Lecture 3

- Topics for today:

    - Examples of parallel processing on CPU processors and GPU's
    - Examples - Financial models with Monte Carlo solutions

- Reading assignments for this lecture

    Cuda documentation for CURAND Library
    Hull, 9^{th} Edition, Chps. 31-32

- Topic for next week: parallel processing for finite difference methods Lecture 4
    - See recommended readings in course syllabus

- Problem Set 2 due in 1 week, and the project proposal is due next week.

3 Parallel Processing to Solve Financial Models with Monte Carlo Simulation

3.1 Multi-threading and code for running pseudo random number generators

- Finish code examples from Lecture 2

- See the Test_Curand_device project/folder, which is an example from Nvidia for simulating uniform random variables on the GPU using the Curand library. → Curand documentation

- See the Test_MRG32k3a_GPU project/folder, which has a program to simulate uniform random variables on the GPU. Uses the step ahead algorithm to set seeds, and Curand

- See the Test_MRG32k3a_Normal_GPU project/folder, which has a program to simulate normal random variables for a stock price process on the GPU. Uses the step ahead algorithm to set seeds, and Curand.

- Go to Visual Studio examples →

3 Parallel Processing to Solve Financial Models with Monte Carlo Simulation (cont.)

3.2 Simulating diffusion processes

- Start with the diffusion in the Black-Scholes model: $dS = rS\,dt + \sigma S\,dz$
- We normally transform to a stochastic differential for $\log S$: $d\log S = (r - ½\,\sigma^2)\,dt + \sigma\,dz$
- The solutions for $\log S(t)$ and $S(t)$

$$\log S(t) = \log S(0) + (r - ½\,\sigma^2)t + \sigma \int_0^t dz$$

$$S(t) = S(0)\exp\left((r - ½\,\sigma^2)t + \sigma \int_0^t dz\right)$$

- The integral is an integral of a Brownian motion and we know that the integral has a normal distribution with mean zero and variance equal to *t*. The stock price has a lognormal distribution. Simulating a normal random variable with this formula produces an exact simulation of the model, and there is no need to simulate over small time steps in this model. One can simulate one lognormal to get the stock price at expiration.

3.2 Simulating diffusion processes (cont.)

- Variations on the Black-Scholes model: consider a model in which r and σ^2 vary deterministically over time. r is taken from the initial term structure of interest rate.

$$d \log S = (r(t) - ½\, \sigma^2(t))\, dt + \sigma(t)\, dz$$

- The solutions for $\log S(t)$

$$\log S(t) = \log S(0) + \int_0^t \left(r(s) - ½\, \sigma^2(s)\right) ds + \int_0^t \sigma(s) dz(s)$$

- The integrals for r and σ^2 can be calculated so that this model can be simulated like the standard Black-Scholes model

- Stochastic volatility is a different matter as the log of the stock price and the volatility must be simulated over small time steps. In many cases, stochastic volatility models are applied with negative correlation between the stock price and the stochastic variance.

3.2 Simulating diffusion processes (cont.)

- **Stochastic Volatility Models**

$$d\log S = (r(t) - ½v(t))\,dt + \sqrt{v(t)}\,dz_1$$

$$dv = \mu(t,v)dt + \sigma(t,v)dz_2$$

- dz_1 and dz_2 must be simulated with correlation over small time steps. Warnings: SV models with positive correlation between the stock price and the stochastic variance can be unstable. FX rates and stochastic volatility? Consider first, a lognormal process for the stochastic variance rate. $v(t) = \exp(y(t))$ or $\log v(t) = y(t)$

$$dy = \kappa(\theta - y)dt + \sigma\,dz_2$$

- Here y is an Ornstein-Uhlenbeck process. One can apply the Euler approximation to the diffusion equation.

$$y(t+\Delta t) - y(t) \equiv \Delta y = \kappa\big(\theta - y(t)\big)\,\Delta t + \sigma\Delta z_2$$

- There are other approximation methods for diffusion equations. For this process, one can solve to get the integral solution to the stochastic differential equation. This provides an exact representation of the process over discrete time periods.

3.2 Simulating diffusion processes (cont.)

- **Stochastic Volatility Models (cont.)**
- The Ornstein-Uhlenbeck process: define $f(t, y) = e^{\kappa t} y(t)$ and apply Ito's lemma

$$
\begin{aligned}
df &= \kappa e^{\kappa t} y\, dt + e^{\kappa t} dy \\
&= \kappa e^{\kappa t} y\, dt + e^{\kappa t}(\kappa(\theta - y)dt + \sigma\, dz_2) \\
&= e^{\kappa t} \kappa\theta\, dt + e^{\kappa t} \sigma\, dz_2
\end{aligned}
$$

- The integral solution

$$e^{\kappa t} y(t) - y(0) = \int_0^t e^{\kappa s} \kappa\theta\, ds + \int_0^t e^{\kappa s} \sigma\, dz_2(s)$$

