1. Abstract

In this exercise, we study three continuous time stochastic process used widely to price financial derivatives: Geometric Brownian Motion (GBM), the Ornstein–Uhlenbeck (OU) process,
and the Cox–Ingersoll–Ross (CIR) model. We also apply the Black-Scholes formula to price European call options and compare the result with that achieved through Monte Carlo simulation. Zero coupon bonds are also priced using the MC simulations and CIR model.

2. Code Structure

We have adding all the prototypes for this exercise in pricer.h. The entry-point into the program is *TaskManager#execute()* which is the only public method in the header. The rest of the functions have private encapsulation. When *TaskManager#execute()* is invoked, the user be prompted with various options corresponding to the particular task. The user can perform as many tasks as they want without having to rerun the program. We have achieved this using a *do-while* loop.

3. How to Execute

To run the program, execute *./run.sh* at the base of the project. *run.sh* is a shell script that combines the compile and runtime steps of the program. It makes it easier for the user to interact with it.

```
Select Mission:
   1: Price European Call using Black-Scholes
   2: Price European Call using Monte Carlo Simulation
   3: Compare Monte Carlo and Black-Scholes Option Prices
   4: Price Asian Call using Monte Carlo Simulation
   5: Generate data dump of interest rate process using Ornstein-Uhlenbeck model
   6: Generate data dump of interest rate process using Cox-Ingersol-Ross(CIR) model
   7: Price Zero coupon bond using Monte Carlo and CIR for interest rate process
   0: Exit
Choose Mission: 1
```

4. Results and Conclusions

4.1. Pricing of European Call Option

4.1.1. Black-Scholes

The Black-Scholes pricing model provides a closed form solution to the price of an option with the following assumptions:

A. The options are European and can only be exercised at expiration
B. No dividends are paid out during the life of the option
C. Efficient markets (i.e., market movements cannot be predicted)
D. Arbitrage-free markets
E. No commissions
F. The risk-free rate and volatility of the underlying are known and constant
G. Underlier returns on the underlying are normally distributed.

More details about the formulas can be found in *BlackScholes.pdf* on the course page under *modules.* The implementation for this is in *TaskManagelackScholes*(). The inputs include price of underlier S, strike price K, time to maturity T, risk-free interest rate r and volatility of S. Here is a sample result:

```
Choose Mission: 1
--------------------------
GOAL: Price European Call using Black Scholes
Please enter S, K, T, r, vol. (space delimited - order matters): 20 18 1 0.1 0.25
Inputs: S=20, K=18, T=1, r=0.1, vol=0.25)
Price: 4.22207
--------------------------
```

4.1.2. Monte Carlo Simulation

Another more widely used way to price options is using Monte Carlo simulations. In this methodology, we generate thousands of possible underlier price paths (assuming log-normal distribution for the returns). At maturity time T, we evaluate the expected value of the option i.e. E[S(T)-K] by considering all the final prices. The inputs include price of underlier S, strike price K, time to maturity T, risk-free interest rate r, volatility of S, number of time slices N and number of simulations M. Here is a sample result:

```
Choose Mission: 2
--------------------------
GOAL: Price European Call using Monte Carlo Simulation
Please enter S, K, T, r, vol, N and M (space delimited - order matters): 20 18 1 0.1 0.25 1000 100000
Inputs: S=20, K=18, T=1, r=0.1 , vol=0.25, N=1000, M=100000
Price: 4.23258
--------------------------
```

4.1.3. Black-Scholes Price Vs Monte Carlo Price

The expectation is that the Black-Scholes and Monte Carlo pricing strategies should give the same result for similar inputs (for a large number of MC simulations). We have implemented the comparison as part of this exercise. From the sample run below for N=252 and M=1,000,000, we observe that the monte carlo price is within **0.094%** of the black scholes price.

```
Choose Mission: 3
--------------------------
GOAL: Compare Black-Scholes and Monte Carlo Simulation in Pricing European Call Option
Please enter S, K, T, r, vol, N and M (space delimited - order matters): 20 18 1 0.1 0.25 252 1000000
Inputs : S=20, K=18, T=1, r=0.1 , vol=0.25, N=252, M=1000000
Black-Scholes Price = 4.22207
Monte Carlo Price = 4.22604
Difference: 0.00397545 = 0.0941588%
--------------------------
```

4.1.4. Asian Option: Monte Carlo Pricing

An Asian option, also known as average option, is an option whose value is contingent on the average value of the underlying stock throughout the life of the option. Thus, the price of this option is not a Markov process as it relies not only on the immediate previous price but the entire series of prices throughout the life of the option. We use Monte Carlo simulations to generate M possible security price paths, calculate the average security price for each path and then calculate the value of the option as $C_i = S_{avg} - K$ where K is the strike price. The expected value for all the M paths will give the price of the Asian call option. See *TaskManager#asianCallOptionPriceMonteCarlo()* for implementation details. Here is a sample run and result:

```
Choose Mission: 4
--------------------------
GOAL: Price Asian Option using Monte Carlo Simulation
Please enter S, K, T, r, vol, N and M (space delimited - order matters): 20 18 1 0.1 0.25 252 1000000
Inputs : S=20, K=18, T=1, r=0.1 , vol=0.25, N=252, M=1000000
Price 2.91361
--------------------------
```

