Pricing Asian Options via Monte Carlo methods

Juan Pablo Araujo
CQF - Module 3 Assignment

May 10, 2017

Contents

1 Introduction

In this document we seek to value Asian Options via Monte Carlo methods, and explore how the option value behaves when parameters change. We also explore the stability and convergence of the Monte Carlo method.

1.1 Asian Options

Asian options are strongly path dependent[1] contracts. In particular, Asian's payoff is a function of the average price of the underlying during the options life. For the purpose of this work, we consider 'European type' Asian options.[2]

Asians payoff come in two forms: the fixed strike (or average price) Asian, and the floating strike (or average strike) Asian.

- Fixed strike

$$Call_{fixed} = max(S_M - E, 0)$$
$$Put_{fixed} = max(E - S_M, 0)$$

- Floating Strike

$$Call_{floating} = max(S_T - E_M, 0)$$
$$Put_{floating} = max(E_M - S_T, 0)$$

Where, S_M is the average value of the underlying asset on the average period M, E is the fixed strike, S_T is the value of the underlying asset at expiration date T, and E_M is the floating strike which is the average value of the underlying asset on the average period M. The average value also comes in two forms: arithmetic and geometric average.

- Arithmetic average: $S = \frac{1}{M}\sum_{i=1}^{M} S(t_i)$
- Geometric average: $S = (\prod_{i=1}^{M} S(t_i))^{\frac{1}{M}}$

The latter applies for discrete sampling which is the focus of this work. In practice, Asian options are monitored discretely, but with enough observations we can consider it virtually continuous. The discrete solution empirically appears to converge to the continuous one with $1/N$, where N is the number of observations.

1.2 Monte Carlo Method

Monte Carlo is a method to calculate integrals or expectations using random numbers and probabilities. One of the main benefits of this method is that it is easily implemented and can efficiently be used to value a large spectrum of 'European style' exotic options. It also provides accurate enough results as long as sufficient number of sample paths are simulated. Although it works well with 'European style' options, it is difficult to be applied for 'American style' options.

For the purpose of this work, we will simulate paths assuming the underlying follows a Geometric Brownian Motion $\frac{dS_t}{S_t} = rdt + \sigma dX_t$, where X_t is the Wiener process, and r is the risk free rate

[1] The contract has time dependency (is time-inhomogeneous). For Asians in particular, there is a 'strong path dependency', meaning that an extra variable needs to be introduced in the valuation. In this case, the payoff not only depends on the value of the underlying at the present time, but is also a function of the underlying average.

[2] The option may only be exercised at the expiration date.

(valuation should be under the risk neutral measure). In this case, we can write the value of the option on the underlying S in the form $V(S,t) = \mathbb{E}^{\mathbb{Q}}[e^{\int_t^T r_\tau d\tau} Payoff(S_T)]$. Considering a fixed risk free interest rate r, the equation can be written: $V(S,t) = e^{-r(T-t)}\mathbb{E}^{\mathbb{Q}}[Payoff(S_T)]$. Finally, replacing the expectation we get $V(S,t) = e^{-r(T-t)}\frac{1}{N}\sum_{i=1}^{N} Payoff(S_i)$, which is the average of the discounted Payoffs under the risk neutral measure. For the simulation of the underlying, we will use two methods: the Forward Euler-Maruyama method (E-M), which is a method to approximate a numerical solution to the stochastic differential equation (SDE) $\frac{dS_t}{S_t} = rdt + \sigma dX_t$, and the discretization for the close form solution to the previous SDE:

- Euler-Maruyama: $S_t = S_{t-1}(1 + r\delta t + \sigma\phi\sqrt{\delta t})$
- Closed form: $S_t = S_{t-1}exp((r - \frac{\sigma^2}{2})\delta t + \sigma\phi\sqrt{\delta t})$

Figure 1: Simulations under the E-M method

1.3 Implementation: Python

For the implementation of the above methods we use Python. It's a high level powerful programming language with an elaborate ecosystem of tools an libraries. Some of the advantages of this language presents [1]:

- Open source: Python and the majority of supporting libraries and tools available are open source.
- Syntax: Easy to learn, code is compact and highly readable.
- Interpreted: Is an interpreted language which makes rapid prototyping and development in general more convenient.

