Pricing Geometric and Arithmetic Asian Options with Brownian Bridge and Stratified Sampling Method

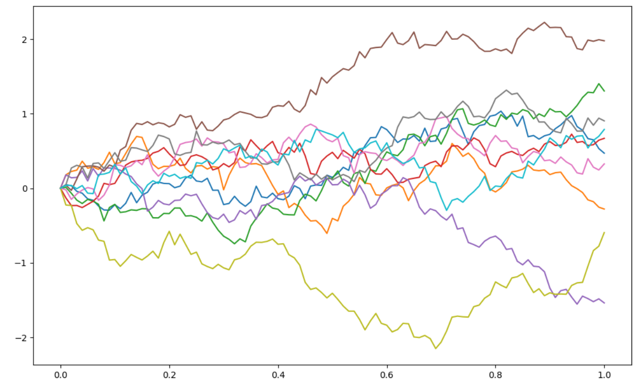
Hang Miao

1 Introduction

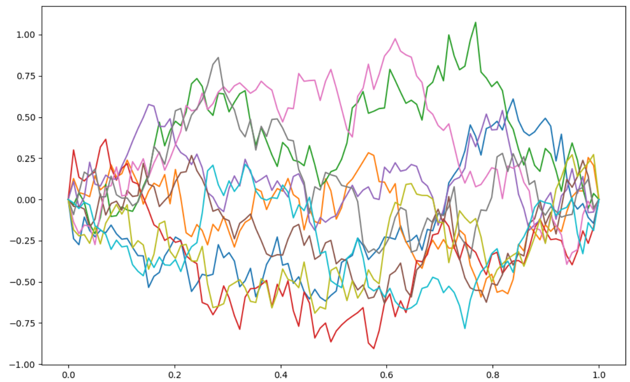
Asian option, named after its birthplace, was first introduce in 1987 to compute the option linked to the average price of crude oil. Due to its payoff depends on the average of price of the underlying asset instead of price itself, it has lower implied volatility and become harder to be manipulated than the equivalent vanilla option, therefore less expensive. Asian option is especially appealing to the investors or corporate treasury managers that need to hedge on the average revenue or cost from a stream of cash flow. For example, multinational corporations accrued stream of revenue in foreign currency could use Asian put to hedge against the exchange rate risk. Airlines could use Asian call to lock in the cost from raw materials like crude oil. Albeit other exotic options has lost their luster in the recent years, Asian option becomes more and more commonplace. For commodity like gold and oil, vast majority of options are Asian. In fact, Asian option is considered as vanilla in crude oil markets.

Based on average type, there are two types of Asian option, average by geometric mean and arithmetic mean. The closed form solution under Black-Scholes environment for geometric option is derived by Kemna and Vorst (1990) and Turnbull and Wakeman (1991). However, the analytical formula for pricing Asian options with arithmetic mean poses a great challenge mainly because the distribution of the arithmetic average asset price is usually not analytical available. Since majority of Asian option adopt arithmetic mean, one must rely on numerical and simulation methods to approximate the valuation. Simulation method include Kemna and Vorst (1990) using control variates, Boyle et al. (1997) summarized other variance reduction techniques. Numerical methods include Geman and Yor (1993) which derived an analytical expression for the Laplace transform of the Asian option price, making it possible to determine the option value by means of numerical (inversion) methods. Thompson (1999); Fusai and Kyriakou (2016) derived the upper and lower bounds for Asian option, the range is quit tight for direct application. While Monte Carlo simulation methods is far more robust and always converge to the true valuation with convergence rate $n^{-\frac{1}{2}}$, it also suffered from the high computational cost. This paper aims to improve the computational efficiency by using Monte Carlo with pseudo random sequence and quasi Monte Carlo with low discrepancy sequence combined with Brownian bridge and stratified sampling techniques.

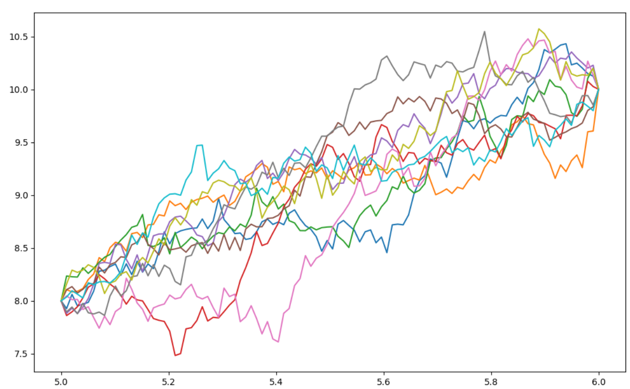
The Brownian bridge $X_{0\rightarrow T}^{0\rightarrow 0}(t)$ is a Brownian motion $W(t)$ subject to the condition that value at the end is pinned down to the same value at the start, i.e $W(T) = W(0) = 0$. From the visualization of the Brownian motion in Figure 1a compared with Brownian motion bridge in Figure 1b, Brownian bridge is tied down at both ends, so the process in between forms a bridge, albeit a very jagged one. The generalized Brownian Bridge process $X_{t_1 \to t_2}^{x_1 \to x_2}(t)$, showed in Figure 1c and 1d, is obtained from a standard Brownian motion process $W(t)$ restricted to a fixed time interval $[t_1, t_2]$, and conditioning on the events that $W(t_1) = x_1$ and $W(t_1) = x_2$. I extended the



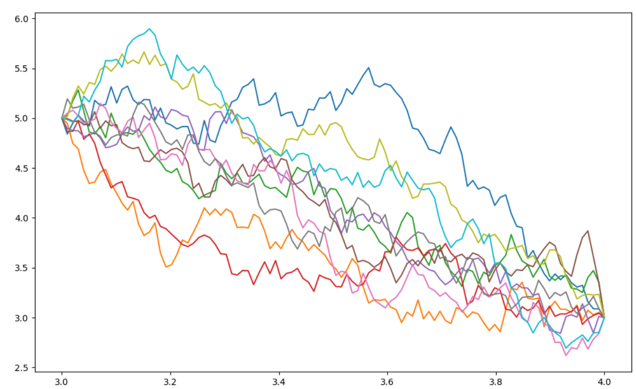
(a) Brownian Motion $W(t)$



(b) Brownian Bridge $X_{0 \to 1}^{0 \to 0}(t)$



(c) General Brownian Bridge $X_{5 \to 6}^{8 \to 10}(t)$

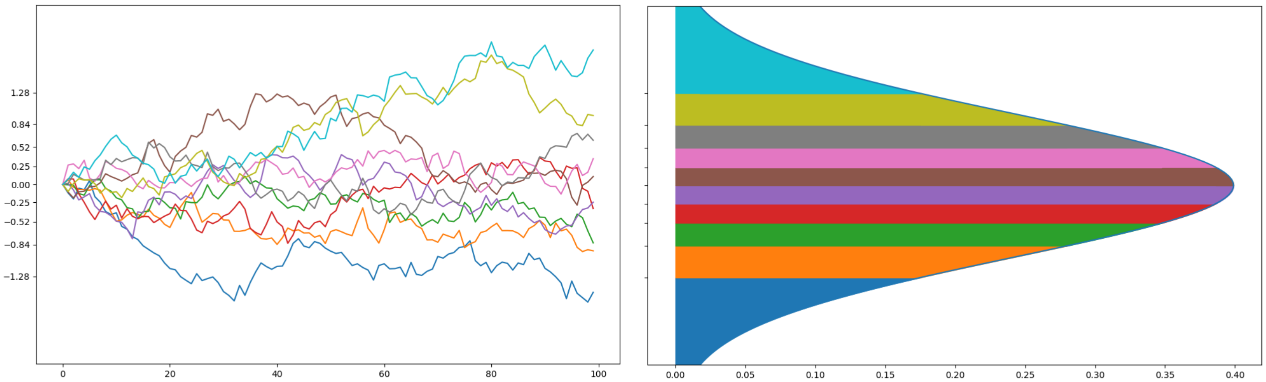


(d) General Brownian Bridge $X_{3 \to 4}^{5 \to 3}(t)$

Figure 1: Visualization of Brownian Motion, Brownian Bridge and Generalized Bronian Bridge

definition of Brownian bridge from the time domain $[0, T]$ with two end points 0 and 0 to the generalized one on any closed time interval $[t_1, t_2]$ with any fixed end points x_1, $x_2 \in \mathbb{R}$. The associated Theorems for the generalized Brownian bridge is formalized and proved. Based on the theorems, the algorithm for generating Brownian bridge is derived.

Stratified sampling is one of variance reduction techniques for improving the computational efficiency of Monte Carlo simulation. The general idea of stratified sampling is to partition the population space Ω into n disjoint subspaces Ω_i where $\Omega_i \cap \Omega_j = \emptyset$, $\cup_{i=1}^{n} \Omega_i = \Omega$ with known probability p_i. Suppose N samples are needed, instead of drawing N samples directly from population space Ω, Np_i samples are randomly drawing from each stratum Ω_i, then combine them together. Such disassemble and assemble procedure like the algorithm of divide and conquer, leads to the randomly drawing sample with lower variance in general. Stratified sampling method is frequently used together with Brownian bridge in option pricing. Since most of the payoff function is involved with only terminal value of the underlying asset process, the stratified sampling method could be applied to stratified the Brownian motion at the expiry of the underlying process, then use Brownian bridge to "fill in" the path in between. The visualization of the stratified sampling algorithm accompanied by Brownian bridge is showed in Figure 2.



