Asian Option Pricing using Monte Carlo

Tanya Sandoval

April 27, 2016

1 Asian Option Pricing using Monte Carlo

This report presents the results of using Monte Carlo (MC) to price an **Asian Call option** as given by:

$$V(S,t) = \mathbb{E}^{\mathbb{Q}}[e^{-\int_t^T r dr} \text{Payoff}(S_T)] \tag{1}$$

The model was developed using **Python 2.7**. All the related code can be found elsewhere.

For brevity, the results for a put option have been omitted and also the details of the theory and mathematics of Asian options. The documentation of the *Milstein* convention and *antithetic variance reduction* technique can also be found elsewhere.

1.1 Initial parameters

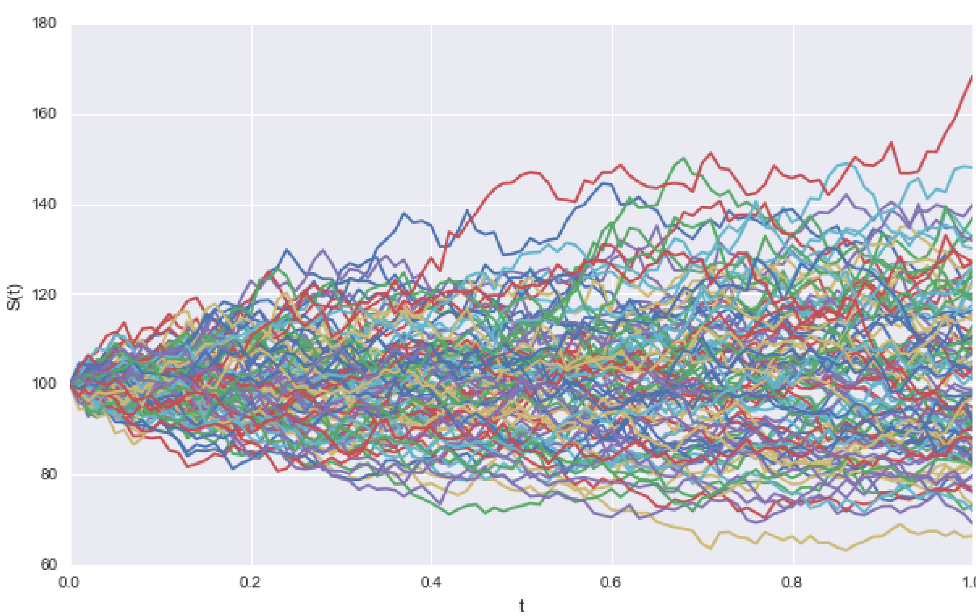
The following parameters are used for the simulation, unless otherwise stated:

- Today's stock price S0 = 100
- Strike E = 100
- Time to expiry (T-t) = 1 year (100 time steps)
- Volatility = 20%
- Constant risk-free interest rate r = 5%

1.2 Stock Path Generation

We start by using MC to generate 100 stock price paths using the *Milstein* convention. In this process we use the antithetic variance reduction technique which increases the statistics to compute the option price with minimal computational effort.

The below plot shows 100 of these paths:

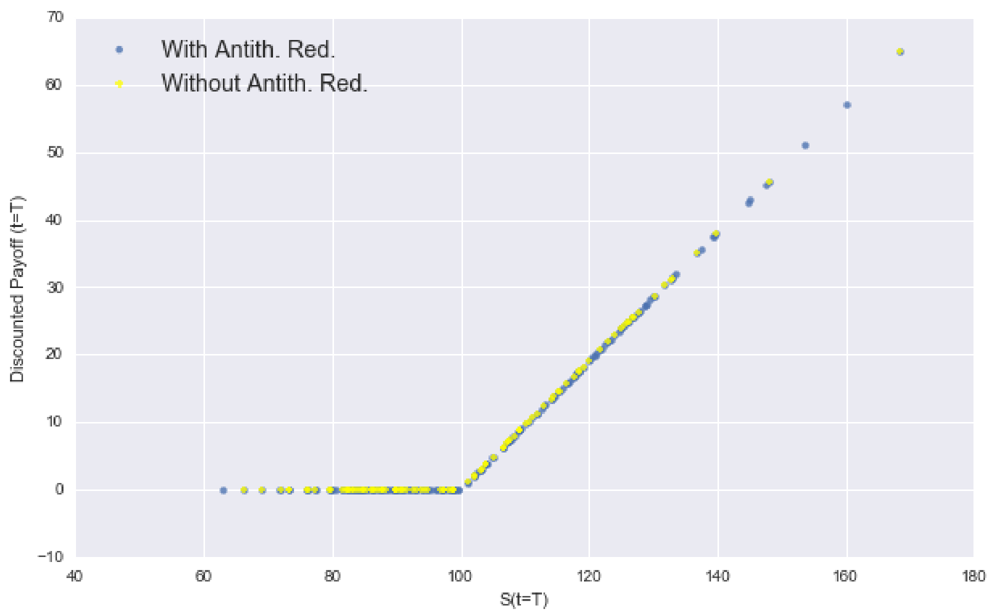


1.3 European Call Pricing

The same methodology is used to price a European Call option in order to compare to the Asian results later on. For the same input parameters above the following result is obtained (later one we discuss the error in the estimate):

```
EU V(T) = 10.089201 (0.131768)
```

The corresponding Payoff function at expiry (t=T) is shown below, where the effect of the variance reduction technique is shown to boost the statistics. This hence reduces the estimate error at little computational cost.



1.4 Asian Call Pricing

An Asian option (or average value option) falls in the *exotics* options class. For Asian options the payoff is determined by the average of the underlying price over some pre-set period of time, hence it is a path-dependent instrument.

An Asian Call value is then calculated using the same parameters above but with different methodologies to calculate the average of the stock price:

- **Arithmetic** averaging using **both continuous** ('AC_c') and **discrete** ('AC_d') sampling
- **Geometric** averaging using **continuous** sampling **only** ('GC_c')

Note in the context of MC, we refer to 'continous sampling' when all the data points available up to that time are used, whereas 'discrete sampling' uses a sample determined by sampling period k (10 days in this case).

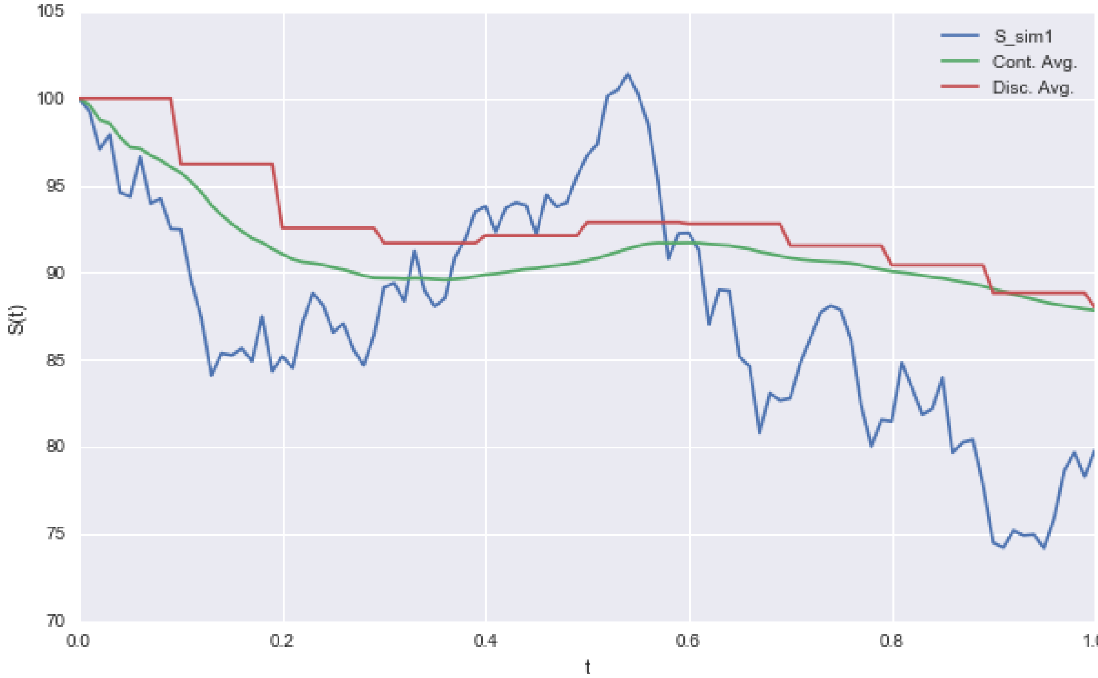
For all the examples below a **fixed strike** is assumed unless otherwise stated. This is defined as