- Libraries: There is a wide range of powerful libraries available and supply grows steadily.
- Speed: In spite is an interpreted language, code execution speed is similar to languages like C or C++.
- Market: Python is getting popular at large financial institutions and in the hedge fund industry.

2 Results

2.1 Outline of numerical procedure

The numerical procedure used to obtain the value of Asian options is as follows:

- For 1 to the number of paths (N) simulated
  - Simulate paths using E-M or closed form method. Here we get a matrix of N columns (sample paths) and t rows (number of time steps used)
- Calculate arithmetic and geometric average for the average period (M). Here, for each average type, we build an array of N elements. Each element is the average for the M period of each path.
- Now that we have the averages and the paths, we calculate the payoff of calls and puts, for the fixed and floating strike case, and for the arithmetic and geometric average case.
- Discount the payoff at the risk-free rate (risk neutral measure) and calculate the mean.

Then we perform several tests in order to see how the option value behaves when other parameters change: number of simulations (N), average window (M), frequency of sampling, time step, and stock price in t0.

2.2 Simulations

We start by simulating the option value varying the number of simulations. For this case we use: $S_0 = 100$, $T = 1$, $E = 100$, $\sigma = 20\%$, and $r = 5\%$. Where S_0 is the starting price of the underlying, T is the time to maturity, E is the strike price, σ is the volatility and r the risk free rate.

For our first excercise, we use 1.000 time steps and we determine M (average period) to be 1.000. This means that the time to maturity is equal to the length of the average period (we use the whole path for the average). We do this in order to compare our results to closed form solutions for some particular cases of Asian options (fixed and floating with geometric average)[3].