(a) Brownian Bridge $X_{0\to 1}^{0\to x_i}(t)$ (b) Equiprobable Stratum Under $N(0,1)$

Figure 2: Brownian Motion Simulation using Stratified Sampling and Brownian Bridge. Given the ith stratum of $N(0,1)$, the terminal value x_i is randomly sampled. Then the entire path i could be generated using Brownian Bridge $X_{0\to 1}^{0\to x_i}(t)$.

The rest of paper is structured as follows: section 2 set up the Black-Scholes environment, where underlying asset process follows geometric Brownian motion; lay out the foundation for the general derivative pricing formula under risk neutral measure using noarbitrage argument; specify the payoff function for discretely monitored and continuously monitored Geometric Asian call option and discretely monitored arithmetic Asian call option at the end. Section 3 discussed the methods for pricing Geometric Asian option with the focus on deriving the analytical formula for discretely monitored and continuously monitored geometric Asian option. Section 4 is about methods for pricing arithmetic Asian option with a focus on Monte Carlo simulation and Quasi Monte Carlo simulation. Section 5 is the core part of this paper, where I discuss the Brownian bridge and stratified sampling method, I extended the Brownian bridge and proved the related theorem under the general Brownian bridge, provide two algorithms for generating Brownian bridge, the implementation of the algorithm for pricing Asian option at the end. Section 6 shows the testing result and evaluate different methods under different circumstances based on testing result.

2 Model Setting for Underlying Asset Process and Payoff Functions

2.1 Underlying Stock Price Process under Risk Neutral Measure

Suppose the underlying asset price follows Geometric Brownian with continuous dividend payment d.

$$\mathrm{d}S_t = (\alpha - d)S_t\mathrm{d}t \; + \; \sigma S_t \mathrm{d}W_t$$

The market money account risk free return is r. Let the market price of risk $\theta = \frac{\alpha - r}{\sigma}$, and apply Girsanov's Theorem to change the physical measure to risk neutral measure under which

$\mathrm{d}\widetilde{W}_t = \theta \mathrm{d}t + \mathrm{d}W_t$ is a Brownian Motion. The SDE for the underlying asset price process under risk neutral measure is

$$\mathrm{d}S_t = (r-d)S_t \mathrm{d}t \ + \ \sigma S_t \mathrm{d}\widetilde{W}_t$$

The discounted portfolio of holding stock and money account is a martingale under risk neutral measure. Given any derivatives which payoff is $V(T)$ where $V(T)$ is $\mathcal{F}_T$-measurable. Applying martingale representation Theorem, we can find an adapted process $\Gamma(t)$ such that the short position of this derivative can be hedged. Under the no arbitrage condition, the price of the derivative should be the same as the value of the portfolio hedge the short position. Therefore,

$$D_t V_t = D_t X_t = \widetilde{\mathbb{E}}[\, D_T X_T \,|\, \mathcal{F}_t] = \widetilde{\mathbb{E}}[\, D_T V_T \,|\, \mathcal{F}_t] \tag{1}$$

Since r is constant, $D_t = \mathrm{e}^{-rt}$ is a deterministic discount factor independent of $\mathcal{F}_t$.

2.2 Payoff Function for Geometric Asian Call Option

2.2.1 Discretely Monitored

The payoff function for arithmetic Asian call option is:

$$\begin{aligned} V_G(T) &= \left(\left(\prod_{k=1}^{D} S_{t_k} \right)^{\frac{1}{D}} - K \right)^+ \\ &= \left(\exp\left\{ \frac{1}{D} \sum_{k=1}^{D} \log S_{t_k} \right\} - K \right)^+ \end{aligned} \tag{2}$$

Where $0 = t_0 < t_1 < \cdots < t_d = T$ are discretely sampled dates

2.2.2 Continuously Monitored

The payoff function for arithmetic Asian call option is:

$$\begin{aligned} V_G(T) &= \left(\lim_{D\to\infty} \left(\prod_{k=1}^{D} S(t) \right)^{\frac{1}{D}} - K \right)^+ \\ &= \left(\exp\left\{ \frac{1}{T} \int_0^T \log S(t)\, \mathrm{d}t \right\} - K \right)^+ \end{aligned} \tag{3}$$

2.3 Payoff Function for Arithmetic Asian Call Option

The payoff function for arithmetic Asian call option is:

$$V_A(T) = \left(\frac{1}{D} \sum_{k=1}^{D} S_{t_k} - K \right)^+ \tag{4}$$

Where $0 = t_0 < t_1 < \cdots < t_d = T$ are discretely sampled dates

3 Pricing the Geometric Asian Call Option

3.1 Analytical Formula for Discretely Monitored Geometric Asian Option

Since the asset price SDE under risk neutral measure is:

$$\mathrm{d}S_t = (r-d)S_t \mathrm{d}t \;+\; \sigma S_t \mathrm{d}\widetilde{W}_t$$

which follows geometric Brownian Motion, we get the solution for the above SDE as:

$$S_t = S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)t+\sigma\widetilde{W}_t} = S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)t+\sigma\sqrt{t}\mathbf{Z}}, \quad \text{where } \mathbf{Z} \sim \text{Normal}(0,1)$$

The discretely observed underlying asset follows geometric Brownian motion as follows:

$$\begin{aligned}
S_{t_1} &= S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}\mathbf{Z_1}} \\
S_{t_2} &= S_{t_1} \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}\mathbf{Z_2}} = S_0 \mathrm{e}^{2(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}(\mathbf{Z_1}+\mathbf{Z_2})} \\
S_{t_3} &= S_{t_2} \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}\mathbf{Z_3}} = S_0 \mathrm{e}^{3(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}(\mathbf{Z_1}+\mathbf{Z_2}+\mathbf{Z}_3)} \\
&\cdots \\
S_{t_d} &= S_{t_{d-1}} \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}\mathbf{Z_d}} = S_0 \mathrm{e}^{D(r-d-\frac{1}{2}\sigma^2)h+\sigma\sqrt{h}\sum_{k=1}^{D}\mathbf{Z}_k}
\end{aligned}$$

Multiply them together and take the d^{th} root, we have the geometric mean

$$\begin{aligned}
\left(\prod_{k=1}^{D} S_{t_k}\right)^{\frac{1}{D}} &= \left(S_0^D \; \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\sum_{k=1}^{D} k} \; \mathrm{e}^{\sigma\sqrt{h}(D\mathbf{Z}_1+(D-1)\mathbf{Z}_2+(D-2)\mathbf{Z}_3+\cdots+\mathbf{Z}_D)}\right)^{\frac{1}{D}} \\
&= S_0 \; \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}} \; \mathrm{e}^{\frac{\sigma}{D}\sqrt{h}(D\mathbf{Z}_1+(D-1)\mathbf{Z}_2+(D-2)\mathbf{Z}_3+\cdots+\mathbf{Z}_D)}
\end{aligned}$$

The linear combination of finite random variables with normal distribution is still normal distribution.

$$(D\mathbf{Z}_1 + (D-1)\mathbf{Z}_2 + (D-2)\mathbf{Z}_3 + \cdots + \mathbf{Z}_D) \sim N(\mu, \sigma^2) \tag{5}$$

where

$$\begin{aligned}
\mu &= \widetilde{\mathbb{E}}\left[(D\mathbf{Z}_1 + (D-1)\mathbf{Z}_2 + (D-2)\mathbf{Z}_3 + \cdots + \mathbf{Z}_D)\right] \\
&= 0 \\
\sigma^2 &= \widetilde{\mathrm{Var}}\left[(D\mathbf{Z}_1 + (D-1)\mathbf{Z}_2 + (D-2)\mathbf{Z}_3 + \cdots + \mathbf{Z}_D)\right] \\
&= 1^2 + 2^2 + 3^2 + \cdots + D^2 \\
&= \frac{D(D+1)(2D+1)}{6}
\end{aligned}$$

Apply the risk neutral formula we have the

$$
\begin{aligned}
V_0 &= \widetilde{\mathbb{E}}\left[e^{-rT}\,V_G(T)\right] \\
&= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\left(\prod_{k=1}^{D} S_{t_k}\right)^{\frac{1}{D}} - K\right)^+\right] \\
&= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0\,\mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\;\mathrm{e}^{\frac{\sigma}{D}\sqrt{h}\mathbf{Z}_T} - K\right)^+\right] \quad \text{where } \mathbf{Z}_T \sim N\left(0, \frac{D(D+1)(2D+1)}{6}\right) \\
&= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0\,\mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\;\mathrm{e}^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}} - K\right)^+\right] \quad \text{where } \mathbf{Z} \sim N(0,1) \qquad (6)
\end{aligned}
$$

$$
\begin{aligned}
& S_0\,\mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\;\mathrm{e}^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}} - K \geq 0 \\
\iff & \mathbf{Z} \geq \left(\log\frac{K}{S_0} - (r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}\right)/\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}
\end{aligned}
$$

