Seminar Report

Geometric Brownian Motion For Option Pricing

Filip Jansson

Jonathan Leung

Martin Westberg

Group 2

Introduction to Stochastic Processes, HT 2022

Course code: MAA320

Instructor: Ying Ni

Abstract

In financial modeling practices, stochastic processes are found to be a prominent figure of the inner workings of the financial market, and thereby also the derivatives market. This paper is a study on geometric Brownian motion (GBM), a Markov process, where the European options pricing model of Black-Scholes (BS) is the forefront. Appointing fitting values to the drift coefficient and volatility allows for simulating different outcomes with MATLAB, yielding GBM trajectories where our interest lie with the stock price at maturity (S_T). When a large number of S_T has been simulated, further calculations such as expected payoff, premium of the option and even exotic options are made. After providing stochastic theory of the geometric brownian motion process and running the simulations, option pricing, discussion and analysis ensues, featuring visual representations as well numerical evaluations.

The resulting graphs are directly connected to underlying stochastic theory. When modelling Brownian motion (BM), the numerous S_T produces a normally distributed histogram which is due to the drift being zero. The GBM histogram, however, has a lognormal histogram which correlates to the process being exponential. Centrally, enclosed by underlying theory, the same assumptions are clearly observable in the results. The numerical results of option prices reflects convergence to the BS model when increasing the number of simulated trajectories.

Appendix 1 includes produced MATLAB scripts which graphs and calculates in alignment with the disclosures in the definitions and method sections.

Keywords: Option pricing, geometric Brownian motion (GBM), Black-Scholes model.

INDEX

Abstract **2**

1. Introduction **4**

2. Definitions **4**

2.1 Assigned Parameter Values 5

Definition 1: Brownian Motion 5

Definition 2: Geometric Brownian Motion 6

Definition 3: Options 6

Definition 4: Put-Call Parity and Payoff function 7

Definition 5: Exotic Options (European basket option) 8

Definition 6: Exotic Options (Asian option) 8

Definition 7: Black Scholes model (European options) 9

3. Method **9**

4. Results **10**

5. Discussion **16**

6. Conclusion **17**

7. References **19**

Appendix 1 **20**

1. Introduction

Being widely used in mathematics and finance, GBM is commonly used to model growth of stock prices. When studying fluctuations of the financial market, existing noise clouds the initial judgement but assumptions like the no-arbitrage principle led to the famous Black-Scholes formula for European call and put options, where GBM models the underlying stock price. In the discrete case, random walks, specifically the binomial tree method, represent the fluctuations while GBM is the continuous case. [2]

From any time point, as time progresses, the initial growth rate expectedly changes and thus a drift arises in the GBM model. Therefore, it is not ergodic. Studying a lone system only allows observation of average growth from drift, where a collection of systems would also inhibit expected drift and randomness. Correspondingly, with assumption of perfect markets and lack of arbitrage opportunities, the BS model does not have a drift parameter. For purposes of simulation, the volatility parameter is acquired from the daily or historic data (or arbitrarily assigned, convention being around 30 per cent), and the drift coefficient is assigned to equate the risk-free interest rate or a estimated historic value. Running an extensive simulation produces a large amount of stock prices at maturity, S_T. That leaves for expected payoff of the option to be calculated, followed up by discounting the result to receive option price at time of investment, concluding with evaluation of European basket options. It is noteworthy that in order to simulate movements of option prices and the underlying stock with GBM, which is a continuous model, discrete-time has to be adopted. [3]

For an investor who enters a trade, the attention is directed at how, from the time of entry, the stock or option price varies until it is time to sell or exercise. Call and put options can be written or bought, and the profitability depends on how the stock and option prices moves from the moment of entry. However, for the main part, this study revolves European options, which are only exercised or discarded at time of maturity, T. [3]

2. Definitions

As a prerequisite, fundamental concepts are introduced in this segment to provide the tools necessary for the following sections of this paper. The course literature by S.M. Ross (2014) [1] is the main reference for this segment, and since authors may denote variables differently, this course literature is the chosen basis.

2.1 Assigned Parameter Values

The following conventional assignment of parameters is adopted: the risk-free annual interest rate, *r*, equals 0.05 (5%). Further, the drift parameter equals the aforementioned interest rate *r*. As for the annual volatility, σ equals 0.3 (30%).

Inherently, the drift and volatility are annual parameters that, to plot a trajectory with multiple data points, needs conversion to the daily format. That is achieved by:

$$\sigma_d = \sigma_a \cdot \frac{1}{\sqrt{252}} = \frac{0.3}{\sqrt{252}} = 0.0188 \tag{1}$$

$$\mu_d = \mu_a \cdot \frac{1}{252} = \frac{0.05}{252} = 0.0001984 \tag{2}$$

Where σ_a is the annual volatility and σ_d is the computed daily volatility. Henceforth, drift and volatility are implied to be in their annual formats unless explicitly addressed otherwise. In addition, the risk-free interest rate, *r,* is converted from annually compounding to continuously compounding, achieved by evaluating $ln(1 + r) = ln(1.05) = 0.04879$.

In relation to this, the discount factor alpha, α, has the following relationship:

$$\alpha = \mu + \frac{\sigma^2}{2}. \tag{3}$$

Accordingly, for the predetermined parameter values above, alpha is:

$\alpha_d = 0.0001984 + \frac{0.0188^2}{2} = 0.00037512$, and $\alpha_a = 0.05 + 0.3^2 \cdot \frac{1}{2} = 0.005$.

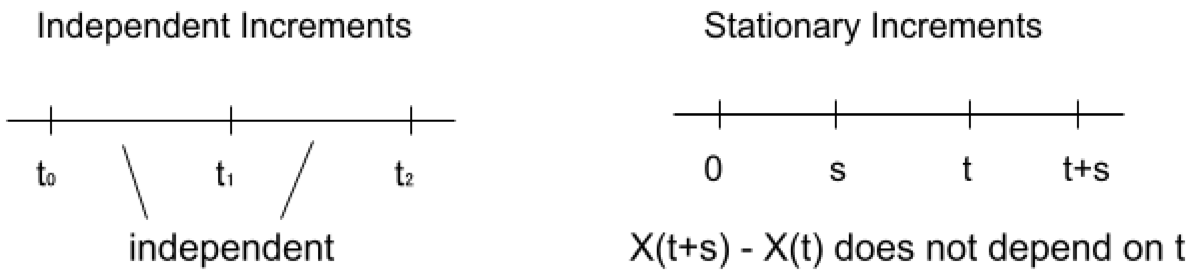
Finally, it is a convention within finance that a trading year represents 252 days where the market is open. Simulating a random walk for a stock price over 252 points therefore represents one calendar year.

Definition 1: Brownian Motion

A Brownian motion process, occasionally termed Wiener process, is a prominent stochastic process and the prerequisites are, including the Markovian property:

$X(0) = 0$, $\{X(t), t \geq 0\}$, and for all $t > 0$ then $X(t)$ has a normal distribution around mean 0 with $Var(X(t)) = \sigma^2 t$.