$$C(T) = \max\left(A(0,T) - K, 0\right) \tag{2}$$

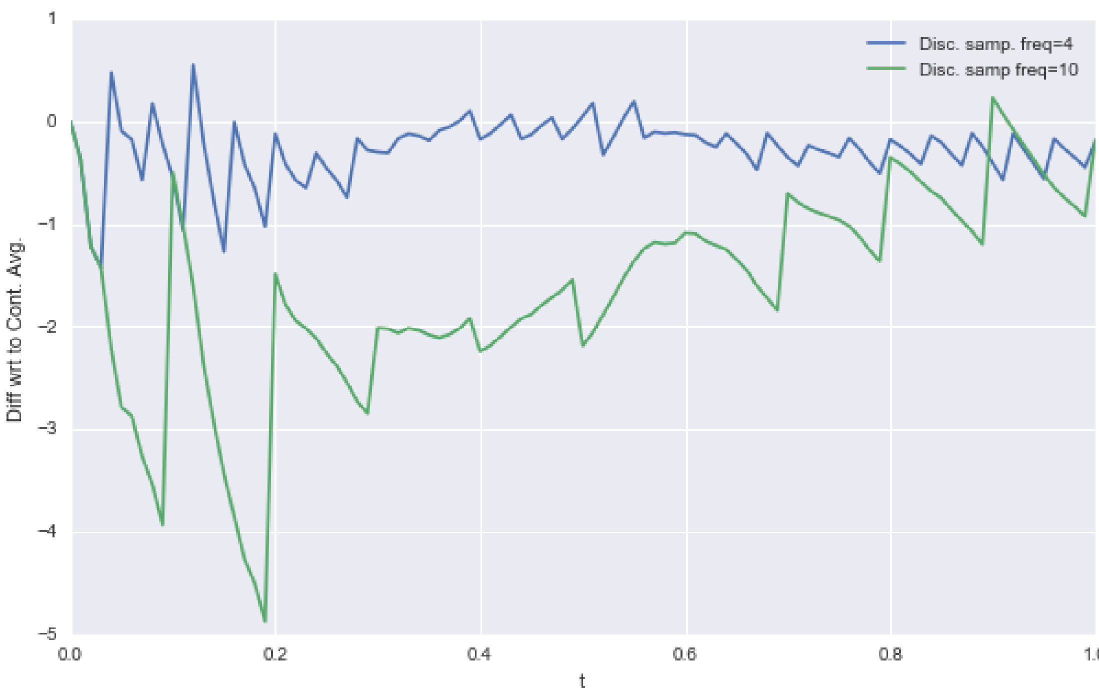
1.4.1 Continuous vs Discrete Sampling

First let's understand the effects of continuous vs discrete sampling. For this purpose we restrict to the Arithmetic case only and assume the Geometric case observes the same behaviour.