- Multiply by $e^{-\kappa}$

$$y(t) = e^{-\kappa t} y(0) + \int_0^t e^{-\kappa(t-s)} \kappa\theta\, ds + \int_0^t e^{-\kappa(t-s)} \sigma\, dz_2(s)$$

3.2 Simulating diffusion processes (cont.)

- **Stochastic Volatility Models, Ornstein-Uhlenbeck processes (cont.)**
- Solve the integral containing the long run mean

$$y(t) = e^{-\kappa t}y(0) + \theta(1 - e^{-\kappa}) + \int_0^t e^{-\kappa(t-s)}\sigma \, dz_2(s)$$

- The integral containing the Brownian motion is normally distributed with mean 0 and the following variance:

$$\text{Var}\left(\sigma \int_0^t e^{\kappa(s-t)} dz_2(s)\right) = \sigma^2 \int_0^t e^{2\kappa(s-t)} ds = \sigma^2 \left(\frac{1 - e^{-2\kappa t}}{2\kappa}\right)$$

- This model can be used to simulate the Ornstein Uhlenbeck process over discrete time periods. The SV model requires simulation over small time steps. Note: for small time steps, the variance is converging to $\sigma^2 \Delta t$.

3.2 Simulating diffusion processes (cont.)

- **Stochastic Volatility Models, Ornstein-Uhlenbeck processes (cont.)**

- Simulating this SV model over small time steps with Δz_1 and Δz_2 simulated as correlated standard normal random variables

$$\Delta \log S = \log S(t+\Delta t) - \log S(t) = (r(t) - ½v(t))\,\Delta t + \sqrt{v(t)}\,\Delta z_1$$

$$v(t) = e^{y(t)}$$

$$y(t+\Delta t) = e^{-\kappa\Delta t}y(t) + \theta\left(1 - e^{-\kappa\Delta t}\right) + \sigma\sqrt{\frac{1 - e^{-2\kappa\Delta t}}{2\kappa}}\,\Delta z_2$$

- The easiest method for simulating correlated random variables here is to first simulate Δz_2 and then simulate a 2nd independent standard normal Δz and calculate $\Delta z_1 = \rho\,\Delta z_2 + \sqrt{1-\rho^2}\,\Delta z$. For stock prices, ρ is typically negative. This is a SV model with lognormal volatility.

- The Heston SV model has the same process for the log of the stock price and uses the following stochastic variance process.

$$dv = \kappa(\theta - v)dt + \sigma\sqrt{v}\,dz_2$$

3.2 Simulating diffusion processes (cont.)

- **Stochastic Volatility Models, Heston square root processes (cont.)**

- The stochastic variance in the Heston model has the following solution over discrete time periods.

$$v(t+\Delta t) = e^{-\kappa\Delta t}v(t) + \theta\left(1 - e^{-\kappa\Delta t}\right) + \sqrt{\mathrm{Var}(\Delta t, v(t))}\,\Delta z_2$$

- with

$$\mathrm{Var}\big(\Delta t, v(t)\big) = \sigma^2\left(\frac{1 - e^{-\kappa\Delta t}}{\kappa}\right)\left(e^{-\kappa\Delta t}v(t) \;+\; \tfrac{1}{2}\theta\,\left(1 - e^{-\kappa\Delta t}\right)\right)$$

- In this model, the stochastic variance, over discrete time periods, is distributed as a non-central chi squared. There are methods for simulating this distribution; the non-central chi squared distribution is a mixture of the chi squared distribution in which the degrees of freedom parameter have a Poisson distribution. In effect, one can simulate the degrees of freedom parameter and then simulate a chi squared to generate the non-central chi squared. If you are interested in running simulations for this model, I can provide the methodology for simulating this distribution.

Example: 3 Factor Hull-White Model

- 3 factor Hull-White model with normally distributed interest rates, and volatilities as deterministic functions of time. Model has a short rate that mean reverts to 2 stochastic mean factors, y_1 and y_2. The y_1 factor is set up with a slow rate of mean reversion and serves as the long run mean. The y_2 factor is set up with a zero mean and a slower rate of mean reversion and it is a curvature factor

$$dr = [\kappa_0(\theta_0(t) + y_1 + y_2 - r) - \lambda_0\sigma_0^2(t)]\,dt + \sigma_0(t)dz_0$$

$$dy_1 = [\kappa_1(\theta_1 - y_1) - \lambda_1\sigma_1^2(t)]\,dt + \sigma_1(t)dz_1$$

$$dy_2 = (-\kappa_2 y_2 - \lambda_2\sigma_2^2(t))\,dt + \sigma_2(t)dz_2$$

- r is the instantaneous interest rate, which we will proxy with the overnight rate.
- $\theta_0(t)$ is a deterministic function of time, which is used to ensure that the model can be calibrated to match the initial term structure of forward rates. We will start with $\theta_0(t)$ = 0 for all t. λ parameters are risk premia, and are set to 0 for this example.
- The model is a 3 factor version of the Vasicek-Hull-White model. The volatilities will be calibrated so that the model fits the current yield curve and either matches or comes close on market implied volatilities (ED futures options, caps/floors, swaptions). I use swaption volatilities and ED futures options in the example.