The Markovian property entails that the future only depends on the present; not the past (memorylessness). Furthermore, a Brownian motion $\{X(t), t \geq 0\}$ has stationary and independent increments. A process with stationary increments only depends on the length of an arbitrary interval for the distribution of the change in position. Additionally, independent increments have independence between any disjoint interval.



Definition 2: Geometric Brownian Motion

Now, consider $\{Y(t), t \geq 0\}$, a Brownian motion process, with σ being the volatility parameter and μ the drift parameter (both constants). Then, GBM is defined as

$X(t) = e^{Y(t)}$.

$dS = \mu \cdot Sdt + \sigma \cdot Sdz$

$d \cdot ln(S) = \left(\mu - \frac{\sigma^2}{2}\right)dt + \sigma dz,$

(4)

Moreover, GBM has stationary and independent intervals as with a Brownian motion process. As is customary in finance, S and t represent stock price and time, respectively. Furthermore, $dz = \epsilon\sqrt{dt}$ which corresponds to the Weiner process, a type of stochastic process. Equation (4) is attained by taking the logarithm of both sides, and it entails that S is lognormally distributed. The GBM process yields a risk-neutral process with which a risk-neutral expectation can be assessed.

Definition 3: Options

Hull (2015) [6] describes in his work that an options contract is a financial instrument that follows a two-party system. The two parties consist of a buyer of the option, known as the holder, and a seller of the option, known as the issuer. The issuer has the obligation to complete the transaction at a predetermined strike price, K, if the holder decides to buy or sell. Buying or selling the option is referred to as exercising the option. Furthermore, the holder of a contract have the rights, but not obligations, to exercise it.

The effects of exercising an option depends on if it is a Put or Call option. A Put option gives the holder the right to sell the underlying asset while a Call option gives the holder the right to buy the underlying asset. There is a wide variety of financial assets that are used as underlying asset for contract such as options.

The two most popular option types are American and European. The main differences between the two are the time at which an option can be exercised. American options can be exercised at any time before the option expires whereas the European option can only be exercised at expiration. An expiration date is the predetermined time limit before the contract can no longer be used. Lastly, there are non-standard options available with special features.

To get results comparable with BS-model, the expected payoffs obtained with brownian motion MATLAB script (III), (IV) and (V) are discounted to time zero in a Monte-Carlo method.

Definition 4: Put-Call Parity and Payoff function

Hull (2015) [6] describes that an investor can be in a short position by betting against a contract or a long position by betting for a contract. The payoff charts look as follows:

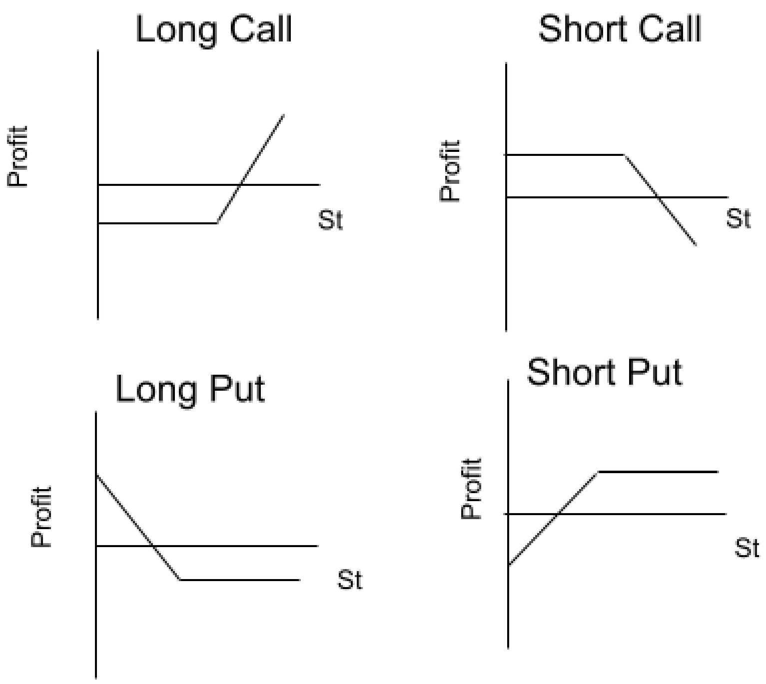


Figure 1: Payoff charts for the different possible option positions.

The conditions for Put-Call Parity can be described by an equality required to prevent arbitrage. Arbitrage happens when the value of two options together offer risk-free profit.

$$S_0 + P_0 = \frac{K}{e^{rt}} + C_0 \quad (5)$$

This means that both options currently have the same value when adjusted for the risk-free interest rate *r*, provided that it is the same stock with the same expiration *T* and strike price *K*.

Put Option $$Payoff = max(K - S_T, 0) \quad (6)$$

If the stock price is below the strike price at maturity, the put option will be exercised, selling the stock at a payoff of $K - S_T$. If the stock price is higher than the strike price at maturity, the option will not be exercised at a payoff of 0.

Call Option $$Payoff = max(0, S_T - K) \quad (7)$$

If the stock price is cheaper than the strike price, the option will not be exercised at a payoff of 0. However, if the stock price is higher than the strike price at maturity the option will be exercised at a payoff of $S_T - K$.

Definition 5: Exotic Options (European basket option)

Exotic options are variations of American and European options following the same principles with minor alterations. European basket options are based on averaged or weighted grouped assets, and the strike price is conventionally the at-the-money value. The option allows the owner to buy or sell the stock at the average or weighted price of the grouped assets. Consequently, the payoff functions of European basket options are:

Call $$Payoff = max(S_{Basket} - K_{Basket}, 0) \quad (8)$$

Put $$Payoff = max(0,\ K_{Basket} - S_{Basket}) \quad (9)$$

Definition 6: Exotic Options (Asian option)

Exotic options are variations of American and European options following the same principles with minor alterations. European basket options are based on averaged or weighted grouped assets, and the strike price is conventionally the at-the-money value. The option allows the owner to buy or sell the stock at the average or weighted price of the grouped assets. Consequently, the payoff functions of European basket options are:

Call $$Payoff = max(S_{Average} - K, 0) \quad (10)$$

Put $$Payoff = max(0, K - S_{Average}) \quad (11)$$

Definition 7: Black Scholes model (European options)

$$c = S_0 \cdot N(d_1) - K \cdot e^{-r\cdot T} N(d_2) \quad (12)$$

$$p = K \cdot e^{-r\cdot T} N(-d_2) - S_0 \cdot N(-d_1) \quad (13)$$

where

$$d_1 = \frac{1}{\sigma\sqrt{T}} \cdot \left(\frac{ln(S_0)}{K} + \frac{r+\sigma^2}{2} \cdot T \right)$$

and

$$d_2 = d_1 - \sigma\sqrt{T}$$

Key assumptions for the BS model: lognormal distribution, options can only be exercised at expiry (European), no dividends, risk-free interest rate, volatility is constant and there is no arbitrage opportunities.