Plotting one of the simulated stock paths and corresponding arithmetic continuous and discrete averages (using a sampling period of 10 days), we get the following:

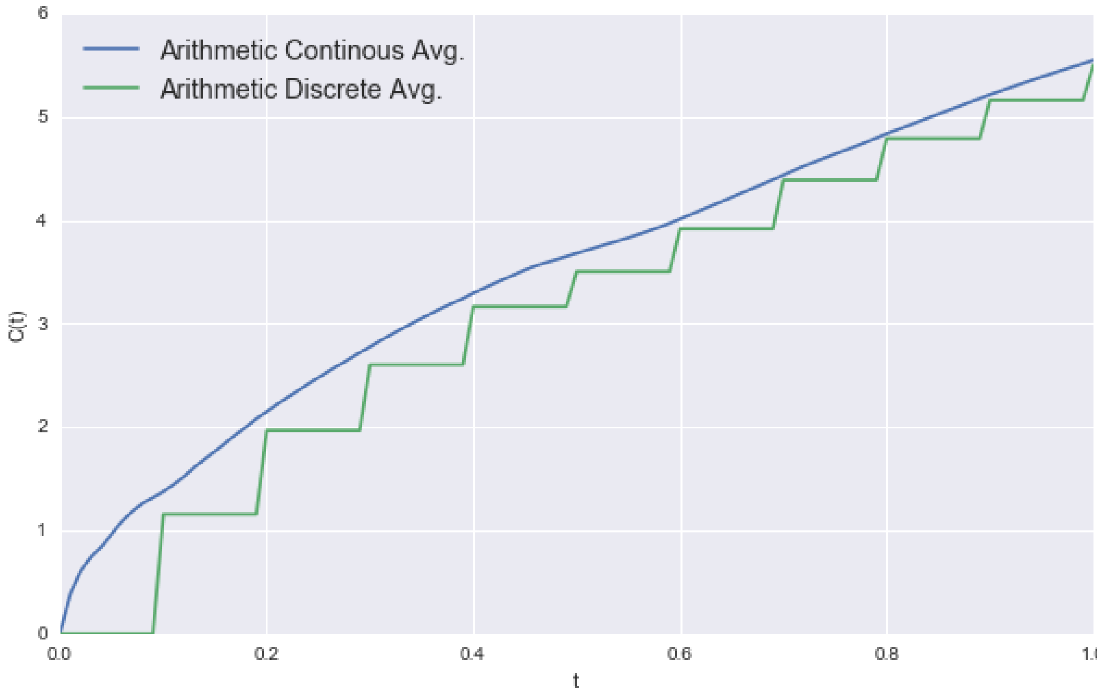


Hence we note that A_c and A_d differ slightly. This difference grows with the sampling period k because for the same time to expiry we have less data points to calculate the average. This behaviour is shown in the plot below where d_10 is more unstable than d_4. As the number of data points increase, we expect the difference to get smaller, shown by the convergence trend close to expiry