Example: 3 Factor Hull-White Model

- First, we can solve the model to get the discount bond pricing function.

$$D(0,T) = \hat{E}_0\left(\exp - \int_0^T r(s)ds\right)$$

- Because r is normally distributed, the integral (summation) of r is also normally distributed. The easiest method for solving the discount function is to solve the PDE for the pricing function.

$$0 = \frac{\partial D}{\partial t} + \frac{1}{2}\sigma_0^2(t)\frac{\partial^2 D}{\partial r^2} + \frac{1}{2}\sigma_1^2(t)\frac{\partial^2 D}{\partial y_1^2} + \frac{1}{2}\sigma_2^2(t)\frac{\partial^2 D}{\partial y_2^2} + \kappa_0(\theta_0(t) + y_1 + y_2 - r - \lambda_0\sigma_0^2(t))\frac{\partial D}{\partial r}$$

$$+ \kappa_1(\theta_1 - y_1 - \lambda_1\sigma_1^2(t))\frac{\partial D}{\partial y_1} + \kappa_2(-y_2 - \lambda_2\sigma_2^2(t))\frac{\partial D}{\partial y_2} - rD$$

- The solution is an exponential affine function:

$$D(t,T) = \exp(-A(t,T) - B_0(t,T)r(t) - B_1(t,T)y_1(t) - B_2(t,T)y_2(t))$$

Example: 3 Factor Hull-White Model

- Evaluate the partial derivatives using the proposed solution and plug into the PDE. Then organize the derivatives with respect to each one of the state variables and the constant. Start by picking up all of the terms in the PDE which include r:

$$0 \;=\; -\frac{\partial B_0}{\partial t} + \kappa_0 B_0(t,T) - 1$$

- Then collect the terms associated with y_1 and y_2, and the remaining terms which are associated with a constant (does not vary with any state variables)

y_1: $$0 \;=\; -\frac{\partial B_1}{\partial t} + \kappa_1 B_1(t,T) - \kappa_0 B_0(t,T)$$

y_1: $$0 \;=\; -\frac{\partial B_2}{\partial t} + \kappa_2 B_2(t,T) - \kappa_0 B_0(t,T)$$

Constant:

$$0 \;=\; -\frac{\partial A}{\partial t} + \tfrac{1}{2}\sigma_0^2(t)B_0^2(t,T) + \tfrac{1}{2}\sigma_1^2(t)B_1^2(t,T) + \tfrac{1}{2}\sigma_2^2(t)B_2^2(t,T)$$
$$(\lambda_0\sigma_0^2(t) - \kappa_0\theta_0(t))B_0(t,T) + (\lambda_1\sigma_1^2(t) - \kappa_1\theta_1)B_1(t,T) + \lambda_2\sigma_2^2(t)B_2(t,T)$$

Example: 3 Factor Hull-White Model

- These equations are ordinary differential equations (ODE) and they are easy to solve.

$$B_0(t,T) = \frac{1 - e^{-\kappa_0(T-t)}}{\kappa_0}$$

$$B_1(t,T) \ = \int_t^T e^{-\kappa_1(s-t)} \kappa_0 B_0(s,T)\, ds$$

$$B_1(t,T) = \frac{1 - e^{-\kappa_1(T-t)}}{\kappa_1} - \ e^{-\kappa_1(T-t)} \left(\frac{1 - e^{-(\kappa_0 - \kappa_1)(T-t)}}{\kappa_0 - \kappa_1} \right)$$

$$B_2(t,T) \ = \int_t^T e^{-\kappa_2(s-t)} \kappa_0 B_0(s,T)\, ds$$

$$B_2(t,T) = \frac{1 - e^{-\kappa_2(T-t)}}{\kappa_2} - \ e^{-\kappa_2(T-t)} \left(\frac{1 - e^{-(\kappa_0 - \kappa_2)(T-t)}}{\kappa_0 - \kappa_2} \right)$$

Example: 3 Factor Hull-White Model

- The constant term involves an integral which can be solved analytically. In the interest of time, I solve it numerically on a computer using an ODE solver.