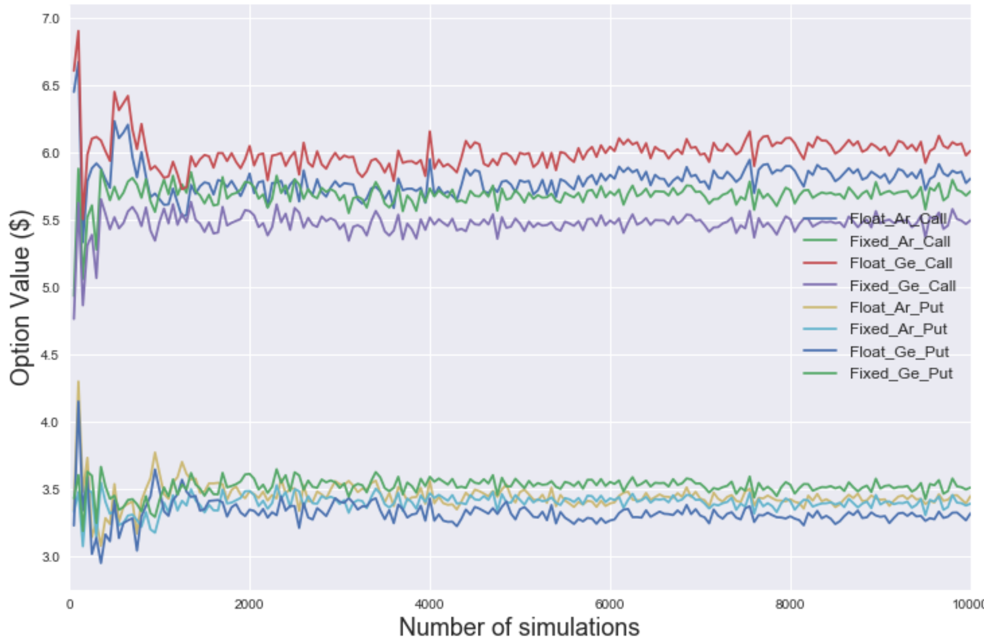

Figure 2: Simulations under the E-M method. We show up to 10.000 simulations

	Calls				Puts				
	Arithmetic		Geometric		Arithmetic		Geometric		MC Error
N	Floating	Fixed	Floating	Fixed	Floating	Fixed	Floating	Fixed	$\frac{\sigma}{\sqrt{N}}$
100	6.67	5.88	6.91	5.63	4.30	3.48	4.15	3.60	2.0×10^{-2}
150	5.33	5.06	5.51	4.86	3.48	3.07	3.34	3.19	1.6×10^{-2}
250	5.89	5.61	6.11	5.39	3.14	3.47	3.01	3.60	1.2×10^{-2}
500	6.24	5.75	6.45	5.52	3.54	3.34	3.42	3.45	9.0×10^{-3}
1.000	5.65	5.70	5.86	5.49	3.62	3.30	3.49	3.42	6.0×10^{-3}
2.500	5.80	5.80	6.01	5.59	3.50	3.50	3.37	3.62	4.0×10^{-3}
5.000	5.80	5.69	6.00	5.47	3.48	3.42	3.35	3.54	3.0×10^{-3}
10.000	5.81	5.71	6.01	5.50	3.45	3.39	3.32	3.51	2.0×10^{-3}
20.000	5.90	5.75	6.11	5.54	3.37	3.37	3.25	3.49	1.4×10^{-3}
30.000	5.89	5.76	6.10	5.54	3.39	3.36	3.27	3.47	1.2×10^{-3}
40.000	5.82	5.77	6.03	5.56	3.41	3.36	3.28	3.48	1.0×10^{-3}
50.000	5.80	5.79	6.01	5.57	3.41	3.38	3.28	3.50	9.0×10^{-4}
60.000	5.80	5.74	6.00	5.52	3.43	3.35	3.30	3.47	8.0×10^{-4}
70.000	5.84	5.78	6.05	5.56	3.45	3.39	3.32	3.51	7.5×10^{-4}
80.000	5.81	5.74	6.02	5.52	3.44	3.36	3.32	3.48	7.1×10^{-4}
90.000	5.81	5.75	6.02	5.54	3.45	3.38	3.32	3.50	6.7×10^{-4}
100.000	5.82	5.74	6.02	5.52	3.42	3.37	3.29	3.49	6.3×10^{-4}
1.000.000	5.86	5.76	6.07	5.54	3.40	3.35	3.28	3.47	2.0×10^{-4}

Table 1: Option value for varying N Monte Carlo simulations using E-M method

As it can be observed, at around 10.000 simulations we get a reasonable convergence. In particular, for this case, exact values for Calls fixed and floating geometric strike are 5.5468 and 6.0723 respectively, and for Puts fixed and floating geometric strike 3.4633 and 3.2788, so we can see that we get a fairly close result with the Monte Carlo simulation[3]. As Monte Carlo error has order $\frac{\sigma}{\sqrt{N}}$, 10.000 simulations provide a reasonable approximation for a two decimal place accuracy solution (error of 2.0×10^{-3}).

Also, the following relations can be verified (see proof in [3]): $Call_{fixedarithmetic} \geq Call_{fixedgeometric}$, $Call_{floatarithmetic} \leq Call_{floatgeometric}$, $Put_{fixedarithmetic} \leq Put_{fixedgeometric}$, $Put_{floatarithmetic} \geq Put_{floatgeometric}$.

With respect to the time required to compute the simulations, the following figure shows time required to process up to 100.000 simulations:

