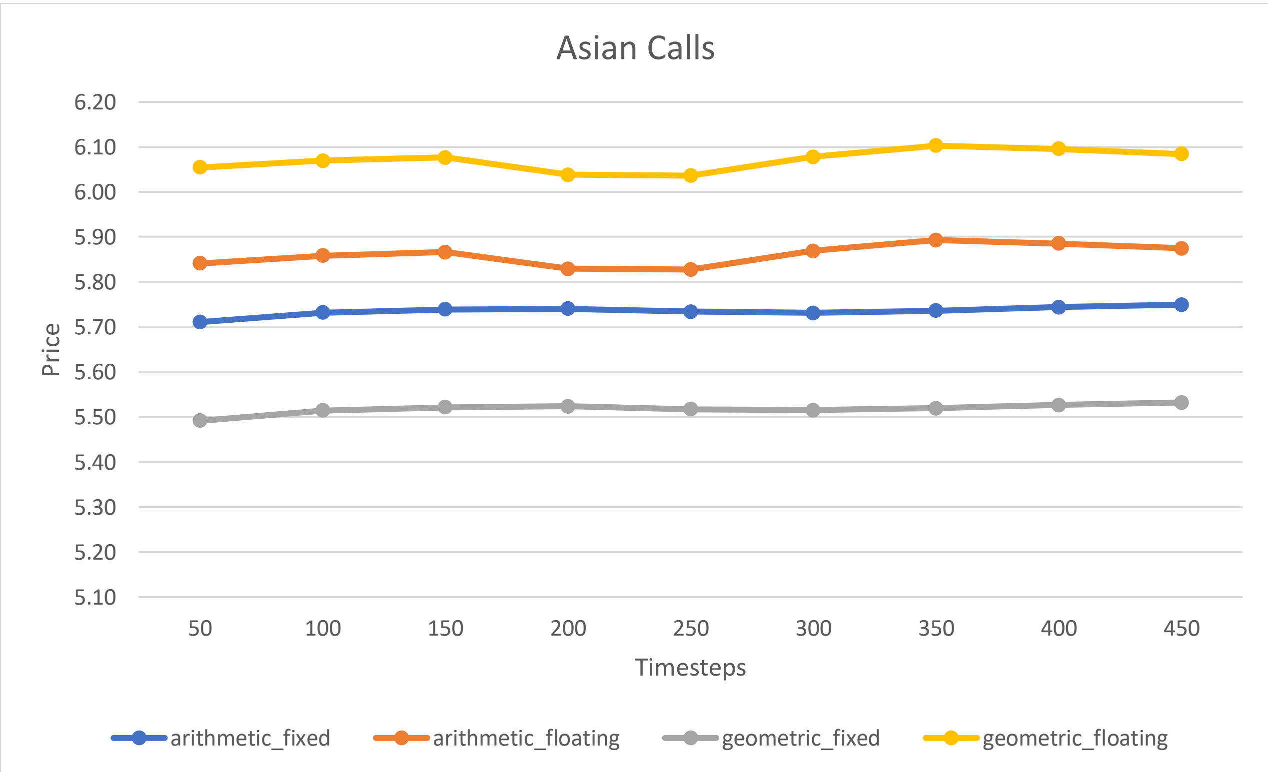

Figure 13: Asian call values for varying timestep. Number of simulations = 100,000