Let $l = \left(\log\frac{K}{S_0} - (r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}\right)/\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}$, the following is the continuation of equation (6)

$$
\begin{aligned}
V_0 &= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0\,\mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\;\mathrm{e}^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}} - K\right)^+\right] \\
&= e^{-rT}\frac{1}{\sqrt{2\pi}}\int_l^\infty \left(S_0\,\mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\;\mathrm{e}^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}} - K\right) e^{-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z} \\
&= e^{-rT}\left[\frac{S_0 e^{h\frac{1+D}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}}\int_l^\infty e^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z} - \frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[\frac{S_0 e^{h\frac{1+D}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}} e^{\sigma^2 h\frac{(D+1)(2D+1)}{12D}}\int_l^\infty e^{-\frac{\left(\mathbf{Z}-\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\right)^2}{2}}\,\mathrm{d}\mathbf{Z} - \frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[\frac{S_0 e^{h\frac{1+D}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}} e^{\sigma^2 h\frac{(D+1)(2D+1)}{12D}}\int_{l-\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}}^\infty e^{-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z} - \frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\,\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[S_0 e^{h\left(\frac{1+D}{2}(r-d-\frac{1}{2}\sigma^2)+\sigma^2\frac{(D+1)(2D+1)}{12D}\right)}\Phi\left(\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}} - l\right) - K\Phi(-l)\right] \qquad (7)
\end{aligned}
$$

plug into all the given parameters and use the "CumulativeNormal(double x)" function in the "Normals.h" to compute $N(\pm d_\pm)$ we have the option price calculated via explicit formula written in file "BSM_GeometricAsianFormula".

3.2 Analytical Formula for Continuously Monitored Geometric Asian Option

Since the asset price SDE under risk neutral measure is:

$$\mathrm{d}S_t = (r-d)S_t\mathrm{d}t \ + \ \sigma S_t \mathrm{d}\widetilde{W}_t$$

which follows geometric Brownian Motion, we get the solution for the above SDE as:

$$S_t = S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)t+\sigma\widetilde{W}_t} = S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)t+\sigma\sqrt{t}\mathbf{Z}}, \quad \text{where } \mathbf{Z} \sim \text{Normal}(0,1)$$

The payoff function for Geometric Asian in continuous case is:

$$\begin{aligned}
V_G(T) &= \left(\lim_{d\to\infty}\left(\prod_{k=1}^{D} S_{t_k}\right)^{\frac{1}{D}} - K\right)^+ \\
&= \left(\exp\left\{\frac{1}{T}\int_0^T \log S_t \,\mathrm{d}t\right\} - K\right)^+ \\
&= \left(\exp\left\{\frac{1}{T}\int_0^T \log S_0 \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)t+\sigma\widetilde{W}_t}\,\mathrm{d}t\right\} - K\right)^+ \\
&= \left(S_0\exp\left\{\frac{1}{T}\int_0^T (r-d-\frac{1}{2}\sigma^2)t\,\mathrm{d}t\right\}\exp\left\{\frac{\sigma}{T}\int_0^T \widetilde{W}_t\,\mathrm{d}t\right\} - K\right)^+ \\
&= \left(S_0\exp\left\{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right\}\exp\left\{\frac{\sigma}{T}\int_0^T \widetilde{W}_t\,\mathrm{d}t\right\} - K\right)^+ \\
&= \left(S_0\exp\left\{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right\}\exp\left\{\frac{\sigma}{T}\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right\} - K\right)^+ \qquad (8)
\end{aligned}$$

The last equation follows the Ito's lemma,

$$\begin{aligned}
\mathrm{d}(t\widetilde{W}_t) &= \widetilde{W}_t\mathrm{d}t + t\mathrm{d}\widetilde{W}_t \\
\int_0^T \mathrm{d}(t\widetilde{W}_t) &= \int_0^T \widetilde{W}_t\mathrm{d}t + \int_0^T t\mathrm{d}\widetilde{W}_t \\
\Longrightarrow \int_0^T \widetilde{W}_t\mathrm{d}t &= T\widetilde{W}_T - \int_0^T t\mathrm{d}\widetilde{W}_t \\
&= T\int_0^T \mathrm{d}\widetilde{W}_t - \int_0^T t\mathrm{d}\widetilde{W}_t \\
&= \int_0^T (T-t)\mathrm{d}\widetilde{W}_t \qquad (9)
\end{aligned}$$

By the definition of Ito's integral since the integrand $(T-t)$ is a deterministic process we have

$$\widetilde{\mathbb{E}}\left[\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right] = 0 \qquad (10)$$

By Ito Isometry we have

$$
\begin{aligned}
\widetilde{\mathrm{Var}}\left[\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right] &= \widetilde{\mathbb{E}}\left[\left(\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right)^2\right] - \left(\widetilde{\mathbb{E}}\left[\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right]\right)^2 \\
&= \widetilde{\mathbb{E}}\left[\int_0^T (T-t)^2\mathrm{d}t\right] \\
&= \frac{T^3}{3} \qquad (11)
\end{aligned}
$$

From the definition of Ito Integral, see Oksendal (2005) p.28. Let $\Pi = \{t_k\}_{k=0}^n$ be a partition of $[0,T]$ with $0 = t_0 < t_1 < t_2 < \cdots < t_k < \cdots < t_n = T$ as $n \to \infty$, we got more and more refined partition, the discretized sum would converge to the Ito integral in L^2

$$
\int_0^T (T-t)\mathrm{d}\widetilde{W}_t = \lim_{n\to\infty}\sum_{k=0}^{n-1}(T-t_k)(\widetilde{W}_{t_{k+1}} - \widetilde{W}_{t_k}) \qquad (12)
$$

Note that the right side of eq. (12) is a Cauhcy sequence in L^2. $\forall\epsilon$, $\exists N$ such that $|(t_i - t_{i'})| < \epsilon$ where $t_{i'} = t_j$ if $t_j \leq t_i < t_{j+1}$ whenever $n > m \geq N$

$$
\begin{aligned}
&\left\|\sum_{i=0}^{n-1}(T-t_i)(\widetilde{W}_{t_{i+1}} - \widetilde{W}_{t_i}) - \sum_{j=0}^{m-1}(T-t_j)(\widetilde{W}_{t_{j+1}} - \widetilde{W}_{t_j})\right\| \\
&= \left\|\sum_{i=0}^{n-1}(t_i - t_{i'})(\widetilde{W}_{t_{i+1}} - \widetilde{W}_{t_i})\right\| \\
&= \widetilde{\mathbb{E}}\left[\sum_{i=0}^{n-1}(t_i - t_{i'})^2(\widetilde{W}_{t_{i+1}} - \widetilde{W}_{t_i})^2\right] \\
&= \sum_{i=0}^{n-1}(t_i - t_{i'})^2(t_{i+1} - t_i) \\
&\leq \epsilon^2 T
\end{aligned}
$$

Since L^2 is complete, the Ito integral is well-defined.
The law of the partial sum on the right side of equation eq. (12) follows Normal distribution. Then its L^2 limit also follows Normal distribution. see Le Gall (2016) p.3.

$$
\lim_{n\to\infty}\sum_{k=0}^{n-1}(T-t_k)(\widetilde{W}_{t_{k+1}} - \widetilde{W}_{t_k}) \xrightarrow{L} N(\mu, \sigma^2) \qquad (13)
$$

Combined the results eqs. (10) to (13), we have

$$
\int_0^T (T-t)\mathrm{d}\widetilde{W}_t \;\sim\; N(0, \frac{T^3}{3}) \qquad (14)
$$

Apply the risk neutral formula we have the

$$
\begin{aligned}
V_0 &= \widetilde{\mathbb{E}}\left[e^{-rT}\,V_G(T)\right] \\
&= \widetilde{\mathbb{E}}\left[e^{-rT}\left(S_0\exp\left\{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right\}\exp\left\{\frac{\sigma}{T}\int_0^T (T-t)\mathrm{d}\widetilde{W}_t\right\}-K\right)^+\right] \\
&= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}e^{\mathbf{Z}_t}-K\right)^+\right] \qquad \text{where } \mathbf{Z}_T \sim N\left(0,\frac{\sigma^2 T}{3}\right) \\
&= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}}-K\right)^+\right] \qquad \text{where } \mathbf{Z} \sim N\left(0,1\right)
\end{aligned} \tag{15}
$$

$$
S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}}-K \geq 0 \iff \mathbf{Z} \geq \left(\log\frac{K}{S_0}-\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}}
$$