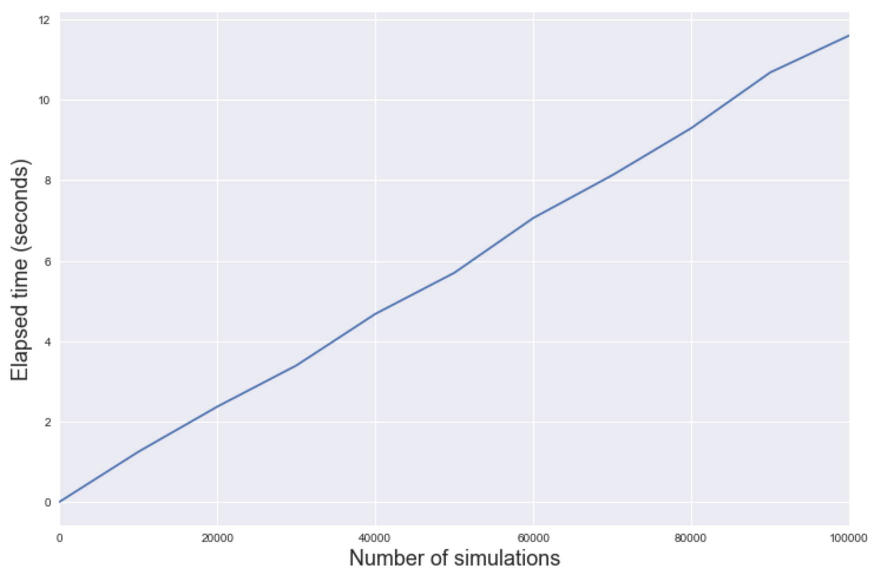

Figure 3: Elapsed time of computing process up to 100.000 simulations

It can be seen that up to 100.000 simulations, process time grows approximately linear with the number of simulations.

2.2.1 Average time window M

In last section we assumed the average window M used to calculate the average was equal to the number of time steps of the simulation (we calculated the average for the whole period or life of the option). Now we change the average time window in order to see how the option value changes. In particular, we evaluate from M equal to one time step to M equal to the time to maturity (whole period)[4]. We keep the values of the other parameters fixed.

[3]See [3] for details on the closed forms solutions to get the exact results.

[4]We calculate the value of Forward-starting Asian options going from one time step, up to the whole period.

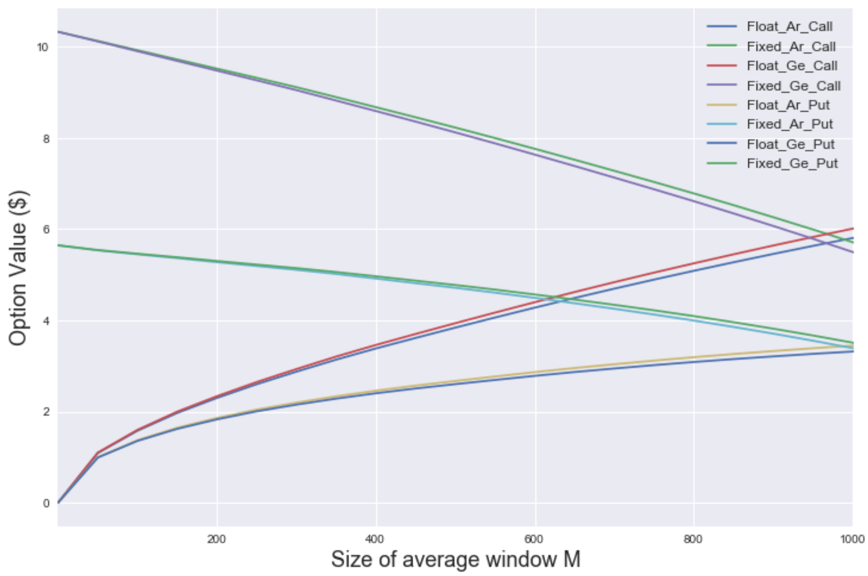

Figure 4: Option value while average time window changes. We go up to 1.000 which covers the whole period.

We can see how for the floating strike calls and puts, the value of the option starts at 0. This is because for a 1 time step average window, the average is the same as S_T so the payoff is 0. Also, we can note that for fixed strike calls, value starts at around 10.33 which approximates to the value of the European plain vanilla call. This also makes sense as for a 1 time step average window, the fixed strike Asians are equivalent to a European plain vanilla call. Additionally, it's interesting to note that fixed strike calls and puts decrease their value while average time window increases, while the opposite happens with floating strike calls and puts. For the fixed strike case, the "smoothing" effect of the average reduces the volatility, i.e reducing it's value. For the floating strike case, as the window increases, the chance of S_T being far from E_M increases, so the volatility increases, i.e increasing it's value.

