Monte-Carlo methods for Asian option pricing

Dorian Baudry

November 3, 2017

Abstract

The goal of this short paper is to briefly explain the theoretical issues of my project about Monte-Carlo methods for Asian options pricing. I first introduce the definition of an Asian option and the model I used, and then the principles of variance reduction techniques.

1 About the model

1.1 Asian option

A financial option is a financial asset which payoff depends on the evolution of the price of others financial assets. In this study the price of this underlying asset at the date t will be noted S_t.
The payoff of the Asian option is simply defined as the difference between the average of the price of the underlying over a sequence of the time period $[0, T]$ and the strike K. One can use the arithmetic or geometric mean, I only consider the arithmetic in this project.
So considering a constant short-term interest rate r, the pricing formula at time 0 of an Asian call is :

$$P_0 = e^{-rT} \times E((\frac{1}{N}\sum_{i=0}^{N-1} S_{t_i} - K)^+)$$

We don't provide more details on this section (about risk neutral measure, etc..) because this project focuses on Monte Carlo methods, not financial mathematics.

1.2 Model for the underlying

I choose to use a geometric brownian motion process to sample the price of the Underlying of my Asian option.
This process satisfies the equation : $dS_t = \mu S_t dt + \sigma S_t dW_t$, where W_t is a Brownian motion.
So the process can be written as follow :

$$S_t = S_0 e^{(\mu - \frac{\sigma^2}{2})t + \sigma W_t}$$

In order to sample the underlying among the times $[0, t_1, t_2, ..., t_{N-1}]$ I used the derived formula :

$$S_{t_{k+1}} = S_{t_k} e^{(\mu - \frac{\sigma^2}{2})\delta_t^k + \sigma \delta_t^k N_k}$$

where N_k $N(0, 1)$, using the properties of the Brownian motion. Note that to simulate J paths of the underlying process we only have to sample $J \times N$ normal variables and to apply this function.

2 Monte Carlo methods

2.1 Parameters and main issues

Here is the short list of the parameters needed by the model :

- S_0 the initial value of the underlying
- μ, σ the parameters of the Brownian motion process

- r the short term interest rate, used in the discount factor
- K the strike price
- $[0, t_1, t_2, ..., t_{N-1} = T]$ the set of dates, T is called time to maturity
- the number of samples J for the Monte Carlo estimators

If we work in a financial approach two main issues must be considered :

- With a fixed J, the estimator of the price must be as precise as possible : if you want to invest a large amount of money in a financial asset you can't take the risk of a wrong estimation. The speed of
- The time for the estimation must be as short as possible. Time is a necessary criterion in finance, as you work in a very competitive environment.

  The aim of the next part is to choose the appropriate Monte Carlo pricing method, discussing these 2 issues. They can be combined with indicators such as the speed of convergence for example.

2.2 Standard Monte-Carlo method

This method is described in 1.2, since we have the J paths of the underlying price during the time period, the Monte-Carlo estimator for the price of the Asian option is :

$$\widetilde{P_0} = \frac{1}{J}\sum_{i=1}^{J} P_0^i$$

where

$$P_0^i = e^{-rT}(\frac{1}{N}\sum_{j=0}^{N-1} S_{t_i} - K)$$

To check out the precision criterion we use the Monte Carlo estimator for the variance of the price, which is basically :

$$\widetilde{V}_{P_0} = \frac{1}{J}\sum_{i=1}^{J}(P_0^i)^2 - \widetilde{P}_0^2$$

Obviously, this very simple method combines the 2 defaults : it is slow and quite unprecise. It is now time to see variance reduction methods.

2.3 Antithetic variable

The principle is very simple : it is about extracting more informations from the same sample. Indeed the purpose of the antithetic method is to create for each i a new variable $\widehat{P_0^i}$ from the same sample that was used to create P_0^i, then to use the Monte-Carlo estimators of $\frac{\widehat{P_0^i}+P_0^i}{2}$ for our pricing.
It is easy to verify that the expectation of this combination of variable is still the price.
Then, since

$$Var(\frac{\widehat{P_0^i} + P_0^i}{2}) = \frac{1}{4}(Var(\widehat{P_0^i}) + Var(P_0^i) + 2Cov(\widehat{P_0^i}, P_0^i))$$

It is easy to check that the empirical variance of this new estimator will be lower that the previous, with an appropriate choice for the control variable (a negative covariance for example).

In the case of the Asian option, the simplest transformation that comes is to change W_t in $-W_t$, which is still a Brownian motion.

2.4 Control variable method

This new method is again about extracting more informations from the same sample, but in different way. This time, we create a completely new variable Y with the same sample and give the Monte Carlo estimators for the variable $Z^i = \widetilde{P_0^i} + c.(Y^i - E(Y))$ with a given c.
The expectation of this new variable is still P_0, and this gives for the variance :

$$Var(Z^i) = Var(P_0^i) + c^2 Var(Y^i) + 2c.Cov(P_0^i, Y^i)$$

Note that for a given Y^i, the optimum for c in term of variance reduction is $c = \frac{-Cov(Y^i, P_0^i)}{Var(Y^i)}$. The corresponding variance is $Var(Z^i) = Var(P^i) - \frac{Cov(P^i, Y^i)^2}{Var(Y^i)}$, which is > 0 (demonstration : Cauchy-Schwartz inequality), and $< Var(P^i)$.
The control variate we use the geometric mean of the underlying price process, instead of the arithmetic mean, which have been proven as a good control variable for this problem. [1] So we use :

$$Y^i = e^{-rT}((\prod_{t=t_0}^{t_{N-1}} S_t)^{\frac{1}{T}} - K)^+$$

According to our criterion this method is the best among this class of Monte Carlo methods

3 Quasi Monte-Carlo and MLMC

3.1 Quasi Monte-Carlo

In the past section we used pseudo-random numbers, and the Law of Large Numbers insured the convergence of our estimators. The problem is that this solution costs a lot of time because the sample has often to be very large to insure a satisfying convergence, even if it is better with reduction variance methods.
The Quasi Monte-Carlo method aims to answer the problem in a different way by abandoning pseudo-random numbers for a low-discrepancy sequence, which is basically a pseudo-random sequence able to fill the space according to a given distribution [2]. This alternative to pseudo-random numbers insure a faster convergence : in $O(\frac{1}{J})$ instead of $O(\frac{1}{\sqrt{J}})$. In the R code I used the Halton, Sobol and Torus sequences.

3.2 Multi-Level Monte-Carlo

To be completed.
For a complete description of Multi-level Monte-Carlo, see Giles Oxford paper published on this subject (2006).

[1] http://perso.telecom-paristech.fr/ gfort/Enseignement/MDI345/VariablesControle.pdf
[2] https://en.wikipedia.org/wiki/Quasi_Monte_Carlo_method