Let $l=\left(\log\frac{K}{S_0}-\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}}$, the following is the continuation of equation (12)

$$
\begin{aligned}
V_0 &= e^{-rT}\,\widetilde{\mathbb{E}}\left[\left(S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}}-K\right)^+\right] \qquad \text{where } \mathbf{Z} \sim N\left(0,1\right) \\
&= e^{-rT}\frac{1}{\sqrt{2\pi}}\int_l^\infty \left(S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}}-K\right)e^{-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z} \\
&= e^{-rT}\left[\frac{S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}}\int_l^\infty e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z}-\frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[\frac{S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}}e^{\frac{\sigma^2 T}{6}}\int_l^\infty e^{-\frac{\left(\mathbf{Z}-\sigma\sqrt{\frac{T}{3}}\right)^2}{2}}\mathrm{d}\mathbf{Z}-\frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[\frac{S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}}{\sqrt{2\pi}}e^{\frac{\sigma^2 T}{6}}\int_{l-\sigma\sqrt{\frac{T}{3}}}^\infty e^{-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z}-\frac{K}{\sqrt{2\pi}}\int_l^\infty e^{-\frac{\mathbf{Z}^2}{2}}\mathrm{d}\mathbf{Z}\right] \\
&= e^{-rT}\left[S_0 e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)+\frac{\sigma^2 T}{6}}\Phi\left(\sigma\sqrt{\frac{T}{3}}-l\right)-K\Phi\left(-l\right)\right]
\end{aligned} \tag{16}
$$

$$
\begin{aligned}
-l &= -\left(\log\frac{K}{S_0}-\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}} \\
&= \left(\log\frac{S_0}{K}+\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}} \\
\sigma\sqrt{\frac{T}{3}}-l &= \sigma\sqrt{\frac{T}{3}}+\left(\log\frac{S_0}{K}+\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}} \\
&= \left(\log\frac{S_0}{K}+\frac{T}{2}(r-d+\frac{1}{6}\sigma^2)\right)/\sigma\sqrt{\frac{T}{3}}
\end{aligned}
$$

plug into all the given parameters and use the "CumulativeNormal(double x)" function in the "Normals.h" to compute $N(\pm d_\pm)$ we have the option price calculated via explicit formula written in file "BSM_GeometricAsianFormula".

3.3 Monte Carlo Methods for Discretely Monitored Geometric Asian Option

The Monte Carlo simulation method could be used to compute the discretely monitored geometric Asian option by approximate the integral of the discounted expected payoff. Due to strong law of large number, the average of a sequence of i.i.d random variable converge almost surely to the expected value. From equation (6), the discretely monitored geometric Asian option could be simplified as follows:

$$
\begin{aligned}
V_0 &= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\left(\prod_{k=1}^{D} S_{t_k}\right)^{\frac{1}{D}} - K\right)^{+}\right] && (17)\\
&= \widetilde{\mathbb{E}}\left[e^{-rT}\left(S_0\, \mathrm{e}^{(r-d-\frac{1}{2}\sigma^2)h\frac{1+D}{2}}\, \mathrm{e}^{\sigma\sqrt{h\frac{(D+1)(2D+1)}{6D}}\mathbf{Z}} - K\right)^{+}\right] && (18)
\end{aligned}
$$

The one step Monte Carlo method could be implemented by simply simulate a sequence of $\{\boldsymbol{Z}^{(i)}\}_{i=1}^{N}$, where $\mathbf{Z}^{(i)} \sim \text{Normal}(0,1)$, with large N and compute the sample mean for the discounted payoff in eq. (18).

Since Asian option has path dependent payoffs, like pricing all the other path dependent options, we could apply discretization methods to the underlying SDE, simulate the entire path and calculate the original discounted payoff in eq. (17) directly. For example, we could apply Euler Scheme to the underlying spot process $S(t)$ as follows:

$$S_{t_{k+1}}^{(i)} = S_{t_k}^{(i)} + (r-d)S_{t_k}^{(i)}h + \sigma S_{t_k}^{(i)}\sqrt{h}\mathbf{Z}_{t_k}^{(i)},\ k = 0,\cdots 11.$$

Or we could apply Milstein scheme to the underlying spot process $S(t)$ as follows:

$$S_{t_{k+1}}^{(i)} = S_{t_k}^{(i)} + (r-d)S_{t_k}^{(i)}h + \sigma S_{t_k}^{(i)}\sqrt{h}\mathbf{Z}_{t_k}^{(i)} + \frac{1}{2}\sigma^2 S_{t_k}^{(i)}h[\mathbf{Z}_{t_k}^{(i)2} - 1],\ k = 0,\cdots 11.$$

In addition, we could write the original risk neutral formula for discretely monitored geometric Asian option in terms of log spot as follows:

$$
\begin{aligned}
V_0 &= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\left(\prod_{k=1}^{D} S_{t_k}\right)^{\frac{1}{D}} - K\right)^{+}\right]\\
&= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\exp\left\{\frac{1}{D}\sum_{k=1}^{D}\log S_{t_k}\right\} - K\right)^{+}\right] && (19)
\end{aligned}
$$

Applying the Euler scheme or Milstein scheme (coincide with each other for log geometric Brownian motion) to the SDE of log spot, we get

$$\log S_{t_{k+1}}^{(i)} = \log S_{t_k}^{(i)} + (r-d-\frac{1}{2}\sigma^2)h + \sigma\sqrt{h}\mathbf{Z}_{t_k}^{(i)},\ k = 0,\cdots 11$$

where $\log S_{t_0} = \log(100)$, $h = \frac{1}{12}$, $\mathbf{Z}_{t_k}^{(i)} \sim \text{Normal}(0,1)$. Then the payoff for each path could be computed by exponentiate the arithmetic mean of the log spot at the monitored dates, compared with strike and discounted by the risk free rate. The risk neutral formula could then be approximated by the sample mean payoffs of each simulated path. For each subinterval and discretization scheme, $\mathbf{Z}_{t_k}^{(i)}$ is generated by Park Miller linear congruential random generator implemented by "random2" and "ParkMiller". This Monte Carlo option pricing algorithm is implemented in the files "ExoticBSEngine" and "PathDependentAsian". I write a statistc getherer in "MCSamples" in order to record the simulation result and compute the standard error(i.e standard deviation for sample mean).

3.4 Monte Carlo Methods for Continuously Monitored Geometric Asian Option

The Monte Carlo simulation method could be used to compute the continuously monitored geometric Asian option by approximate the integral of the discounted expected payoff. Due to strong law of large number, the average of a sequence of i.i.d random variable converge almost surely to the expected value. From equation (15), the continuously monitored geometric Asian option could be simplified as follows:

$$V_0 = \widetilde{\mathbb{E}}\left[e^{-rT}\left(\exp\left\{\frac{1}{T}\int_0^T \log S_t \,\mathrm{d}t\right\} - K\right)^+\right] \tag{20}$$

$$= \widetilde{\mathbb{E}}\left[e^{-rT}\left(S_0\, e^{\frac{T}{2}(r-d-\frac{1}{2}\sigma^2)}\, e^{\sigma\sqrt{\frac{T}{3}}\mathbf{Z}} - K\right)^+\right] \tag{21}$$

The one step Monte Carlo method could be implemented by simply simulate a sequence of $\{\boldsymbol{Z}^{(i)}\}_{i=1}^N$, where $\mathbf{Z}^{(i)} \sim \text{Normal}(0,1)$, with large N and compute the sample mean for the discounted payoff in eq. (21). Since there is no practical use of continuously monitored geometric Asian option, plus the closed form solution is already derived in (16), the Monte Carlo implementation is omitted. The implementation of the closed form formula is in the files with name "BSM_GeometricAsianFormula".

3.5 Quasi Monte Carlo Simulation with Sobol Low Discrepancy Sequence

The concept that low-discrepancy sequences provide an efficient method of "filling" the hypercube $[0,1]^d$ with a smaller number points for quasi-Monte Carlo integration than might be required by traditional numerical integration for the same accuracy.
In contrast with the regular Monte Carlo method, in which, $\mathbf{Z}^{(i)}$ is obtained by first generating a uniform random number using Park Miller method, then convert that uniform random number to $\mathbf{Z}^{(i)}$ by the standard normal inverse CDF function, Quasi Monte Carlo produce the $\mathbf{Z}^{(i)}$ by first obtain instead of uniform random number but a 1-dimensional low discrepancy number and then convert this number to $\mathbf{Z}^{(i)}$ by the same standard normal inverse CDF function. A quasirandom or low discrepancy sequence, such as the Faure, Halton, Hammersley, Niederreiter or Sobol sequences, is "less random" than a pseudorandom number sequence, but more useful for such tasks

as approximation of integrals in higher dimensions, and in global optimization. This is because low discrepancy sequences tend to sample space "more uniformly" than random numbers. Algorithms that use such sequences may have superior convergence.
Preponderance of the experimental evidence amassed to date suggest that Sobol sequences are in many aspects superior to other low discrepancy sequences in financial application. Therefore, I use Sobol sequence to replace the previous pseudo random sequence produced by Park Miller. Considering there are a lot of Sobol generators works well and effective but its interface is not what the rest of code expect, I use the adapter design pattern to create an adapter class that fits the interface and having the existing effective Sobol generator as inner object. The adapter class is implemented in "Random_Sobol". The inner object of Sobol sequence is implemented in "sobol".