2.2.2 Frequency of sampling

Up to now, we have considered a 'continuous' sampling in order to calculate the average (we calculate averages considering all time steps in the path of the option). In this section we are going to sample the path of the option, and calculate the average using this sample. To sample each path, we equally distribute the samples in time through the option path. The following figure shows the procedure:

Figure 5: Example for a path sampled 4 times

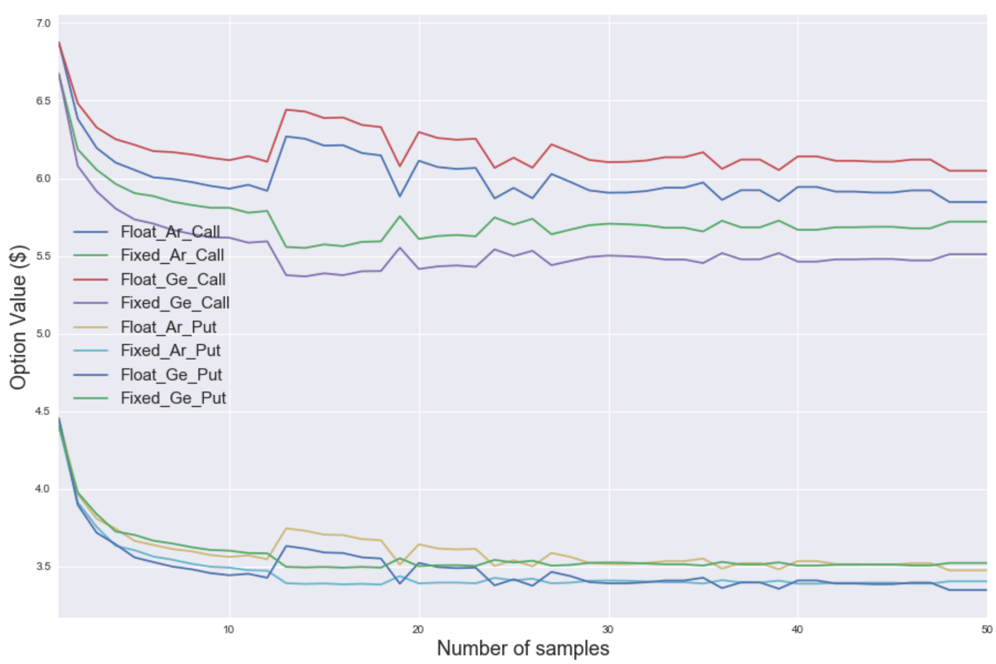

Figure 6: Option values vs. number of samplings

It is interesting to point out that considering a 1.000 time step simulation, after 30 equally distributed samples we have a reasonable convergence to the 'continuous' value. The next figure shows how close are the continuous and sampled average in a path sampled 25 times:

Figure 7: Example for a path sampled 25 times

2.2.3 Time step

Up to now we have considered 1.000 time steps for the valuation of the options. Here we explore how the option value varies as the time step used on the simulation varies. We assume we sample continuously and through the whole path (M equals de number of time steps). It is interesting to see that only about 30 time steps are needed to achieve a reasonable convergence.

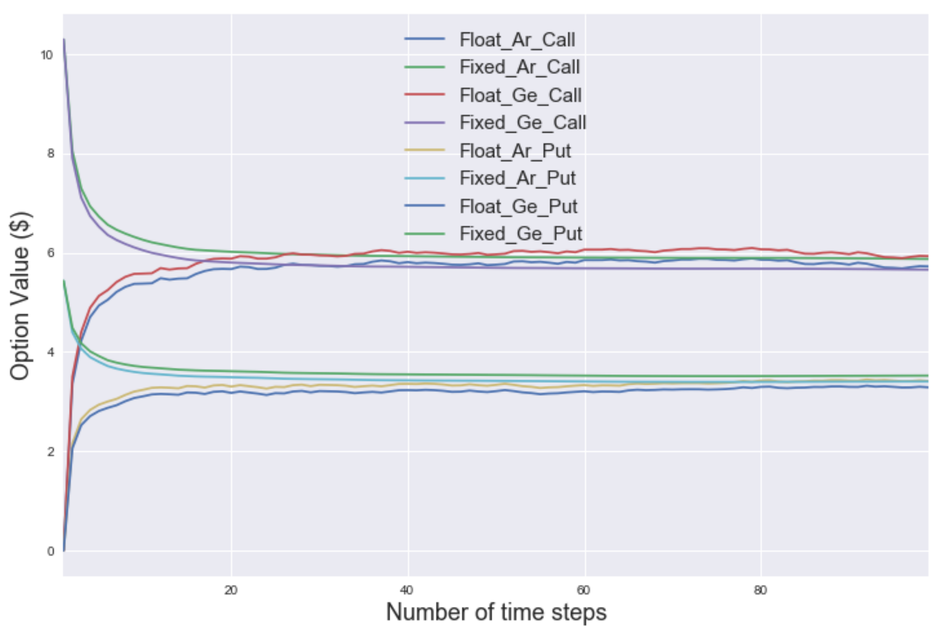

Figure 8: Option value vs. number of time steps used to simulate

2.2.4 Initial underlying value S_0

Previous examples have considered $S_0 = 100$. However is interesting to see how the price of the option changes as S_0 changes, holding all other variables constant:

