FRE 6951

Problem Set 2

Due: April 26, 2019

1) Write a program to run simulations for the exponential distribution on the CPU using multi-threading. You need to run on at least 2 threads. Build the model and run with a total of 1 million independent simulations. Calculate the mean and variance of the simulated distribution, and the standard error for the estimate of the mean. Plot the cumulative distribution for the simulated random numbers and compare it to a plot of the actual cumulative distribution function. For the distribution function, I recommend writing the simulated numbers to a text file so that you can use either Excel or Python to do the plots. Note: The distribution depends on the value that you set for the mean parameter, μ, in the exponential distribution. Simulate the exponential by simulating the uniform $U(0,1)$ and inverting the exponential distribution function.

   Exponential distribution function: $F(x) = 1 - \exp\left({-x}/{\mu}\right)$ for $x \geq 0$

   Exponential density function: $f(x) = \exp\left({-x}/{\mu}\right)$

2) Repeat problem number #1, but run the simulation on a GPU.

3) Consider the pricing for a barrier knock out option on a stock price, in which the option expiration is European (exercise only at expiration) and the knock out feature is in effect at any time prior to expiration. Write a program to run a Monte Carlo simulation of the Black-Scholes model to value this barrier option. The model needs to have at least daily time steps for an option that expires in 1 year (365 days). Write the program so that the simulation paths are calculated on a GPU. Set up your code so that you can input the interest rate, the stock price volatility, the initial level of the stock price, call or put, the strike price, and the knock out price (barrier price). (a) Value a call option with a knock out at a price above the strike. (b) Value a put option with a knock out at a price below the strike.

For this problem set, turn in your source code and results. Include a plot of the distribution function in #1. Where necessary, you can provide screen shots (screen shots of the execution in the Console).