4 Pricing the Arithmetic Asian Call Option

4.1 Monte Carlo Simulation

There is no closed form solution for Arithmetic Asian Option. The Monte Carlo simulation method could be used to compute the discretely monitored arithmetic Asian option by approximate the integral of the discounted expected payoff. Due to strong law of large number, the average of a sequence of i.i.d random variable converge almost surely to the expected value. Plug in the payoff function of equation (4) into the risk neutral formula of equation (1), the arithmetic Asian option could be derived as follows:

$$V_0 = \widetilde{\mathbb{E}}\left[e^{-rT}\left(\frac{1}{D}\sum_{k=1}^{D} S_{t_k} - K\right)^+\right] \tag{22}$$

Since Asian option has path dependent payoffs, like pricing all the other path dependent options, we could apply discretization methods to the underlying SDE, simulate the entire path and calculate the discounted payoff in eq. (22) directly. For example, we could apply Euler Scheme to the underlying spot process $S(t)$ as follows:

$$S_{t_{k+1}}^{(i)} = S_{t_k}^{(i)} + (r-d)S_{t_k}^{(i)}h + \sigma S_{t_k}^{(i)}\sqrt{h}\mathbf{Z}_{t_k}^{(i)},\ k = 0, \cdots 11.$$

Or we could apply Milstein scheme to the underlying spot process $S(t)$ as follows:

$$S_{t_{k+1}}^{(i)} = S_{t_k}^{(i)} + (r-d)S_{t_k}^{(i)}h + \sigma S_{t_k}^{(i)}\sqrt{h}\mathbf{Z}_{t_k}^{(i)} + \frac{1}{2}\sigma^2 S_{t_k}^{(i)}h[\mathbf{Z}_{t_k}^{(i)2} - 1],\ k = 0, \cdots 11.$$

In addition, we could write the risk neutral formula in eq. (22) arithmetic Asian option in terms of log spot as follows:

$$\begin{aligned} V_0 &= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\frac{1}{D}\sum_{k=1}^{D} S_{t_k} - K\right)^+\right] \\ &= \widetilde{\mathbb{E}}\left[e^{-rT}\left(\frac{1}{D}\sum_{k=1}^{D} \exp\{\log S_{t_k}\} - K\right)^+\right] \end{aligned} \tag{23}$$

Applying the Euler scheme or Milstein scheme (coincide with each other for log geometric Brownian motion) to the SDE of log spot, we get

$$\log S^{(i)}_{t_{k+1}} = \log S^{(i)}_{t_k} + (r - d - \frac{1}{2}\sigma^2)h + \sigma\sqrt{h}\mathbf{Z}^{(i)}_{t_k}, \ k = 0, \cdots 11$$

where $\log S_{t_0} = \log(100)$, $h = \frac{1}{12}$, $\mathbf{Z}^{(i)}_{t_k} \sim \text{Normal}(0,1)$. Then the payoff for each path could be computed by exponentiate the arithmetic mean of the log spot at the monitored dates, compared with strike and discounted by the risk free rate. The risk neutral formula could then be approximated by the sample mean payoffs of each simulated path. For each subinterval and discretization scheme, $\mathbf{Z}^{(i)}_{t_k}$ is generated by Park Miller linear congruential random generator implemented by "random2" and "ParkMiller". This Monte Carlo option pricing algorithm is implemented in the files "ExoticBSEngine" and "PathDependentAsian". I write a statistc getherer in "MCSamples" in order to record the simulation result and compute the standard error(i.e standard deviation for sample mean).

4.2 Quasi Monte Carlo Simulation with Sobol Sequence

The concept that low-discrepancy sequences provide an efficient method of "filling" the hypercube $[0,1]^d$ with a smaller number points for quasi-Monte Carlo integration than might be required by traditional numerical integration for the same accuracy.
In contrast with the regular Monte Carlo method, in which, $\mathbf{Z}^{(i)}$ is obtained by first generating a uniform random number using Park Miller method, then convert that uniform random number to $\mathbf{Z}^{(i)}$ by the standard normal inverse CDF function, Quasi Monte Carlo produce the $\mathbf{Z}^{(i)}$ by first obtain instead of uniform random number but a 1-dimensional low discrepancy number and then convert this number to $\mathbf{Z}^{(i)}$ by the same standard normal inverse CDF function. A quasirandom or low discrepancy sequence, such as the Faure, Halton, Hammersley, Niederreiter or Sobol sequences, is "less random" than a pseudorandom number sequence, but more useful for such tasks as approximation of integrals in higher dimensions, and in global optimization. This is because low discrepancy sequences tend to sample space "more uniformly" than random numbers. Algorithms that use such sequences may have superior convergence.
Preponderance of the experimental evidence amassed to date suggest that Sobol sequences are in many aspects superior to other low discrepancy sequences in financial application. Therefore, I use Sobol sequence to replace the previous pseudo random sequence produced by Park Miller. Considering there are a lot of Sobol generators works well and effective but its interface is not what the rest of code expect, I use the adapter design pattern to create an adapter class that fits the interface and having the existing effective Sobol generator as inner object. The adapter class is implemented in "Random_Sobol". The inner object of Sobol sequence is implemented in "sobol".

5 Brownian Bridge and Stratified Sampling

5.1 Brownian Bridge

In this section, I extended the definition of Brownian bridge from the time domain $[0,T]$ with two end points 0 and 0 to the generalized one on any closed time interval $[t_1, t_2]$ with any fixed end points x_1, $x_2 \in \mathbb{R}$. The associated Theorems for the generalized Brownian bridge is formalized and proved. Based on the theorems, the algorithm for generating Brownian bridge is derived.

Definition 5.1 (Brownian Bridge)**.** Let $W(t)$ be a Brownian motion on $\Big(\Omega, \mathscr{F}, \mathbb{P}, \{\mathscr{F}(t)\}_{t\geq 0}\Big)$. Fix $0 \leq t_1 < t_2 < \infty$, $x_1, x_2 \in \mathbb{R}$, then the Brownian bridge $X_{t_1 \to t_2}^{x_1 \to x_2}(t)$ from $X_{t_1 \to t_2}^{x_1 \to x_2}(t_1) = x_1$ to $X_{t_1 \to t_2}^{x_1 \to x_2}(t_2) = x_2$ on $[t_1, t_2]$ is defined as

$$X_{t_1 \to t_2}^{x_1 \to x_2}(t) = x_1 + \frac{(x_2 - x_1)(t - t_1)}{t_2 - t_1} + W(t) - W(t_1) - \frac{t - t_1}{t_2 - t_1}\big(W(t_2) - W(t_1)\big) \tag{24}$$

Definition 5.2 (Brownian Bridge as Scaled Stochastic Integral)**.** Let $W(t)$ be a Brownian motion on $\Big(\Omega, \mathscr{F}, \mathbb{P}, \{\mathscr{F}(t)\}_{t\geq 0}\Big)$. Fix $0 \leq t_1 < t_2 < \infty$, $x_1, x_2 \in \mathbb{R}$, then the Brownian bridge $X_{t_1 \to t_2}^{x_1 \to x_2}(t)$ from $X_{t_1 \to t_2}^{x_1 \to x_2}(t_1) = x_1$ to $X_{t_1 \to t_2}^{x_1 \to x_2}(t_2) = x_2$ on $[t_1, t_2]$ is defined as

$$X_{t_1 \to t_2}^{x_1 \to x_2}(t) = x_1 + \frac{(x_2 - x_1)(t - t_1)}{t_2 - t_1} + (t_2 - t)\int_{t_1}^{t} \frac{1}{t_2 - u} \mathrm{d}W(u) \tag{25}$$

Theorem 5.1 (Distribution of Brownian Bridge)**.** *Let $W(t)$ be a Brownian motion on $\Big(\Omega, \mathscr{F}, \mathbb{P}, \{\mathscr{F}(t)\}_{t\geq 0}\Big)$. Fix $0 \leq t_1 < t_2 < \infty$, $x_1, x_2 \in \mathbb{R}$, then the Brownian bridge $X_{t_1 \to t_2}^{x_1 \to x_2}(t)$ is a Gaussian process with mean $\frac{x_1(t_2-t)+x_2(t-t_1)}{t_2-t_1}$, variance $\frac{(t_2-t)(t-t_1)}{t_2-t_1}$ and covariance $s \wedge t - t_1 - \frac{(s-t_1)(t-t_1)}{t_2-t_1}$.*