The *N*-function is a cumulative p.d.f. for a standard-normal distribution, further explained as the probability that a standard-normal random variable $Z \sim norm(0, 1)$ is less than a number *x*, $P(Z < x)$

3. Method

Having established the background and connection between concepts, testing the theory of GBM can be done accordingly: simulate a million stock prices and save the value at maturity, S_T, and calculate the expected option payoff with nearby strike prices K. Thereafter, discount the result to acquire the options' premium at the time zero (the beginning moment of the contract). This works because the process simulated is per definition risk-neutral. A goal with the investigation is seeing whether the GBM process converges to BS model. MATLAB is used for ordinary BS calculations.

Learning about stochastic processes invites for experimentation. Arbitrarily assigning parameter values is not the sole alternative; this paper features MATLAB script (I) which estimates historical volatility and drift from Yahoo finance's historical data page with discretional time interval. In Sweden, OMXS30 is a main stock index for the country. The historical data of OMXS30 is downloaded from Yahoo Finance with date range 05 Oct. 2021 to 05 Oct. 2022 where the file is in comma-separated-values (.csv) format. The final index close price value within the time interval is 1909.6 and it will serve as the starting point of the intended simulations, i.e. $S_0 = 1909.6$. Using different parameter values grants different outcomes for the price development of underlying asset when simulations are run and thus any option price comparisons should be with fixed, respective parameter values.

Worthy of note is that brokers within the real financial market charges their investors a courtage fee, which will be disregarded in this report. However, it could be introduced as an error within risk assessment, perhaps at a level of 0.5% fee of total amount per transaction.

4. Results

The benchmark of comparison is the BS model, which GBM converges to, being the discrete-time binomial tree process. The MATLAB script (VI) calls MATLAB's internal BS function with parameter values $S_0 = 1909.6,\ \mu = 0.05,\ \sigma = 0.3,\ K = 1950$, yielding:

$Call = 252.8264$ and $Put = 198.1238$ (14)

To verify, these values are plugged into the put-call parity equation (5) below, which indeed shows that $r = 0.05$.

$$1909.6 + 198.1238 = 1950 \cdot e^{-rT} + 252.8264.$$

With the aim to examine option pricing by simulating geometric brownian motion over a year, the values assigned to the parameters for simulating GBM are as laid out in subsection 2.1:

$$\sigma_d = \sigma_a \cdot \frac{1}{\sqrt{252}} = \frac{0.3}{\sqrt{252}} = 0.0188$$

$$\mu_d = \mu_a \cdot \frac{1}{252} = \frac{0.05}{252} = 0.0001984$$

$$r_c = ln(1 + r_a) = ln(1.05) = 0.04879$$

Further, the parameters estimated with script (I), with aforementioned OMXS30 interval are:

$$\sigma_a = 0.04518$$

$$\mu_a = -0.1758$$

To illustrate the general idea of GBM (with the same intuition underlying BM), running the MATLAB script (II) results in the following graph:

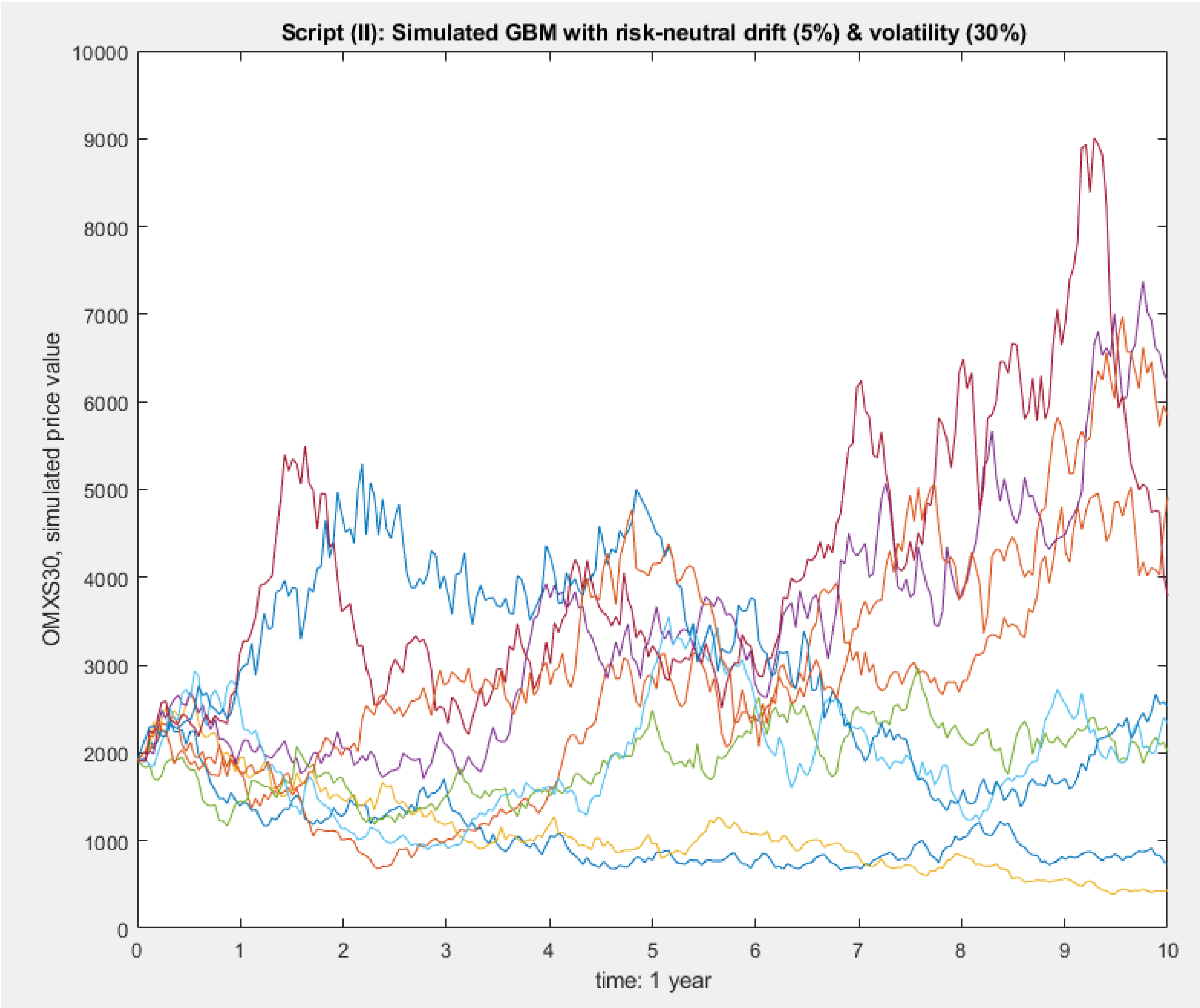


Figure 2: Nine simulated GBM trajectories, script: (II).

