Stochastic Simulations

Autumn Semester 2020

Prof. Fabio Nobile *Assistants: Juan Pablo Madrigal Cianci*

Homework 1: Due date: 18 November 2020

Homework 1: Quasi-Monte Carlo Methods

The following problem is optional and counts for a bonus of up to 0.2 points in the final grade.

An Asian option (or average value option) is a special type of option contract. For Asian options the payoff is determined by the average underlying price over some preset period of time. Consider the problem of pricing an Asian option with maturity $T > 0$ based on the stock price S, which is given as the solution to the stochastic differential equation

$$dS_t = rS_t\,dt + \sigma S_t\,dW_t\ , \quad \text{with } S_0 \text{ given, },$$

where W_t denotes a standard one-dimensional Wiener process. One can show that $S_t = S_0 e^{X_t}$, where $X_t = (r - \sigma^2/2)t + \sigma W_t$ with W being a standard Wiener process. It follows that S_t has a log-normal distribution for any $t > 0$. For $m \in \mathbb{N}$, let $t_i = i\Delta t$ with $\Delta t = T/m$ denote the discrete observation times of the stock price S_t (e.g. daily at market closure). The payoff of an Asian call option is given by

$$\Psi_j(S_{t_0}, S_{t_1}, \ldots, S_{t_m}) = P_j\left(\frac{1}{m}\sum_{i=1}^{m} S_{t_i} - K\right)$$

where K denotes the strike price, for some given function P_j, $j = 1, 2$. Use Quasi-Monte Carlo to estimate the expected payoff $\mathbb{E}\big(\Psi_j(S_{t_0}, S_{t_1}, \ldots, S_{t_m})\big)$, using the process parameters $m = 256$, $r = 0.5$, $\sigma = 0.3$, $T = 2$, $S_0 = 5$, $K = 10$ for the following functions:

1. $$P_1(z) = z_+ = \max(z, 0)$$

2. $$P_2(z) = \frac{\log(1 + e^{\beta z})}{\beta}, \quad \beta > 0.$$

Notice that P_2 is a smooth approximation of P_1. Use the module `sobol_new.py` available on the course's website to generate Sobol sequences.[1] The `Python` [2] syntax `R = generate_points(N,d,0)`

[1] These functions were adapted from John Burkardt's website page at the Florida State University: `http://people.sc.fsu.edu/∼jburkardt/m_src/m_src.html`. There you can also find many other sequence generators.

[2] Download the files `sobol_new.py` and `Sobol_new-joe-kuo-6.21201` from the course website and use them by writing `from sobol_new import *` at the beginning of your python script. Both files should be in the same directory

generates a matrix `R` of size $N \times d$ corresponding to N vectors of dimension d for each payoff $P_j,\ j = 1, 2$. Estimate the QMC error using the CLT by estimating the variance with a *randomized* QMC. Plot both the estimated error based on random shifts as functions of the number of samples N, say, and estimate the convergence rate. Compare with the standard Monte Carlo error and explain the observed results for each payoff. Experiment with different values of β.

Consider now a Asian binary option with payoff given by

$$\tilde{\Psi}_j(S_{t_0}, S_{t_1}, \ldots, S_T) = \tilde{P}_j \left(\frac{1}{m} \sum_{i=1}^{m} S_{t_i} - K \right), \quad j = 1, 2,$$

with

1. $$\tilde{P}_1(z) = 20 \times \mathbf{1}_{z \geq 0},$$

2. $$\tilde{P}_2(z) = 20 \left(1 + \exp(-2\gamma z)\right)^{-1}, \quad \gamma > 0.$$

Here, $\mathbf{1}(\cdot)$ is the indicator function. as before, notice that $\tilde{P}_2$ is a smooth approximation of $\tilde{P}_1$. Repeat the previous experiments for different values of γ and comment on your results.