4.2. Interest Rate (IR) Models and Zero Coupon Bond (ZCB) Pricing

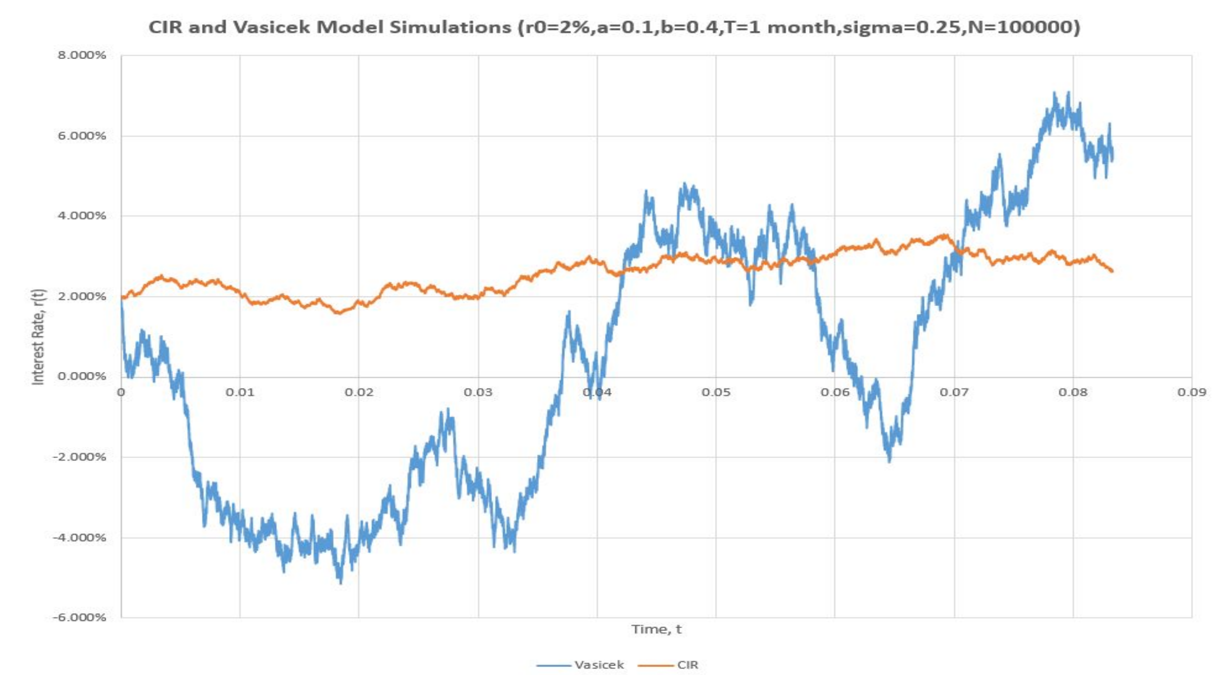
4.2.1. Ornstein-Uhlenbeck IR Model and Cox-Ingersoll-Ross (CIR) IR Model

In this section we explore two stochastic interest rate models namely: OU and CIR. These two models are similar except that in the CIR model, the stochastic term is scaled by the square root of the previous interest rate multiplied by volatility of the rate. This extra term ensures that the volatility of the process is small for small previous rates. This, in comparison, we expect the OU model to exhibit more volatile behavior. Also, by choosing $2ab \geq \sigma^2$, one can ensure non-negative values for the CIR model. Here is a sample run and result plotted in excel. We indeed observe that for both models, the price seems exhibit mean reversion i.e.

it oscillates around r_0, and that the OU model exhibits more volatile behavior compared to CIR model.

```
Choose Mission: 5
--------------------------
GOAL: Generate Ornstein-Uhlenbeck interest rate process
Please enter a, b, r0, vol, T and N (space delimited - order matters): 0.1 0.4 0.02 0.25 0.83333 1000000
Inputs :: a=0.1, b=0.4, r0=0.02, vol=0.25, T=0.83333, N=1000000
Output: See vasicek.xls at the base of directory.
--------------------------
To see the options again, enter 1. Otherwise enter 0: 0
Choose Mission: 6
--------------------------
GOAL: Generate Cox-Ingersol-Ross interest rate process
Please enter a, b, r0, vol, T and N (space delimited - order matters): 0.1 0.4 0.02 0.25 0.83333 1000000
Inputs :: a=0.1, b=0.4, r0=0.02, vol=0.25, T=0.83333, N=1000000
Output: See cir.xls at the base of directory.
--------------------------
```

Graph of OU (Vasicek) and CIR models.



Mean reversion i.e the price oscillates about an axis

4.2.2. Pricing ZCB using Monte Carlo and CIR IR model

The interest rate models above can be utilized in pricing coupons. In particular, we will price a zero coupon bond (ZCB) in this exercise using the tools we have developed so far. We use the Monte Carlo simulation methodology to generate thousands of paths for the price process and use the CIR model to simulate the stochastic interest rate processes for each path. The price of a ZCB is given as $ZCB = E[e^{-\int_0^t r(s)ds}]$. In the MC simulation, we achieve this by calculating the average of $exp^{sum\ of\ Interest\ Rates\ to\ T}$ for all paths. Here is a sample result:

```
Choose Mission: 4
--------------------------
GOAL: Price Asian Option using Monte Carlo Simulation
Please enter S, K, T, r, vol, N and M (space delimited - order matters): 20 18 1 0.1 0.25 252 1000000
Inputs : S=20, K=18, T=1, r=0.1 , vol=0.25, N=252, M=1000000
Price 2.91361
--------------------------
```

Thanks.