Proof. From the definition of Brownian bridge in (24)

$$\begin{aligned}
X_{t_1 \to t_2}^{x_1 \to x_2}(t) &= x_1 + \frac{(x_2 - x_1)(t - t_1)}{t_2 - t_1} + W(t) - W(t_1) - \frac{t - t_1}{t_2 - t_1}\big(W(t_2) - W(t_1)\big) \\
&= \frac{x_1(t_2 - t) + x_2(t - t_1)}{t_2 - t_1} + \frac{(t_2 - t)\big(W(t) - W(t_1)\big) - (t - t_1)\big(W(t_2) - W(t)\big)}{t_2 - t_1} \\
&\overset{L}{=} \frac{x_1(t_2 - t) + x_2(t - t_1)}{t_2 - t_1} + \frac{(t_2 - t)\sqrt{t - t_1}Z_1 - (t - t_1)\sqrt{t_2 - t}Z_2}{t_2 - t_1} \\
&\overset{L}{=} \frac{x_1(t_2 - t) + x_2(t - t_1)}{t_2 - t_1} + \frac{\sqrt{(t_2 - t)(t - t_1)(t_2 - t_1)}}{t_2 - t_1}Z_3 \\
&\overset{L}{=} \frac{x_1(t_2 - t) + x_2(t - t_1)}{t_2 - t_1} + \sqrt{\frac{(t_2 - t)(t - t_1)}{t_2 - t_1}}Z_3
\end{aligned} \tag{26}$$

where Z_1, Z_2, Z_3 are independent random variables with standard normal distribution. From eq. (26), we could conclude the distribution of $X^{a \to b}(t)$ follows Gaussion with mean $a + \frac{(b-a)t}{T}$ and variance $\frac{t(T-t)}{T}$.

$$
\begin{aligned}
&\mathrm{Cov}[X_{t_1\to t_2}^{x_1\to x_2}(t),\ X_{t_1\to t_2}^{x_1\to x_2}(s)]\\
&= \mathbb{E}\left[\Big(W(t)-W(t_1)-\frac{t-t_1}{t_2-t_1}\big(W(t_2)-W(t_1)\big)\Big)\Big(W(s)-W(t_1)-\frac{s-t_1}{t_2-t_1}\big(W(t_2)-W(t_1)\big)\Big)\right]\\
&= \mathbb{E}\left[\big(W(t)-W(t_1)\big)\big(W(s)-W(t_1)\big)\right]-\frac{s-t_1}{t_2-t_1}\mathbb{E}\left[\big(W(t)-W(t_1)\big)\big(W(t_2)-W(t_1)\big)\right]\\
&\quad -\frac{t-t_1}{t_2-t_1}\mathbb{E}\left[\big(W(t_2)-W(t_1)\big)\big(W(s)-W(t_1)\big)\right]+\frac{(s-t_1)(t-t_1)}{(t_2-t_1)^2}\mathbb{E}\left[\big(W(t_2)-W(t_1)\big)^2\right]\\
&= s\wedge t-t_1-\frac{(s-t_1)(t-t_1)}{t_2-t_1}-\frac{(s-t_1)(t-t_1)}{t_2-t_1}+\frac{(s-t_1)(t-t_1)}{t_2-t_1}\\
&= s\wedge t-t_1-\frac{(s-t_1)(t-t_1)}{t_2-t_1}
\end{aligned}
$$

□

Corollary 5.1 (Distribution of the Generalized Brownian Bridge)**.** *Let $W(t)$ be a path of Brownian motion. Suppose the value of the Brownian motion $W(t)$ at times $t_1 < t_2 < \cdots < t_k$ is determined by $W(t_1) = x_1$, $W(t_2) = x_2$, $\cdots$, $W(t_k) = x_k$, given $s \in [t_i, t_{i+1}]$, the conditional distribution of $\Big(W(s)|W(t_1) = x_1, W(t_2) = x_2, \cdots, W(t_k) = x_k\Big) = \Big(W(s)|W(t_i = x_i, W(t_{i+1} = x_{i+1}\Big)$ by Markov property of Brownian motion, and*

$$\Big(W(s)|W(t_i = x_i, W(t_{i+1} = x_{i+1}\Big) \overset{L}{=} N\Big(\frac{(t_{i+1}-s)x_i+(s-t_i)x_{i+1}}{(t_{i+1}-t_i)}, \frac{(t_{i+1}-s)(s-t_i)}{(t_{i+1}-t_i)}\Big) \tag{27}$$

Theorem 5.2 (Stochastic Differential Equation of Brownian Bridge)**.** *Let $W(t)$ be a Brownian motion on $\Big(\Omega, \mathscr{F}, \mathbb{P}, \{\mathscr{F}(t)\}_{t\geq 0}\Big)$. Fix $0 \leq t_1 < t_2 < \infty$, $x_1, x_2 \in \mathbb{R}$, then the stochastic differential equation of Brownian bridge $X_{t_1\to t_2}^{x_1\to x_2}(t)$ is definied as*

$$\mathrm{d}X_{t_1\to t_2}^{x_1\to x_2}(t) = \frac{x_2}{t_2-t}\mathrm{d}t - \frac{X_{t_1\to t_2}^{x_1\to x_2}(t)}{t_2-t}\mathrm{d}t + \mathrm{d}W(t) \tag{28}$$

Proof. Applying the Ito's formula to the definition 2 of Brownian bridge in equation (25), we got

$$
\begin{aligned}
X_{t_1\to t_2}^{x_1\to x_2}(t) &= x_1+\frac{(x_2-x_1)(t-t_1)}{t_2-t_1}+(t_2-t)\int_{t_1}^{t}\frac{1}{t_2-u}\mathrm{d}W(u)\\
\mathrm{d}X_{t_1\to t_2}^{x_1\to x_2}(t) &= \frac{x_2-x_1}{t_2-t_1}\mathrm{d}t-\int_{t_1}^{t}\frac{1}{t_2-u}\mathrm{d}W(u)\cdot\mathrm{d}t+\frac{(t_2-t)}{t_2-t}\mathrm{d}W(t)\\
&= \frac{x_2-x_1}{t_2-t_1}\mathrm{d}t-\frac{X_{t_1\to t_2}^{x_1\to x_2}(t)-x_1-\frac{(x_2-x_1)(t-t_1)}{t_2-t_1}}{t_2-t}\mathrm{d}t+\mathrm{d}W(t)\\
&= \frac{(x_2-x_1)(t_2-t)+x_1(t_2-t)+x_2(t-t_1)}{(t_2-t_1)(t_2-t)}\mathrm{d}t-\frac{X_{t_1\to t_2}^{x_1\to x_2}(t)}{t_2-t}\mathrm{d}t+\mathrm{d}W(t)\\
&= \frac{x_2}{(t_2-t_1)}-\frac{X_{t_1\to t_2}^{x_1\to x_2}(t)}{t_2-t}\mathrm{d}t+\mathrm{d}W(t)
\end{aligned}
$$

$\square$

5.2 Algorithm for Brownian Bridge Generation

The algorithm for Brownian bridge generation could be easily derived from corollary 5.1 and theorem 5.2. Suppose the starting time t_0, the ending time $t_n = T$ and their associated starting value x_0 and terminal value x_T is given. The Brownian bridge $X_{t_0 \to t_n}^{x_0 \to x_T}(t)$ at $t = t_i$ where $t_0 < t_1 < t_2 < \cdots < t_{n-1} < t_n = T$ could be exactly computed by the algorithm in section 5.2.1 and could be approximated by algorithm in section 5.2.2

5.2.1 Exact Brownian Bridge Generation Algorithm

By corollary 5.1, $X_{t_0 \to t_n}^{x_0 \to x_T}(t)$ where $t_i = t_1, t_2, \cdots, t_n$ could be computed iteratively conditioned on the preceding value x_i and the terminal value x_T. The algorithm is illustrated as follow,

$$\begin{aligned} x_{i+1} &= X_{t_i \to t_n}^{x_{x_i} \to x_T}(t_{i+1}) \\ &= \frac{(t_n - t_{i+1})x_i + (t_{i+1} - t_i)x_T}{(t_n - t_i)} + \sqrt{\frac{(t_n - t_{i+1})(t_{i+1} - t_i)}{(t_n - t_i)}}\boldsymbol{Z}_i \end{aligned} \tag{29}$$

where $\{\boldsymbol{Z}_i\}_{i=1}^n$ are i.id follows standard normal distribution. I implemented this algorithm in the file "Random_Brownian_Bridge.hpp" and "Random_Brownian_Bridge.cpp".

5.2.2 Approximated Brownian Bridge Generation Algorithm

According to theorem 5.2, the Brownian bridge $X_{t_0 \to t_n}^{x_0 \to x_T}(t)$ where $t_i = t_1, t_2, \cdots, t_n$ could also be approximated iteratively given the preceding value x_i and the terminal value x_T. The algorithm is illustrated as follow,

$$x_{i+1} = x_i + \frac{x_T}{t_n - t_i}(t_{i+1} - t_i) - \frac{x_i}{t_n - t_i}(t_{i+1} - t_i) + \sqrt{(t_{i+1} - t_i)}\boldsymbol{Z}_i$$

where $\{\boldsymbol{Z}_i\}_{i=1}^n$ are i.id follows standard normal distribution. I implemented this algorithm in the file "Brownian_Bridge.py".