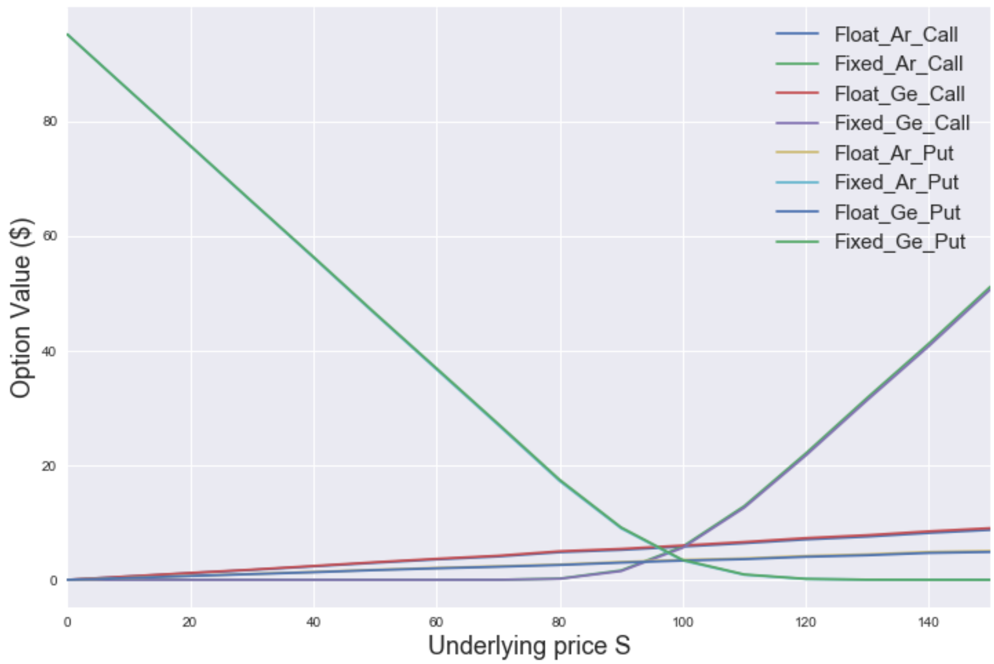

Figure 9: Option value vs. Underlying price

It can be observed that in-the-money fixed strike options are more expensive than in-the-money floating strike options. This can be explained due to the effect that the floating strike tends to follow the price of the underlying. For example, in the case of a Call with strike of \$100, if the underlying price starts at \$140 and stays there until expiration, fixed strike will get a payoff of \$40, but floating strike will average \$140 so will get a payoff of \$0. The contrary happens with out-of-the money options. As the floating strike tends to follow the underlying, there is a higher probability for the option to end up in-the-money with respect to fixed strike options, i.e out-of-the-money floating strikes are more expensive than fixed strike options.

2.2.5 Closed form vs. E-M method

Up to now we have simulated using the Forward Euler-Maruyama method, which is a numerical approximation to the solution of the stochastic differential equation (SDE) $\frac{dS_t}{S_t} = rdt + \sigma dX_t$. But how does the simulation change if we use the discretization for the close form solution to the previous SDE?. Here we re-do the simulations done at the beginning of section 2.2 using the following discretization for path simulation: $S_t = S_{t-1} exp((r - \frac{\sigma^2}{2})\delta t + \sigma\phi\sqrt{\delta t})$