In Figure 2, the underlying idea is that each trajectory is a random walk over time, consisting of up- and down movements from the starting point. The end of the trajectories (to the right) is the simulated asset price at maturity, S_T. Applying the payoff function and evaluating the expectancy of the nine payoffs, then discounting for time to obtain the option's premium is the core principle of the following paragraphs. It follows from theory that a mean GBM trajectory should grow by the risk-free interest rate, which is $\mu = 0.05$.

As established in the definitions section, GBM is an exponential process with a non-zero drift. The histogram of GBM is thus lognormal. Since the payoff is dependent of *K* and whether it is a call or put option, there are multiple payoffs to consider. Further so, the exotic European basket option payoffs of both the call- and put varieties are evaluated. Arbitrarily, but consistently, assigning a strike price *K* allows use of payoff equations (6) through (11). and the BS model is evaluated with MATLAB script (VI). The numerical results are best visualized with script (III)'s plots:

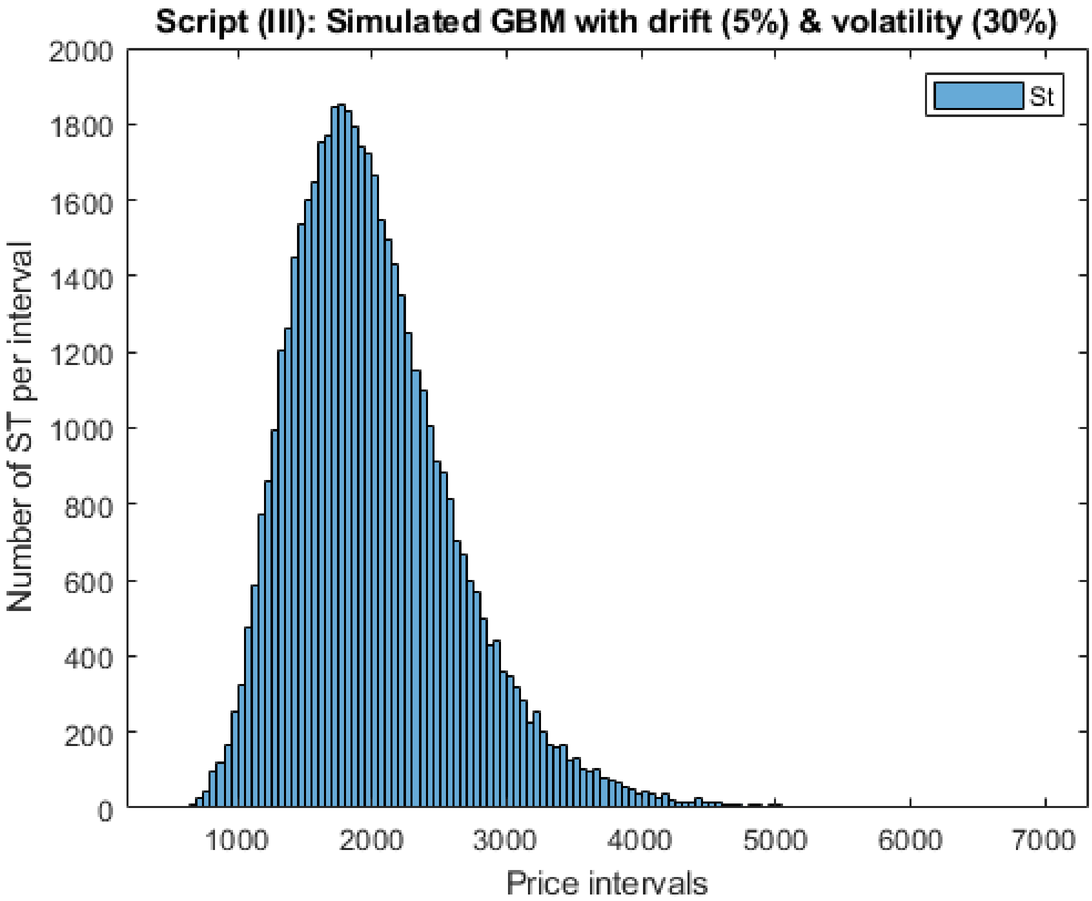


Figure 3: Lognormally distributed histogram of S_T, script: (III).

The results in figure 3 agrees with MATLAB's built in function for GBM simulation as displayed below:

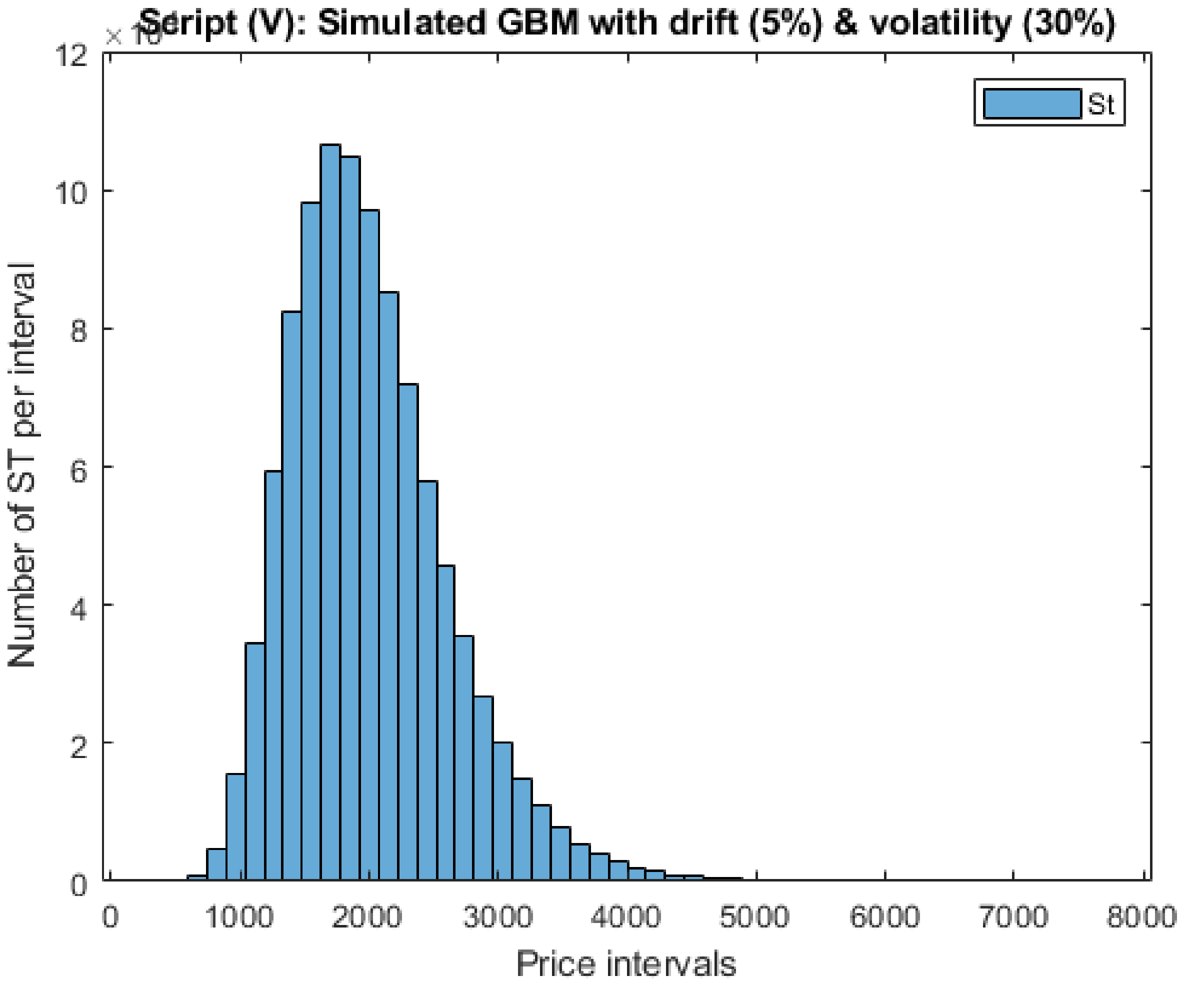


Figure 4: Lognormally distributed histogram of S_T, script: (V).

Having attained realized stock prices at maturity, the payoffs are shaped to produce the following option prices simulations:

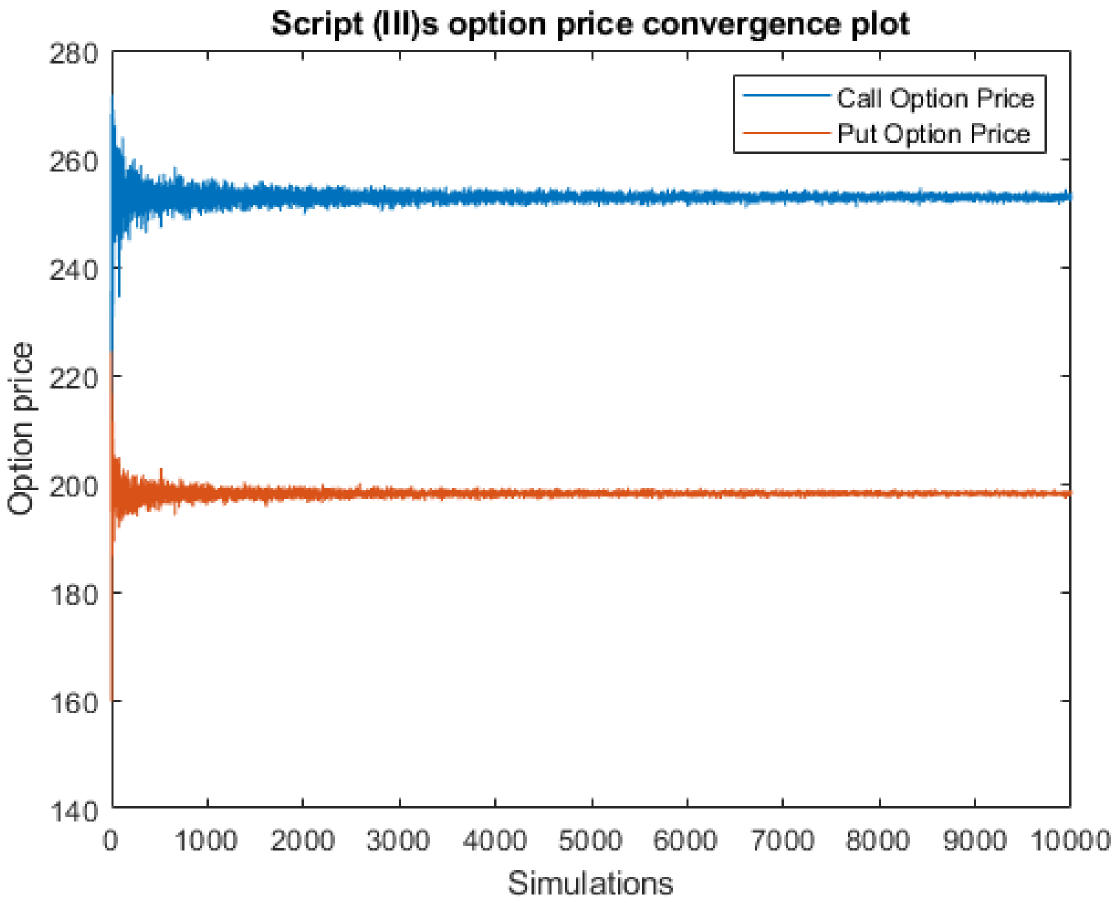


Figure 5: European option price converges, script: (III).

Running script (III)'s option price convergence plot yields figure 5. Further support is provided by script (IV) which also plots option price convergence:

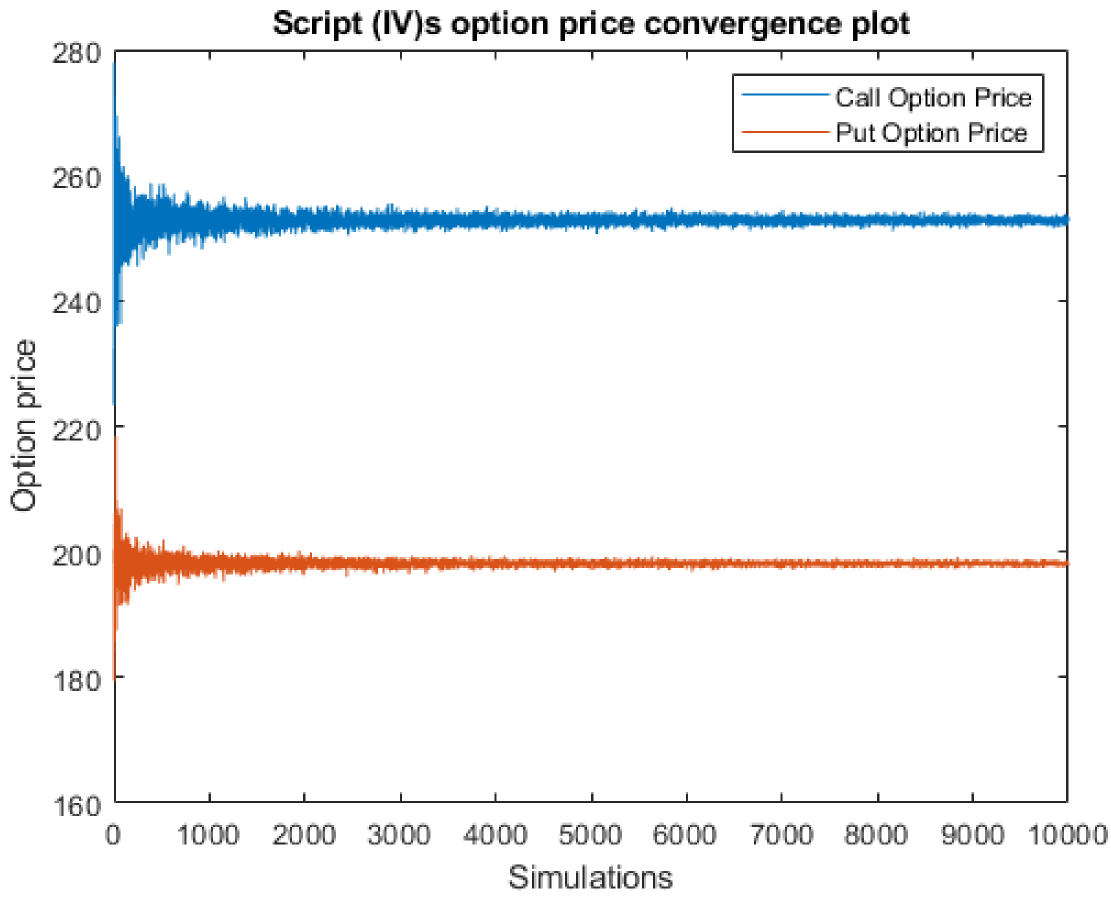


Figure 6: European option price converges, script: (IV).

In the simulations, the mean of resulting prices at maturity, mean(S_T), varies and thereby also the option prices. With enough simulations, a convergence toward the BS model occurs. Projecting two horizontal lines at price levels where the simulated option prices approach, going left to right, shows that the GBM option prices simulated are converging to the BS model's results, which are stated in equation (14), here restated:

$Call = 252.8264$ and $Put = 198.1238$.

The conclusive results are now followed up with an alteration. For European basket options, the payoff is different and implementing the method similarly, the convergence plot produced with script (VII):

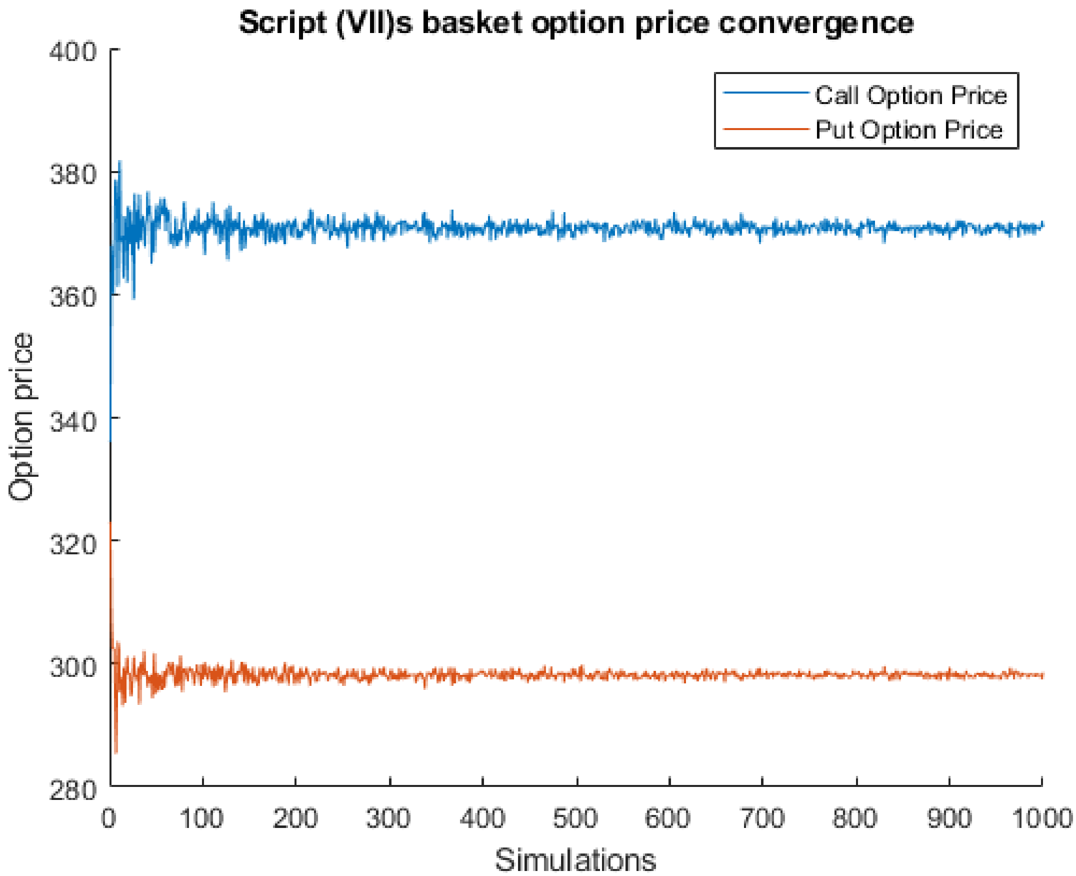


Figure 7: European basket option price converges, script: (VII).

The European basket option price converges to roughly $Call = 370.79$ and $Put = 298.18$. This coincides with executing script (VI) with $K = 2900$ and $S_0 = 2831.2$.

The conclusion section addresses the implications of these European basket option pricing results. Before the discussion section, the Asian option is also examined.

As for the Asian option, the payoff has a different structure and with support from previous results, the Asian option price was evaluated to $Call = 139.98$ and $Put = 130.28$. This will also briefly be addressed in the conclusion section.

5. Discussion

The option prices were simulated with the stochastic theory of GBM as a primary navigation tool and MATLAB's accurate production of visual representations were successful. However, execution of the simulation was important and inputs needed to be established. Although it is possible to simulate inaccurate data that does not agree with real market forces, when simulated and interpreted correctly, it leads to improved knowledge of stochastic calculus and its applications. Furthermore, implementing MATLAB helps to conceptualize graphical output, and interpreting the output reassures the user whether the parameters and equations are suitably applied. It is thus possible to recover from taking a few missteps, optimize the approach and become more knowledgeable of applied stochastic calculus.

GBM is a non-negative stochastic process, and it suits financial modeling well since, similarly to the BS model and put-call parity, it should be assumed that there should be no arbitrage opportunities in buying and selling stocks. In addition, the Markovian property applies to GBM, meaning the future only depends on the present state; not the past.

Proposed further experimental studies are conceivable using MATLAB script (I) which allows the user to calculate historical parameter values with which the GBM simulating MATLAB scripts (II), (III), (IV) and (V) can produce different outcomes. The theory of the Black-Scholes model assumes volatility to be constant, but in real markets that is not realistic. Experimenting with varying parameters (constant, linear or not) is part of risk assessment which is not within the scope of this paper. The expected result is that using a higher (lower) value of the drift parameter will tend to produce higher (lower) simulated stock prices at maturity which affects the payoffs of the options and thus their prices. As for the one-year OMXS30 data, the mean return was negative.