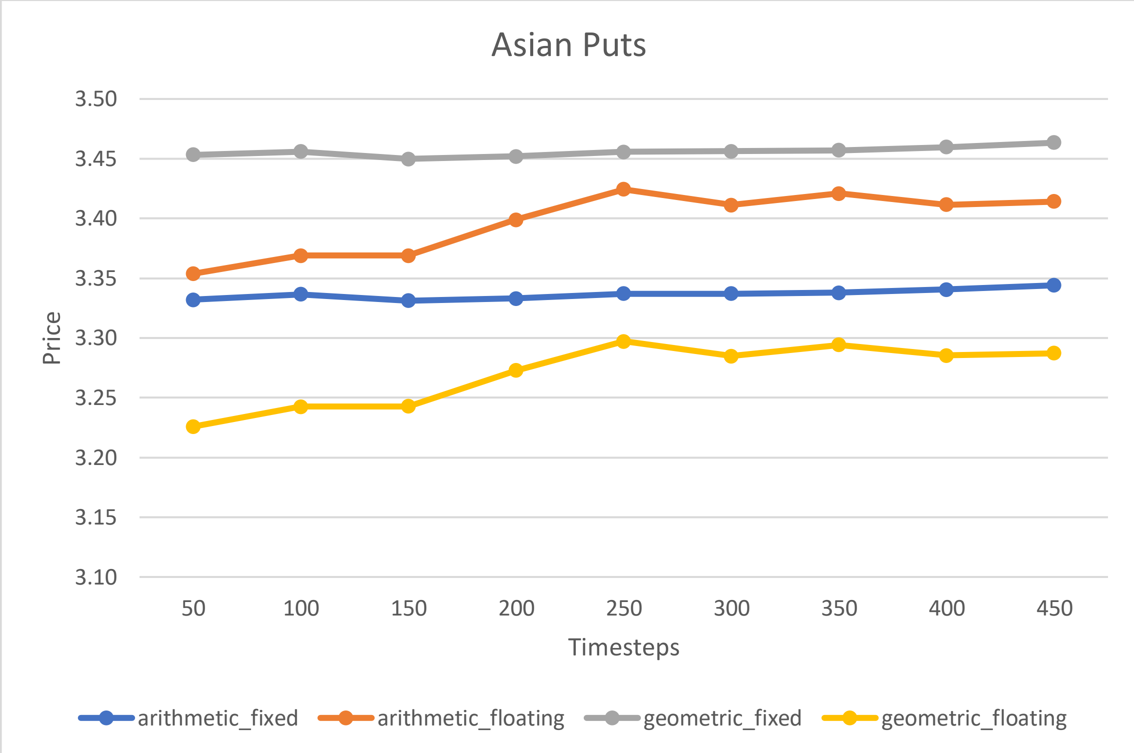

Figure 14: Asian put values for varying timestep. Number of simulations = 100,000

For the final analysis it was decided that 100,000 simulations should be used and 260 timesteps. The choice of simulations was decided as this is a number that gives a highly accurate results without being too computationally onerous. The choice of time step was chosen as the number of work days in the time period 31/9/2017 – 31/9/2018, and so each step could be considered a day which aids understanding.

Table 2: Asian Call Option prices comparing the Euler-Marayuma (EM) and closed form methods for simulating the underlying

	EM	Exact	Difference
arithmetic fixed strike	5.7325	5.7327	-0.0002
arithmetic floating strike	5.8519	5.8518	0.0002
geometric fixed strike	5.5160	5.5160	0.0000
geometric floating strike	6.0604	6.0605	0.0000

Table 3: Asian Put Option prices comparing the Euler-Marayuma (EM) and closed form (exact) methods for simulating the underlying

	EM	Exact	Difference
arithmetic fixed strike	3.3371	3.3369	0.0001
arithmetic floating strike	3.4183	3.4182	0.0001
geometric fixed strike	3.4559	3.4557	0.0003
geometric floating strike	3.2915	3.2914	0.0001

When comparing the EM and closed form methods it is clear that there is not a large difference between the two when compared to the standard error of the price. In the case of an arithmetic fixed strike call the standard error of the price is 130 times larger than the difference in price between the two methods and therefore using the EM scheme without the Milstein correction will give as good an answer in this case as the with the Milstein correction. This is good news for using this the EM scheme to simulate more complicated dynamics (stochastic volatility etc).

When evaluating a pricing model it is good to price vanilla options where a closed form solution exists as summarised in table 4. These results demonstrate that while the model is not incorrect, there is error.