	Calls				Puts				MC Error
	Arithmetic		Geometric		Arithmetic		Geometric		
N	Floating	Fixed	Floating	Fixed	Floating	Fixed	Floating	Fixed	$\frac{\sigma}{\sqrt{N}}$
100	6.67	5.88	6.91	5.63	4.30	3.48	4.15	3.60	2.0×10^{-2}
150	5.33	5.06	5.50	4.86	3.47	3.07	3.33	3.19	1.6×10^{-2}
250	5.89	5.61	6.11	5.39	3.14	3.47	3.01	3.59	1.2×10^{-2}
500	6.24	5.75	6.45	5.52	3.54	3.34	3.42	3.45	9.0×10^{-3}
1.000	5.66	5.70	5.86	5.49	3.62	3.30	3.49	3.41	6.0×10^{-3}
2.500	5.80	5.80	6.01	5.59	3.50	3.50	3.37	3.62	4.0×10^{-3}
5.000	5.80	5.69	6.00	5.47	3.48	3.42	3.35	3.54	3.0×10^{-3}
10.000	5.81	5.71	6.01	5.50	3.44	3.39	3.32	3.51	2.0×10^{-3}
20.000	5.90	5.75	6.11	5.54	3.37	3.37	3.25	3.49	1.4×10^{-3}
30.000	5.89	5.76	6.10	5.54	3.39	3.36	3.27	3.47	1.2×10^{-3}
40.000	5.82	5.77	6.03	5.56	3.41	3.36	3.28	3.48	1.0×10^{-3}
50.000	5.80	5.79	6.01	5.57	3.41	3.38	3.28	3.50	9.0×10^{-4}
60.000	5.80	5.74	6.00	5.52	3.43	3.35	3.30	3.47	8.0×10^{-4}
70.000	5.84	5.78	6.05	5.56	3.45	3.39	3.32	3.51	7.5×10^{-4}
80.000	5.81	5.74	6.02	5.52	3.44	3.36	3.32	3.48	7.1×10^{-4}
90.000	5.81	5.75	6.02	5.54	3.45	3.38	3.32	3.50	6.7×10^{-4}
100.000	5.82	5.74	6.02	5.52	3.42	3.37	3.29	3.49	6.3×10^{-4}
1.000.000	5.86	5.76	6.07	5.54	3.40	3.35	3.28	3.47	2.0×10^{-4}

Table 2: Option value for varying N Monte Carlo simulations using discretization of closed form method

Considering two decimal rounding, results are almost identical to the E-M simulations shown in 2.2.

3 Conclusions

Throughout this work, we valued Asian Options via Monte Carlo methods, and explore how the option value behaves when parameters change. We also explored the stability and convergence of the Monte Carlo method.

We started by calculating option values for Calls and Puts, with floating and fixed strikes, using arithmetic and geometric averages. A reasonable convergence is achieved around 10.000 MC simulations.

We observed how the option price behaved when the size of the average window M changed. We could see that for a window of 1 time step, fixed strikes options turn into plain vanilla European options and floating strike options are worthless. Also we could observe that fixed strike calls and puts decrease their value while average time window increases, while the opposite happens with floating strike calls and puts.

Frequency of sampling was also tested. We observed that considering a 1.000 time step simulation, after 30 'equally distributed sampling during the life of the option' we achieve a reasonable convergence to the 'continuous' value[5].

We also tested how changing time steps on the simulation changes the option value. Interestingly we saw that just with about 30 time steps we achieve a reasonable convergence.

Also we explored how the initial underlying value of the underlying affects the option value. We could note that in-the-money fixed strike options are more expensive than in-the-money floating strike ones. The contrary happens with out-of-the-money options.

Finally, we tested how option value changes if we change the path generation scheme from Forward Euler-Maruyama to the discretization of the closed form solution. We saw that considering two decimal places, results are almost identical.

[5] We assume continuous value as sampling all time steps.

References

[1] Y. Hilpisch, *Derivatives Analytics with Python: Data Analysis, Models, Simulation, Calibration and Hedging.* The Wiley Finance Series, Wiley, 2015.

[2] P. Wilmott, *Paul Wilmott on quantitative finance.* John Wiley & Sons, 2013.

[3] K. Stavri, "Theoretical and numerical schemes for pricing exotics," Master's thesis, 2014.

[4] P. Jäckel, *Monte Carlo methods in finance.* J. Wiley, 2002.