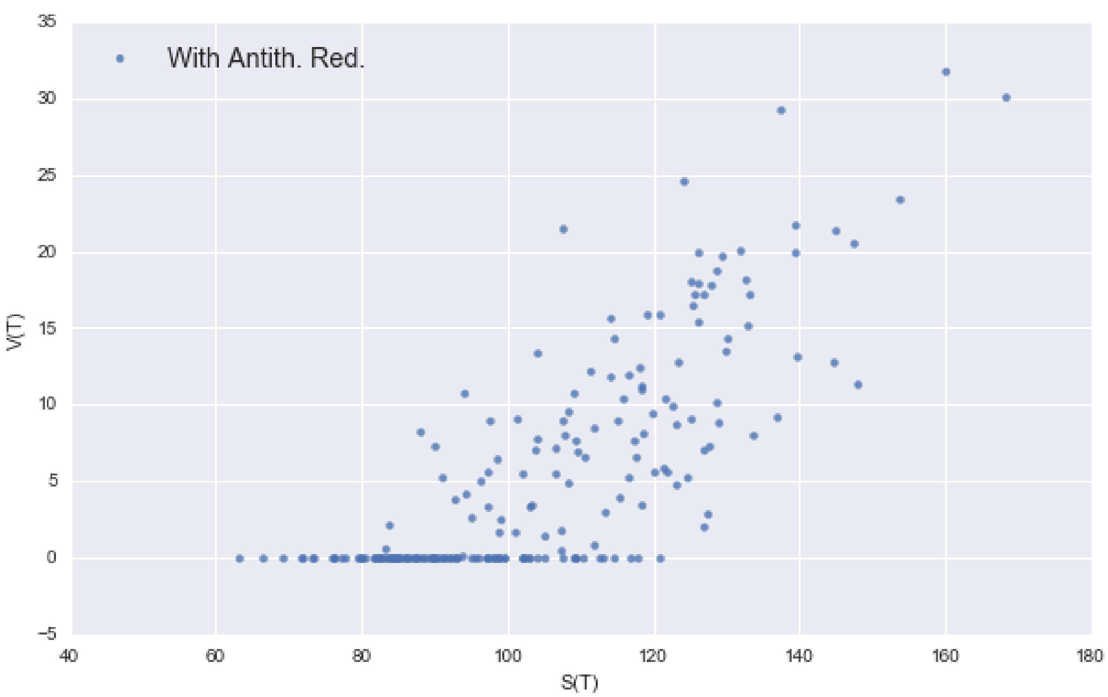
However, for practical and legal reasons path-dependent quantities are never measured continuously. There is minimum time step between sampling since it is difficult to incorporate every single traded price into an average, for example, data can be unreliable and the exact time of a trade may not be known accurately.

The option price also evolves with number of time steps to expiry and whether the average is calculated discretely or continously, as shown in the below plot:



1.4.2 Payoff function

The Payoff function at expiry $t = T$ of the Asian call is also a lot more complex than it's European equivalent, which is expected given it's path-dependent nature. This justifies the need for using Monte Carlo methods for the pricing.



1.5 Results and Errors

The results from varying the parameters in the model to study the effects on the pricing are presented now. The key for all below tables is the following:

- N_S: number of simulations
- V: European call equivalent
- AC_c: Arithmetic continuous average Asian Call price
- AC_d: Arithmetic discrete average Asian Call price
- GC_c: Geometric continuous average Asian Call price
- The error is represented in brackets next to the option values and is defined as the standard deviation of the estimate divided by the square root of the number of simulations (std/$\sqrt{N}$)
- All the option values and errors are displayed to 6 significant figures

1.5.1 Varying the Number of Simulations

We see that as the number of simulations increases, the error in the estimate decreases (approximately by a factor of 10 every time the simulations increase by a factor of 10). Also note that the effect of the variance reduction technique is to effectively increase N_S by a factor of 2, which is why the error presented is smaller than for the N_S values shown. Run times weren't compared since the underlying python code for any of these cases takes less than 2 seconds and this was deemed quite acceptable.

Out[9]:

	N_S	r	sigma	S0	K	Class	V	AC_c	AC_d	GC_c
0	100.0	0.05	0.2	100.0	100.0	ATM	10.089201 (0.131768)	5.546591 (0.071872)	5.499740 (0.071726)	5.333837 (0.069442)
1	1000.0	0.05	0.2	100.0	100.0	ATM	10.610881 (0.015165)	5.783801 (0.008124)	5.707166 (0.008020)	5.564119 (0.007840)
2	10000.0	0.05	0.2	100.0	100.0	ATM	10.388515 (0.001471)	5.771337 (0.000798)	5.685983 (0.000786)	5.552812 (0.000770)

1.5.2 Varying the Strike

In this case we see the following:

- ITM option is of course more expensive than OTM option for both the Asian and European Call
- The difference between the stock price (S0) and exercise price (K) has little effect on the estimate error (although there is perhaps a slight increase as the option evolves from OTM to ITM)
- The methodology used to calculate the average of the stock (continous vs dicrete, arithmetic vs geometric) for the Asian option seems to have little effect in the error too
- The price of the European equivalent is always higher and by a considerable amount. However the difference (diff_GC_c) seems similar across ITM, ATM and OTM (although this might not always hold depending on other parameters values)