$$A(t,T) = -\int_t^T \Big(\tfrac{1}{2}\sigma_0^2(s)B_0^2(s,T) + \tfrac{1}{2}\sigma_1^2(s)B_1^2(s,T) + \tfrac{1}{2}\sigma_2^2(s)B_2^2(s,T)$$
$$+ (\lambda_0\sigma_0^2(t) - \kappa_0\theta_0(t))B_0(s,T) + (\lambda_1\sigma_1^2(s) - \kappa_1\theta_1)B_1(s,T) + \lambda_2\sigma_2^2(t)B_2(s,T)\Big)\,ds$$

- You can verify that this solution works for the discount bond pricing function by evaluating the partial derivatives and plugging back into the PDE.
- For a model application, I simplify the model by setting the $\theta_0(t)$ = 0 and making all of the volatilities constants. The model will not provide an exact fit for the initial term structure. I will try to see how close I can get without $\theta_0(t)$. If you were managing an interest rate derivatives portfolio, you should incorporate the $\theta_0(t)$ term to produce the exact fit. Asset management firms would apply this model without the $\theta_0(t)$ term and look for relative value trading opportunities.
- The solution for the zero rates is a linear function in the state variables

$$R(0,T) = \tfrac{1}{T}\left(A(0,T) + B_0(0,T)\,r(0) + B_1(0,T)\,y_1(0) + B_2(0,T)\,y_2(0)\right)$$

Example: 3 Factor Hull-White Model

- I apply this model to OIS. And I treat 3m LIBOR as a deterministic spread over 3m OIS. From the model, 3m OIS is determined as follows

$$D(3m) = \exp(-A(3m) - B_0(3m)r(t) - B_1(3m)y_1(t) - B_2(3m)y_2(t))$$
$$= \frac{1}{1 + R_{3m}(t) \times \frac{Days\,(3m)}{360}} = \exp\left(-r_{3m}(t) \times \frac{Days\,(3m)}{365}\right)$$

- Here r_{3m} is the 3m zero rate for OIS, continuously compounded, and R_{3m} is the simple interest rate for 3m OIS. *T*-*t* for 3*m* will be close to 0.25. 3m LIBOR, the simple interest rate, is modeled as a fixed spread over 3m OIS.

$$R_{3mL}(t) = R_{3m}(t) + SPR_{3mL}$$

- Payoff function for calls on 3 month LIBOR: max[0, $R_{3mL}(t) - K$]

- Payoff function for puts on 3 month LIBOR: max[0, $K - R_{3mL}(t)$]

Example: 3 Factor Hull-White Model

- The simulation model for derivatives on 3 month LIBOR uses the following model for fixing the spread: $R_{3mL}(T) = R_{3m}(T)$ + Spread. I use the model to calculate the 3m OIS rate and calculate the spread between the actual market quote for 3m LIBOR and the model rate for 3m OIS. I do the same for all of the forward 3m LIBORs: calculate spread between forward 3m OIS and forward 3m LIBOR.

- First, I calibrate the model parameters to fit the initial OIS yield curve, and then I adjust the volatilities to get the model close on the valuation for a few at-the-money swaptions.

- I set up and run the model calculations for the discount bond coefficients in Excel. I use a C++ program to do the numerical integration for the constant terms, and to do the Monte Carlo simulations for the options. The C++ programs are compiled as dll's, which I can call from Excel.

- I have separate C++ programs to do the Monte Carlo simulations for Eurodollar futures options, using multiple processors on the CPU and a GPU.

Example: 3 Factor Hull-White Model

- Need to simulate the model, system of 3 diffusion equations, back on slide 10. The state variables, y_1 and y_2, are Ornstein Uhlenbeck processes so we will apply the methods for simulating Ornstein Uhlenbeck processes over small time intervals. We need am equation for dr.

$$y_1(t+\Delta t) = e^{-\kappa_1 \Delta t} y_1(t) + \left(\theta_1 - {\lambda_1 \sigma_1^2}/{\kappa_1}\right)\left(1 - e^{-\kappa_1 \Delta t}\right) + \sigma \sqrt{\frac{1 - e^{-2\kappa_1 \Delta t}}{2\kappa_1}}\, \Delta z_1$$

$$y_2(t+\Delta t) = e^{-\kappa_2 \Delta t} y_2(t) + \left({-\lambda_2 \sigma_2^2}/{\kappa_2}\right)\left(1 - e^{-\kappa_2 \Delta t}\right) + \sigma \sqrt{\frac{1 - e^{-2\kappa_2 \Delta t}}{2\kappa_2}}\, \Delta z_2$$

$$r(t+\Delta t) = \; e^{-\kappa_0 \Delta t} r(t) + \int_t^{t+\Delta t} e^{\kappa_0 (s-(t+\Delta t))} \left(\kappa_0\left(\theta_0(s) + y_1(s) + y_2(s)\right) - \lambda_0 \sigma_0^2(s)\right) ds$$

$$+ \int_t^{t+\Delta t} e^{\kappa_0 (s-(t+\Delta t))} \sigma_0 \, dz_0$$

- Go to code examples →