5.3 Algorithm for Brownian Bridge with Stratified Sampling

The general idea of stratified sampling is to partition the population space Ω into n disjoint subspaces Ω_i where $\Omega_i \cap \Omega_j = \emptyset$, $\cup_{i=1}^n \Omega_i = \Omega$ with known probability p_i. Suppose N samples are needed, instead of drawing N samples directly from population space Ω, Np_i samples are randomly drawing from each stratum Ω_i, then combine them together. For simplicity, denote the number of stratum as M by which the sample size N is divisible, we use proportional stratified sampling where each stratum Ω_i has the same probability $p_i = 1/M$.

The inverse sampling technique is often accompanied by stratified sampling. Instead of stratify the population space Ω into equiprobable stratum Ω_i, we stratified the probability space into equiprobable stratum $[0, \frac{1}{M}], [\frac{1}{M}, \frac{2}{M}], \cdots [\frac{M-1}{M}, 1]$ and draw $Np_i = \frac{N}{M}$ many of samples $\{U_j^i\}_{j=1}^{\frac{N}{M}}$

from each stratum $[\frac{i}{M}, \frac{i+1}{M}]$ uniformly. The stratified terminal value points of the Brownian bridge are obtained by transforming $\{U^i_j\}_{i,j=1}^{M,\frac{N}{M}}$ into $\{x^{i,j}_T\}_{i,j=1}^{M,\frac{N}{M}}$. Given each terminal value, the Brownian bridge could be generated using algorithm in section 5.2.1 and section 5.2.2. I implemented the exact Brownian bridge generation algorithm with stratified sampling in the file "Random_Brownian_Bridge.hpp" and "Random_Brownian_Bridge.cpp". I implemented the approximated Brownian bridge generation algorithm with stratified sampling in the file "Brownian_Bridge.py".

6 Program Result and Benchmark Result

Table 1: C++ implementation result for Geometric and Arithmetic Asian Option. Spot price $S_0 = 100$, risk free interest rate $r = 0.05$, dividend rate is $d = 0.02$, annualized volatility $\sigma = 0.3$, expire in one year $T = 1$, number of path $N = 10000$ for MC simulation, and number of observation $n = 16$ for discrete Asian option.

		Arithmetic Asian			Geometric Asian				
K		Park Miller	Sobol	Stratified BB	Park Miller	Sobol	Stratified BB	Closed_Discrete	Closed_Continuous
80	mean	21.1833	21.1474	21.2034	20.5651	20.525	20.5833	20.571	20.3685
	sd	0.168799	0.169024	0.169024	0.164492	0.16475	0.1648	na	na
	speed	0.02	0.023	0.042	0.035	0.04	0.037	0	0
90	mean	13.4687	13.4512	13.5206	12.95	12.933	13.0038	12.9633	12.6545
	sd	0.150117	0.150187	0.149945	0.144809	0.144888	0.144688	na	na
	speed	0.014	0.025	0.054	0.039	0.039	0.04	0	0
100	mean	7.7187	7.73537	7.77502	7.30299	7.31769	7.35415	7.31193	6.95361
	sd	0.122079	0.121838	0.121586	0.116103	0.115891	0.115711	na	na
	speed	0.018	0.02	0.043	0.027	0.046	0.068	0	0
110	mean	4.02329	4.01807	4.02908	3.70758	3.69579	3.7065	3.70439	3.38709
	sd	0.0915166	0.0912285	0.0910316	0.0852494	0.0850516	0.0849238	na	na
	speed	0.012	0.015	0.034	0.012	0.014	0.046	0	0
120	mean	1.93488	1.91955	1.91807	1.70952	1.69588	1.69576	1.7039	1.47918
	sd	0.0644312	0.0643179	0.0640556	0.058287	0.0582771	0.0581264	na	na
	speed	0.012	0.015	0.03	0.03	0.03	0.038	0	0
130	mean	0.885683	0.869677	0.877241	0.738096	0.724176	0.732666	0.721356	0.587983
	sd	0.043441	0.0435446	0.043064	0.0377445	0.0380218	0.0376446	na	na
	speed	0.012	0.035	0.025	0.011	0.014	0.026	0	0

6.1 Conclusion

All the algorithms above for pricing Asian option are implemented in C++ and benchmarked by QuantLib. The pricing result, standard deviation and speed are showed in Table 1. The associated QuantLib benchmark result is showed in Table 2. The convergence diagnostics analysis for estimated mean and standard deviation are plotted in Figure 3 and Figure 4.

Compared with program result in Table 1 and benchmark result in Table 1, the results in general matches with each other. The results for analytical formula of continuous geometric Asian option are exactly same, however, those of the discrete geometric Asian option are slightly different, the discrepancy is caused by the observation time. Since one year is not divisable for 16 observations used in Quantlib, it rounds off with 30 minutes resolution scale.

The standard deviation of Brownian bridge with stratified sampling achieve the minimum in

Table 2: QuantLib Benchmark Result for Geometric and Arithmetic Asian Option. Spot price $S_0 = 100$, risk free interest rate $r = 0.05$, dividend rate is $d = 0.02$, annualized volatility $\sigma = 0.3$, expire in one year $T = 1$, number of path $N = 10000$ for MC simulation, and number of observation $n = 16$ for discrete Asian option.

		Arithmetic Asian			Geometric Asian				
K		Park Miller	Sobol	Stratified BB	Park Miller	Sobol	Stratified BB	Closed_Discrete	Closed_Continuous
80	mean	21.038003	21.135275	21.117702	20.418196	20.512772	20.336461	20.561851	20.368462
	sd	0.167360	na	0.166353	0.164113	na	0.163076	na	na
90	mean	13.231104	13.432152	13.453186	13.261625	12.914065	12.972567	12.949595	12.654540
	sd	0.148002	na	0.151468	0.144453	na	0.144255	na	na
100	mean	7.555778	7.713805	7.618919	7.426611	7.297035	7.224186	7.296199	6.953600
	sd	0.121556	na	0.120023	0.117262	na	0.114618	na	na
110	mean	4.091444	4.008054	4.051418	3.706049	3.686738	3.737635	3.690390	3.387081
	sd	0.091300	na	0.092058	0.0852494	na	0.084934	na	na
120	mean	1.804135	1.914549	1.835899	1.628790	1.690058	1.639213	1.693843	1.479183
	sd	0.059853	na	0.061089	0.058287	na	0.056898	na	na
130	mean	0.831419	0.868601	0.790071	0.703181	0.721617	0.785031	0.715242	0.587979
	sd	0.040667	na	0.039473	0.036053	na	0.039043	na	na

general. Which conformed with our intuition that stratified sampling could enhance the sampling efficiency by evenly draw samples according to their predetermined probability. The standard deviation for Quasi Monte Carlo method is invalid. Since low discrepancy Sobol sequence used in simulation is not statistically mutually independent, rather they followed determined pattern which make them evenly dispersed in the given region. Such property increase the efficiency of Monte Carlo integration.

In terms of computational speed, pseudo random sequence by Park Miler achieves the fastest speed due to its simplicity of linear congruential generator. Sobol sequence shares the same order of speed as pseudo generator, but takes a bit of extra time due to its complexity of generation process. Stratified Brownian bridge algorithm takes the longest time, for the reason that it takes time to stratify the terminal domain and generate Brownian bridge.