Out[10]:

	N_S	r	sigma	S0	K	Class	V	AC_c	AC_d	GC_c	diff_GC_c
0	10000.0	0.05	0.2	100.0	95.0	ITM	13.306567 (0.001611)	8.825431 (0.000938)	8.753649 (0.000925)	8.575778 (0.000912)	4.730789
1	10000.0	0.05	0.2	100.0	100.0	ATM	10.388515 (0.001471)	5.771337 (0.000798)	5.685983 (0.000786)	5.552812 (0.000770)	4.835703
2	10000.0	0.05	0.2	100.0	105.0	OTM	7.957399 (0.001319)	3.514907 (0.000641)	3.432210 (0.000630)	3.330691 (0.000612)	4.626708

1.5.3 Varying the Volatility

The error appears to significantly increase when the volatility of the stock increases. This is expected since an increase in uncertainty in stock prices undermines the predictability of the model itself. The increase in option price with volatility is also readily confirmed.

Out[11]:

	N_S	r	sigma	S0	K	Class	V
0	10000.0	0.05	0.2	100.0	100.0	ATM	10.388515 (0.001471)
1	10000.0	0.05	0.3	100.0	100.0	ATM	14.135187 (0.002251)
2	10000.0	0.05	0.4	100.0	100.0	ATM	17.893775 (0.003127)

	AC_c	AC_d	GC_c
0	5.771337 (0.000798)	5.685983 (0.000786)	5.552812 (0.000770)
1	7.961140 (0.001204)	7.832518 (0.001187)	7.507157 (0.001140)
2	10.158977 (0.001646)	9.986547 (0.001624)	9.379621 (0.001526)

1.5.4 Varying the interest rate

In contrast to volatility, varying the interest rate has a negligible effect on the error. The increase in option value seems to be slower than that observed with volatility variation.

Out[12]:

	N_S	r	sigma	S0	K	Class	V
0	10000.0	0.03	0.2	100.0	100.0	ATM	9.352018 (0.001411)
1	10000.0	0.05	0.2	100.0	100.0	ATM	10.388515 (0.001471)
2	10000.0	0.07	0.2	100.0	100.0	ATM	11.485530 (0.001528)

	AC_c	AC_d	GC_c
0	5.293044 (0.000773)	5.204118 (0.000761)	5.094293 (0.000747)
1	5.771337 (0.000798)	5.685983 (0.000786)	5.552812 (0.000770)
2	6.268640 (0.000822)	6.188231 (0.000811)	6.028783 (0.000792)

1.5.5 Float vs Fixed Strike

For the Asian call option the float strike was defined here as:

$$C_{float}(T) = \max\left(S(T) - A(0,T), 0\right) \tag{3}$$

i.e. the strike is taken to be the value of the stock at expiry.

Within errors, the float and fixed strike Asian options seem to be nearly the same, suggesting an 'equivalence' relationship between them.

Out[13]:

	N_S	mode	r	sigma	S0	AC_c
0	10000.0	fixed	0.05	0.2	100.0	5.771337 (0.000798)
1	10000.0	float	0.05	0.2	100.0	5.808603 (0.000846)

	AC_d	GC_c
0	5.685983 (0.000786)	5.552812 (0.000770)
1	5.715516 (0.000831)	6.019974 (0.000874)

1.6 Summary

In summary, Monte Carlo seems a suitable method to efficiently price European and Asian options. The number of simulations has the biggest effect on the option price accuracy, followed by volatility. The variance reduction technique proved to be useful to reduce significantly the error at little computing cost. An equivalence between float and fixed strikes in the pricing seems to suggest an equivalence, although to validate this, more test cases should be examined. For all test cases, the Asian option price appears considerably lower than the European, making it an attractive financial instrument for certain risk management strategies.