MATH 6205 - Numerical Methods for Financial Derivatives
Fall 2018

Purpose:

The objective of this Python program is to simulate the simulate the path of Stock price using Geometric Brownian Motion as the underlying diffusion process for the Stock price. Since Weiner process is a continuous process, we used Euler discretization and Milstein discretization to discretize it and used it to simulate. Monte Carlo simulations is used to calculate the European call and put option values. The error is calculated between the Monte Carlo price and the Black-Scholes price (analytical solution). The input parameters required are provided by the user.

Numerical Methods:

Geometric Brownian Motion is used for the simulation of sample paths. Euler discretization and Milstein discretization are used as a discretization method to discretize the SDE of the stock process. Monte Carlo simulations and Black Scholes formulae is used to calculate the option values.

Output:

```
In [232]: runfile('/Users/khan/Desktop/Sem-III/MATH 6204/Lectures/main.py',
wdir='/Users/khan/Desktop/Sem-III/MATH 6204/Lectures')
Reloaded modules: hw3_Mohammed_Ameer_Khan

Monte Carlo and Black Scholes option values for European Call

Time step  MC_value    BSM_Value   Abs diff Disc method

0.010      27.127033   28.684884   1.557851   Euler
0.001      27.311426   28.684884   1.373458   Euler
0.010      29.691474   28.684884   1.006590  Milstein
0.001      27.091688   28.684884   1.593195  Milstein

Monte Carlo and Black Scholes option values for European Put

Time step  MC_value    BSM_Value   Abs diff Disc method

0.010      28.061125   28.198446   0.137321   Euler
0.001      27.809815   28.198446   0.388631   Euler
0.010      28.784868   28.198446   0.586422  Milstein
0.001      27.052469   28.198446   1.145977  Milstein

In [233]:
```

Analysis:

The Monte Carlo simulated option value has to be converged with the option value generated by the Black Scholes closed form solution as the number of simulations are increased because of the law of large numbers. We have fixed our number of simulations to 1000 for this assignment and we are trying to increase the time steps rather than the number of simulations.

We have used Euler and Milstein discretization methods to discretize the stock price SDE. Milstein has second order terms but Euler has single order terms only. As we can see from the output, as we have increased the time steps from 100 to 1,000, there is a decrease in the absolute error between the Monte Carlo generated call value and the Black Scholes call value. But we don't see this phenomenon for Put values. This might due to the unexplained errors generated when we increase the mesh size i.e. the number of time steps.

Since Milstein discretization method has second order terms, it should explain the variation better and the convergence values should improve as we increase the number of simulations. But we kept our number of simulations fixed and increased the number of time steps. The absolute error between Monte Carlo generated call values and Black Scholes call value increases as we increased the number of time steps. The phenomenon is same for put values as well. This might due to the unexplained errors generated when we increase the mesh size i.e. the number of time steps