6.2 Convergence Diagnostics

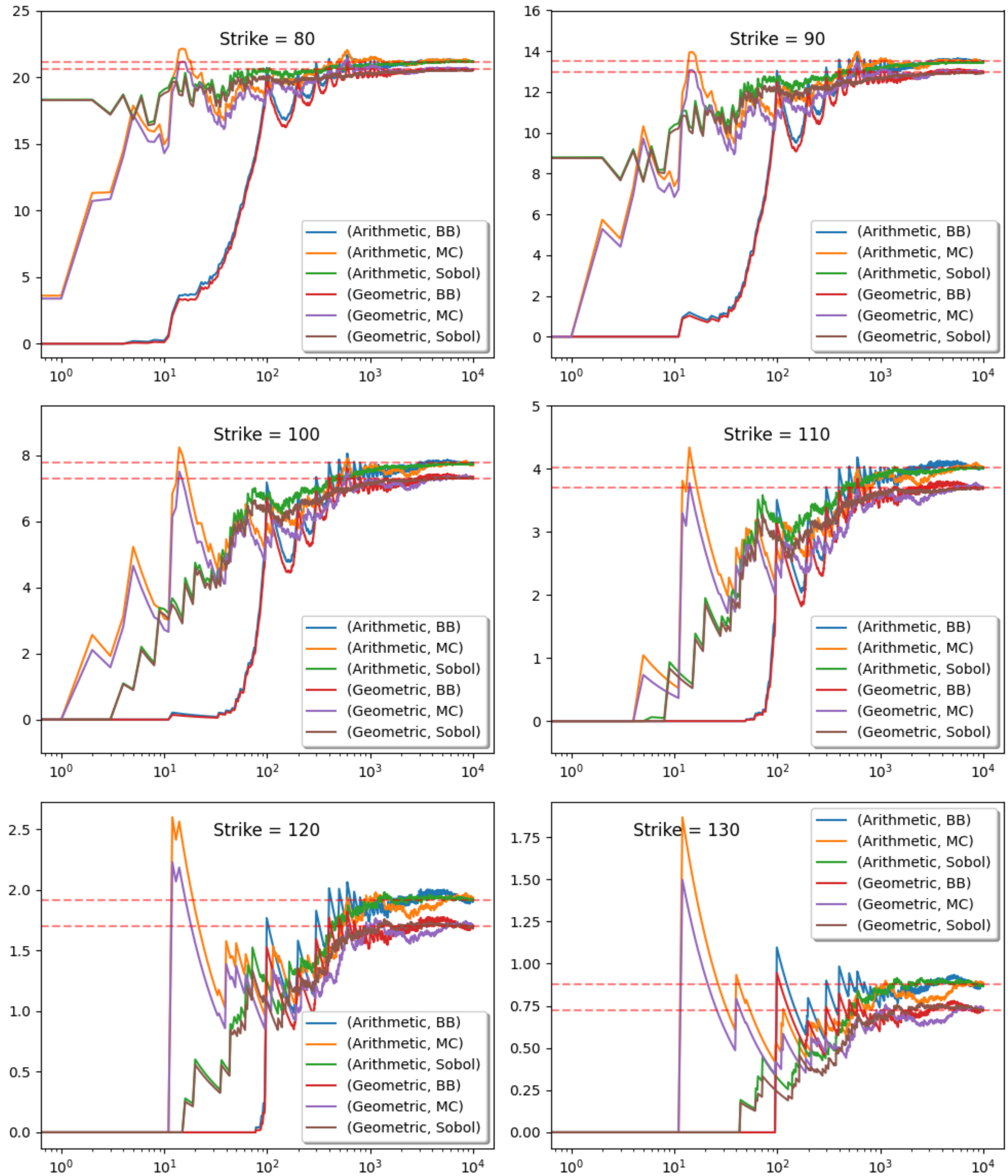


Figure 3: Convergence Diagnostics for Mean

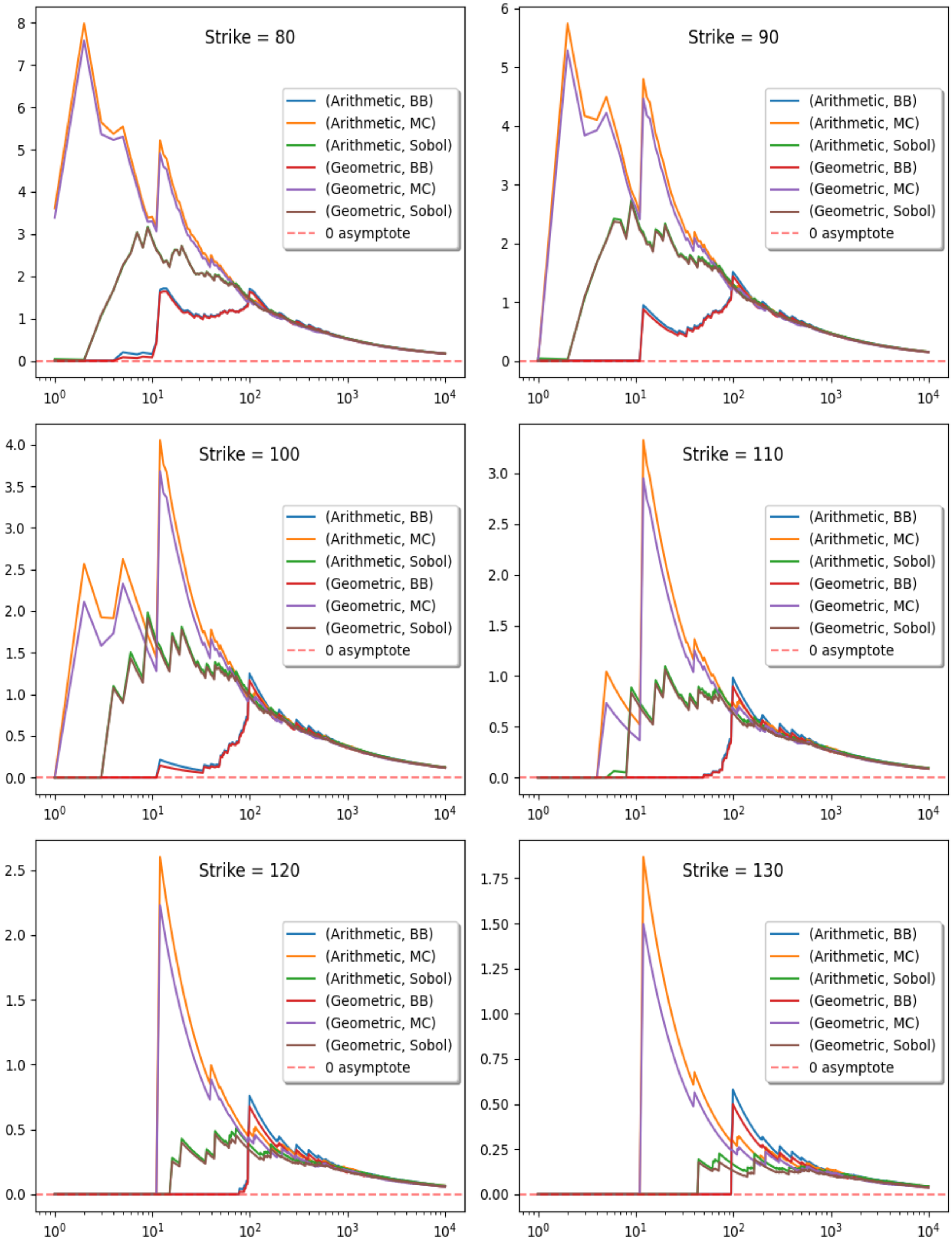


Figure 4: Convergence Diagnostics for Standard Deviation

References

Boyle, P., Broadie, M., and Glasserman, P. (1997). Monte carlo methods for security pricing. Journal of economic dynamics and control, 21(8-9):1267–1321.

Boyle, P. P. (1977). Options: A monte carlo approach. Journal of Financial Economics, 4(3):323–338.

Broadie, M. and Glasserman, P. (1996). Estimating security price derivatives using simulation. Management science, 42(2):269–285.

Curran, M. (1994). Valuing asian and portfolio options by conditioning on the geometric mean price. Management science, 40(12):1705–1711.

Fishman, G. S. and Moore, III, L. R. (1986). An exhaustive analysis of multiplicative congruential random number generators with modulus 2ˆ31-1. SIAM Journal on Scientific and Statistical Computing, 7(1):24–45.

Fusai, G. and Kyriakou, I. (2016). General optimized lower and upper bounds for discrete and continuous arithmetic asian options. Mathematics of Operations Research, 41(2):531–559.

Geman, H. and Yor, M. (1993). Bessel processes, asian options, and perpetuities. Mathematical finance, 3(4):349–375.

Glasserman, P. (2004). Monte Carlo methods in financial engineering, volume 53. Springer.

Harrison, J. and Kreps, D. M. (1979). Martingales and arbitrage in multiperiod securities markets. Journal of Economic Theory, 20(3):381–408.

Harrison, J. and Pliska, S. R. (1981). Martingales and stochastic integrals in the theory of continuous trading. Stochastic Processes and their Applications, 11(3):215–260.

Horvath, A. and Medvegyev, P. (2016). Pricing asian options: a comparison of numerical and simulation approaches twenty years later. Journal of Mathematical Finance, 6:810–841.

Joshi, M. S. and Joshi, M. S. (2004). C++ design patterns and derivatives pricing, volume 1. Cambridge University Press.

Kemna, A. G. and Vorst, A. C. (1990). A pricing method for options based on average asset values. Journal of Banking & Finance, 14(1):113–129.

Le Gall, J.-F. (2016). Brownian motion, martingales, and stochastic calculus, volume 274. Springer.

L'ecuyer, P. (1988). Efficient and portable combined random number generators. Communications of the ACM, 31(6):742–751.

Lehmer, D. H. (1951). Mathematical methods in large-scale computing units. Annu. Comput. Lab. Harvard Univ., 26:141–146.

Oksendal, B. (2013). Stochastic differential equations: an introduction with applications. Springer Science & Business Media.

Park, S. K. and Miller, K. W. (1988). Random number generators: good ones are hard to find. Communications of the ACM, 31(10):1192–1201.

Rubinstein, M. and Reiner, E. (1991). Breaking down the barriers. 4(8):28–35.

Shreve, S. E. et al. (2004). Stochastic calculus for finance II: Continuous-time models, volume 11. Springer.

Thompson, G. W. P. (1999). Fast narrow bounds on the value of asian options. Technical report.

Turnbull, S. M. and Wakeman, L. M. (1991). A quick algorithm for pricing european average options. Journal of financial and quantitative analysis, 26(3):377–389.