The benefit of using multiple coded scripts with the same purpose is that the validity of the results strengthens. In addition, running some large simulations in MATLAB can take long time to execute, in particular script (III) as it builds the expected payoffs for the N trajectories. Faster is script (IV) and (V) but when increasing N it is justified to direct the focus towards the faster. In particular, script (IV) only produces 1 data point S_T at time T=1 and not a full year. On the other hand, script (II) and (III) are plotted over a year so they can display trajectories. A relevant delimitation for the scripts in Appendix 1 is that for large sizes, MATLAB needs increasing memory to handle large workspaces and if the code is not optimized that poses an issue, that would be attributed to human error.

6. Conclusion

In stochastic calculus, GBM is time-continuous but for MATLAB implementation it is adapted to be discrete-time. As the simulation concluded it shows that creating many S_T makes for favorable histograms that conform to lognormal distributions. They can be the foundation, i.e. underlying asset for options contracts. The numerical results are displayed in Figure 5, 6 and 7. For Figure 5 and 6, the option prices converge to:

$Call = 252.8264$ and $Put = 198.1238$.

Approaching the results from multiple angles, the conclusions strengthen each other when visual, theoretical and numerical come together. Since the results align with underlying theory, the results conclude that the geometric brownian motion process, in conjunction with option pricing, approaches the binomial tree method as the number of iterations are increased enough to show the pattern.

As for the European basket option, the conclusion is that constructed assets can have option contracts traded on top. When altering the weights, S_0 changes and the BS model is still a valid calculation to apply with the new parameters in place. The support behind this is that the simulated process is risk-neutral and from that perspective all assets have a growth of the risk free interest rate. Thus, the derivatives market can be built on constructed assets. Venturing this market presents opportunities to speculate or hedge capital. The final tested calculation of Asian options has a different from normal payoff and therefore price. Thus, it was not compared numerically with the other results.

The attained results in conjunction with underlying theory justifies a useful conclusion: the tendencies of stock- and option prices can be partially decoded and estimated. In finance, geometric brownian motion underlies the theoretical constituents of stock price movements, which can be considered as random walks. In conclusion, the Black-Scholes model option prices agree with option pricing with geometric brownian motion.

7. References

[1] S.M. Ross. (2014). Introduction to Probability Models, 11th Edition. Elsevier: Amsterdam.

[2] R.C. Merton. (1971). Optimum Consumption and Portfolio Rules in a Continuous-time model. *Journal of Economic Theory, 3:4*. Elsevier: Amsterdam.

[3] K.D. Brewer, Y. Feng, C.C.Y. Kwan. (2012). Geometric Brownian Motion, Option Pricing, and Simulation: Some Spreadsheet-Based Exercises in Financial Modeling. *Spreadsheets in Education*, *5:4*.

[4] K,I, Moalosi-Court. (2019). Pricing European Option When the Stock Price Process Is Being Driven by Geometric Brownian Motion. *Open Access Library Journal*, *6:5568*.

[5] J.C. Cox, S.A. Ross, M. Rubinstein. Option pricing: A simplified approach. *Journal of Financial Economics, 7 (3): 229*.

[6] J. Hull. (2015). Options, futures, and other derivatives, 9th Edition. Boston: Prentice Hall.

Appendix 1

(I) Estimate historic mu & sigma

```
% Script (I) Imports data from .csv file and calculates volatility & mean.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all,
clc
OmxSeCP=readtable('OmxSe.csv').AdjClose; % OMXS30, 5th oct 2021 to 5th oct 2022.
N=size(OmxSeCP);
n=N(1,1);
dt=1/252;
logRet=diff(log(OmxSeCP));
meanO=mean(logRet); % mu
S=zeros(n,1);
for i=1:n-1
S(i,1)= (logRet(i,1)-meanO)^2;
end
sumO=sum(S);
S2u=1/(n-1)*sumO;
sigma=S2u/sqrt(dt) % estimated volatility
meanO
% Output: sigma(daily)= 0.002846212151597, mean(daily)= —0.000697629857478299
% Annualized: sigma(yr)=0.04518, mean(yr)= -0.1758
```

(II) GBMtrajectplot

```
% Script (II) Produces a series of stock price trajectories over a year with
% Geometric Brownian Motion.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all,
clc
S0=1909.6; % Choose initial stock price value
n=252; % Choose number of how many desired data points
mu=0.05; % Here, choose drift-coefficient value (yearly average).
sigma=0.30; % Here, choose yearly volatility value.
max=3; % Desired # of GBM plots (run code again to gen. more trajectories)
h=10/n; % Making the X-vector for plot
sh=sqrt(h);
r=mu-sigma^2/2;
mh=r*h; % r has mu & sigma.  mh is like the multiplier to priceinc
for nrgraphs=1:max
  priceincrement=S0;
  P(:,1)=[0 S0];
  % For putting prices on n=252 places (setting y values to x-axis)
  for incrementx=2:n % idea is graph either moving a bit up or down
      random=randn(); % randomness into price movement
      priceincrement=priceincrement*exp(mh+sh*sigma*random);
      P(:,incrementx)=[incrementx*h priceincrement];
  end
  plot(P(1,:),P(2,:));
```

```
  hold on;
end
title('Script (II): Simulated GBM with drift (5%) & volatility (30%)');
xlabel('time: 1 year');
ylabel('OMXS30, simulated price value');
```