Table 4: Vanilla Options priced with the Monte Carlo Code

	Monte Carlo	Closed Form	Difference
Call	10.41	10.45	-0.39%
Put	5.58	5.57	0.13%

5 CONTROL VARIATES

As it appears that the largest error in the price is the error associated with determining the population mean of the payoff control variates were used to reduce the variance of the payoff sample in a computationally efficient manner. This has the advantage of reducing the standard error of the price without the time and expense associated with increasing the number of simulations.

The use of control variates is appropriate in this case as for both asian calls and asian puts we know the exact solution of a highly correlated random variable in the form of vanilla calls and puts respectively. We can therefore use this knowledge to reduce the variance of the Asian option payoff distribution.

The following discussion is taken from (Glasserman, 2003). Consider the Asian option payoff Y_i and its associated vanilla option payoff X_i (calls and puts respectively), where *i* is the simulation number (the same random numbers are used to calculate both Y_i and X_i). The pairs (X_i, Y_i), $i = 1,\ldots, n$, are *i.i.d.* and the expectation $E[X]$ of X_i is known (the closed form vanilla solution*e^{rT}). For any fixed *b* we can calculate:

$$Y_i(b) = Y_i - b(X_i - E[X])$$

from the *i*th replication and then compute the sample mean

$$\bar{Y}(b) = \bar{Y} - b(\bar{X} - E[X]) = \frac{1}{n}\sum_{i=1}^{n}(Y_i - b(X_i - E[X])). \tag{8}$$

This is a control variate estimator; the observed error $\bar{X} - E[X]$ serves as a control in estimating $E[Y]$.

As an estimator of $E[Y]$, the control variate estimator (8) is unbiased because

$$E[\bar{Y}(b)] = E[\bar{Y} - b(\bar{X} - E[X])] = E[\bar{Y}] = E[Y].$$

The variance of $Y_i(b)$ is

$$Var[Y_i(b)] = Var[Y_i - b(X_i - E[X])] = \sigma_Y^2 - 2b\sigma_X\sigma_Y\rho_{XY} + b^2\sigma_X^2 \equiv \sigma^2(b), \tag{9}$$

where $\sigma_X^2 = Var[X]$, $\sigma_Y^2 = Var[Y]$, and ρ_{XY} is the correlation between *X* and *Y*. The control variate estimator $\bar{Y}(b)$ has variance $\sigma^2(b)/n$ and the ordinary sample mean $\bar{Y}$ (which corresponds to *b*=0) has variance σ_Y^2/n. Hence the control variate estimator has smaller variance than the standard estimator if $b^2\sigma_X < 2b\sigma_Y\rho_{XY}$.

The optimal coefficient b^* minimizes the variance and is given by

$$b^* = \frac{\sigma_Y}{\sigma_X}\rho_{XY}, \tag{10}$$

which when substituted into (9) gives the following ratio:

$$\frac{Var[\bar{Y} - b^*(\bar{X} - E[X])]}{Var[\bar{Y}]} = 1 - \rho_{XY}^2. \tag{11}$$

The first step to implementing this method is to calculate *b** from the correlations of the asian and vanilla option payoffs. These are summarised in tables 5 and 6.

Table 5: Correlation between asian and vanilla call payoffs and the associated b* values

	Correlation	b*
geometric fixed strike	0.83	0.43
arithmetic fixed strike	0.84	0.45
geometric floating strike	0.88	0.53
arithmetic floating strike	0.88	0.51

Table 6: Correlation between asian and vanilla put payoffs and the associated b* values

	Correlation	b*
geometric fixed strike	0.82	0.51
arithmetic fixed strike	0.81	0.49
arithmetic floating strike	0.76	0.46
geometric floating strike	0.76	0.44

Clearly asian call and put and payoffs are highly correlated to vanilla call and put payoffs which as per (11) results in a large variance reduction.

(Stavri, 2014) provides closed form solutions for geometric asian options to compare the results of the monte carlo simulations against. These are summarised below where the values for b* in the tables above have been used to calculate the results for the control variate method.