(III) GBM Simulation, robust exponential I.

```
%  Script (III) Produces a histogram of a year based on Geometric Brownian motion,
%  put & call evaluations and plots option price convergence.
%  With N larger than 50,000 this script takes a long time to finish.
%  Finally, the script calculates Asian option price.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
clc
dt=1/252; % Time Step Delta_t (dt)
S0=1909.6; % Initial Stock Price S0
mu=0.05; % Drift mu
Sigma=0.3; % Volatility Sigma
n=252; % Interval size for Asian option
Time=linspace(dt,0,1);
S=ones(n,1);
Drift=ones(n,1);
Uncertainty=ones(n,1);
r=zeros(n,1);
% dSt= St*mu*dt+ St*sigma*dz    / Drift + Uncertainty
Drift(1,1)=mu*dt*S0;
Uncertainty(1,1)=normrnd(0,1)*sqrt(dt)*Sigma*S0;% dz= epsilon*sqrt(dt)
change(1,1)=Drift(1,1)+Uncertainty(1,1);
N=50000;
St=zeros(N,1);
alpha=mu-Sigma^2/2;
K=1950;
E_PayoffEuC=0;
E_PayoffEuP=0;
E_PayoffAsC=0;
E_PayoffAsP=0;
for j=1:N
S(1,1)=S0;
for i=2:n
S(i,1)=S(i-1,1)*exp((alpha)*dt+normrnd(0,1)*sqrt(dt)*Sigma);
end
St(j,1)=S(end,1);
E_PayoffEuC=E_PayoffEuC+max((St(j,1)-K),0)*1/N; % for EU Call Option
E_PayoffEuP=E_PayoffEuP+max(K-(St(j,1)),0)*1/N; % for EU Put Option
E_EuCdisc=E_PayoffEuC*exp(-mu); % Discounted to time 0 (Call Premium)
E_EuPdisc=E_PayoffEuP*exp(-mu); % Discounted to time 0 (Put Premium)
meanS(j,1)=mean(S); % Simplified with 1 sample data, interval size n.
E_PayoffAsC=E_PayoffAsC+ max((meanS(j,1)-K),0)*1/N; % for Asian Call Option
E_PayoffAsP= E_PayoffAsP+ max((K-meanS(j,1)),0)*1/N; % for Asian Put Option
end
```

```
figure(1)
histogram(St); hold on
histfit(St)
format bank
mean(St)
% Example Output: mean(St)=2003.48, E_EuCdisc=252.97, E_EuPdisc=197.64
% Example Output 2 (Asian Option): E_PayoffAsP= 130.2821, E_PayoffAsC= 139.9824
title('Script (III): Simulated GBM with drift (5%) & volatility (30%)');
xlabel('Price intervals');
ylabel('Number of ST per interval');
%figure(2)
%plot(Time,S,'b'); % This command plots one of the many random walks.

%% Script (III)s option price convergence plot (Run sections separately)
T=1;
N=linspace(100,1000000,10000); % number of sims
for i=1:10000
St = S0 * exp((alpha) * T + Sigma*sqrt(T) * normrnd(0, 1, [1, N(1,i)]));
payoffEU_C(i,1) = exp(-mu * T) *  mean(max(St - K, 0));
payoffEU_P(i,1) = exp(-mu * T) *  mean(max(K - St, 0));

end
figure(1)
plot(payoffEU_C); hold on
plot(payoffEU_P); hold off
title('Script (III)s option price convergence plot');
xlabel('Simulations');
ylabel('Option price');
```

(IV) GBM Simulation, exponential II.

```
%  Script (IV) simulates Geometric Brownian Motion at time 1 immediately
%  and determines expected option payoff and discounts it to time zero.
%  It concludes by plotting option price convergence.
%  The script can not plot GBM trajectories.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%  Section 1:
clear all,
clc
%
%
S0 = 1909.6; % initial price
T = 1; % time to maturity
mu = 0.05; % short rate
Sigma = 0.3; % volatility
K = 1950; % strike price
%N = 1000; % number of sims
%N = 10000; % number of sims
%N = 100000; % number of sims
N = 1000000; % number of sims
```

```
alpha = mu - (Sigma^2)/2;
St = S0 * exp((alpha) * T + Sigma*sqrt(T) * normrnd(0, 1, [1, N]));
E_PayoffEuC = exp(-mu * T) *  mean(max(St - K, 0));
E_PayoffEuP = exp(-mu * T) *  mean(max(K - St, 0));
E_EuCdisc = E_PayoffEuC*exp(-mu*T); % Discounted to time 0 (Call Premium)
E_EuPdisc = E_PayoffEuP*exp(-mu*T); % Discounted to time 0 (Put Premium)
mean(St);
% Example Output: Call=252.6112, Put=198.1543, mean(St)=2007.2489

%% Script (IV)s option price convergence plot (Run sections separately)
N=linspace(100,1000000,10000); % number of sims
for i=1:10000
St = S0 * exp((alpha) * T + Sigma*sqrt(T) * normrnd(0, 1, [1, N(1,i)]));
payoffEU_C(i,1) = exp(-mu * T) *  mean(max(St - K, 0));
payoffEU_P(i,1) = exp(-mu * T) *  mean(max(K - St, 0));
end
plot(payoffEU_C); hold on
plot(payoffEU_P); hold off
% log(N)
title('Script (IV)s option price convergence plot');
xlabel('Simulations');
ylabel('Option price');
```

(V) GBM Simulation with built-in Matlab function

```
%  Script (V) simulates Geometric Brownian Motion with MATLAB's built in
%  function over a year and plotts a histogram, which confirms Script (III).
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
clc
dt=1/252;
t=252;
S0=1909.6;
mu=0.05;
sigma=0.3;
K=1950;
N=1000000;
GBM = gbm(mu, sigma, "StartState", S0);
[x, y] = GBM.simulate(t, "DeltaTime", dt, "Ntrials", N);
figure(1)
histogram(x(end,:,:),50); hold on
mean(x(end,:,:))
%figure(2) % For plotting N simulated price movements
%plot(y, squeeze(x))
title('Script (V): Simulated GBM with drift (5%) & volatility (30%)');
xlabel('Price intervals');
ylabel('Number of ST per interval');
% Example Output: mean(St)=2007.7
```

(VI) Black-Scholes Simple Evaluation

```
%  Script(VI) is a basic Black-Scholes Model valuation.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mu=0.05;
sigma=0.3;
S0=1909.6;
K=1950;
[Call,Put] = blsprice(S0,K,mu,1,sigma)
% Definitive Output: Call=252.8264, Put=198.1238 (serves as benchmark)
% For S0=2831.2 from script (VII), and K=2900, Black Scholes results:
% Call=370.7918, Put=298.1821
```

(VII) GBM Simulation Basket Option

```
%  Script(VII) is a mixture of previous scripts. It simulates Geometric
%  Brownian Motion and calculates European Basket option payoff and
%  discounts time to zero. It then plots the option price convergence.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
S0M = [1909.6 3752.75]; % initial price for OMX and S&P
weights = [.5; .5];
S0 = S0M * weights; % in this case, S0=2831.2
T = 1; % time to maturity
mu = 0.05; % short rate
sigma = 0.3; % volatility
K = 2900; % strike price
alpha = mu - (sigma^2)/2;
N = 1000000;
dN = 1000;
E_PayoffEuC = zeros(N/dN,1);
E_PayoffEuP = zeros(N/dN,1);
for n = dN:dN:N
   St = S0 * exp((mu - 0.5 * sigma^2) * T + sigma*sqrt(T) * normrnd(0, 1, [1,
n]));
   E_PayoffEuC(n/dN,1) = exp(-mu * T) *  mean(max(St - K, 0));
   E_PayoffEuP(n/dN,1) = exp(-mu * T) *  mean(max(K - St, 0));
end
E_EuCdisc = E_PayoffEuC*exp(-mu*T); % Discounted to time 0 (Call Premium)
E_EuPdisc = E_PayoffEuP*exp(-mu*T); % Discounted to time 0 (Put Premium)
hold on
plot(E_PayoffEuC)
plot(E_PayoffEuP)
hold off
% Example Output: Call=370.7238, Put=298.1846
```