Table 7: Asian call option prices

	Standard	Control Variate	Closed Form solution	Error (Control Variate)
arithmetic fixed strike	5.733	5.751	N/A	N/A
arithmetic floating strike	5.852	5.873	N/A	N/A
geometric fixed strike	5.516	5.534	5.468	0.066
geometric floating strike	6.060	6.082	6.072	0.010

Table 8: Asian put option prices

	Standard	Control Variate	Closed Form solution	Error (Control Variate)
arithmetic fixed strike	3.337	3.334	N/A	N/A
arithmetic floating strike	3.418	3.415	N/A	N/A
geometric fixed strike	3.456	3.452	3.463	-0.011
geometric floating strike	3.291	3.288	3.279	0.010

Table 9: Standard Error of asian call option prices

	Error (Standard)	Error (Control Variate)	Difference
arithmetic fixed strike	0.025	0.014	-0.011
arithmetic floating strike	0.027	0.013	-0.014
geometric fixed strike	0.024	0.013	-0.011
geometric floating strike	0.028	0.013	-0.015

Table 10: Standard error of asian put option prices

	Error (Standard)	Error (Control Variate)	Difference
arithmetic fixed strike	0.017	0.010	-0.007
arithmetic floating strike	0.016	0.011	-0.006
geometric fixed strike	0.017	0.010	-0.007
geometric floating strike	0.016	0.010	-0.006

Using the control variate method results in a meaningful reduction in the standard error of the price. To put this into context in order to achieve the same standard error without using the control variate method for an arithmetic fixed strike call, the number of simulations would need to be increased from 100,000 to 300,000.

In addition the mean of the control variate does not differ materially from the standard method (so the method is unbiased) and is within the standard error of the closed form solutions (where they are given) except in the case of the geometric fixed strike call.

Table 11: Variance ratio between the Control Variate and Standard method for asian call options

	Variance Ratio	1-correlation^2	Difference
arithmetic fixed strike	0.295	0.295	0.000
arithmetic floating strike	0.233	0.233	0.000
geometric fixed strike	0.308	0.308	0.000
geometric floating strike	0.222	0.222	0.000

Table 12: Variance ratio between the Control Variate and Standard method for asian put options

	Variance Ratio	1-correlation^2	Difference
arithmetic fixed strike	0.345	0.345	0.000
arithmetic floating strike	0.417	0.416	0.001
geometric fixed strike	0.335	0.335	0.000
geometric floating strike	0.426	0.426	0.000

Comparing the variance ratios between the two methods in tables 11 and 12 above shows how (11) is confirmed from the data.

6 Conclusion

The difference in price between the geometric and arithmetic averaging forms of asian options can be explained looking at the payoff and knowing that for positive real values (which stock prices are) the arithmetic average will always be higher than the geometric average due to the inequality of arithmetic and geometric means.

For a fixed strike call, *E[max(Sampling function – K,0)]* will be higher for the arithmetic mean as the sampling function is higher. For a floating strike call *E[max(*S_T *– Sampling function, 0)]* will be lower using arithmetic sampling as the arithmetic average is higher than the geometric average thus reducing the expected payoff.

The same goes for puts. The final data is summarised in tables 13 and 14.

Table 13: Asian call option prices using control variate method

	geometric	arithmetic	difference
fixed strike	5.534	5.751	-0.217
floating strike	6.082	5.873	0.209

Table 14: Asian put option prices using control variate method

	geometric	arithmetic	difference
fixed strike	3.45	3.33	0.12
floating strike	3.29	3.42	-0.13

7 References

Ahmed, R. (2017). *CQF Module 3 Workshop Notes.* CQF.

Glasserman, P. (2003). *Monte Carlo Methods in Financial Engineering.* New York: Springer.

Stavri, K. (2014). *Theoretical and Numerical Schemes for Pricing Exotics.* Unversity College London.