Alternative Models to Geometric Brownian Motion.

Monte Carlo Simulations.

1. Constant elasticity of variance (CEV)
2. Mixed Jump diffusion
3. Stochastic Volatility
4. Implied Volatility Function

CEV Model

$$dX_t = (r-q)X_t dt + \sigma X_t^\alpha dW_t$$

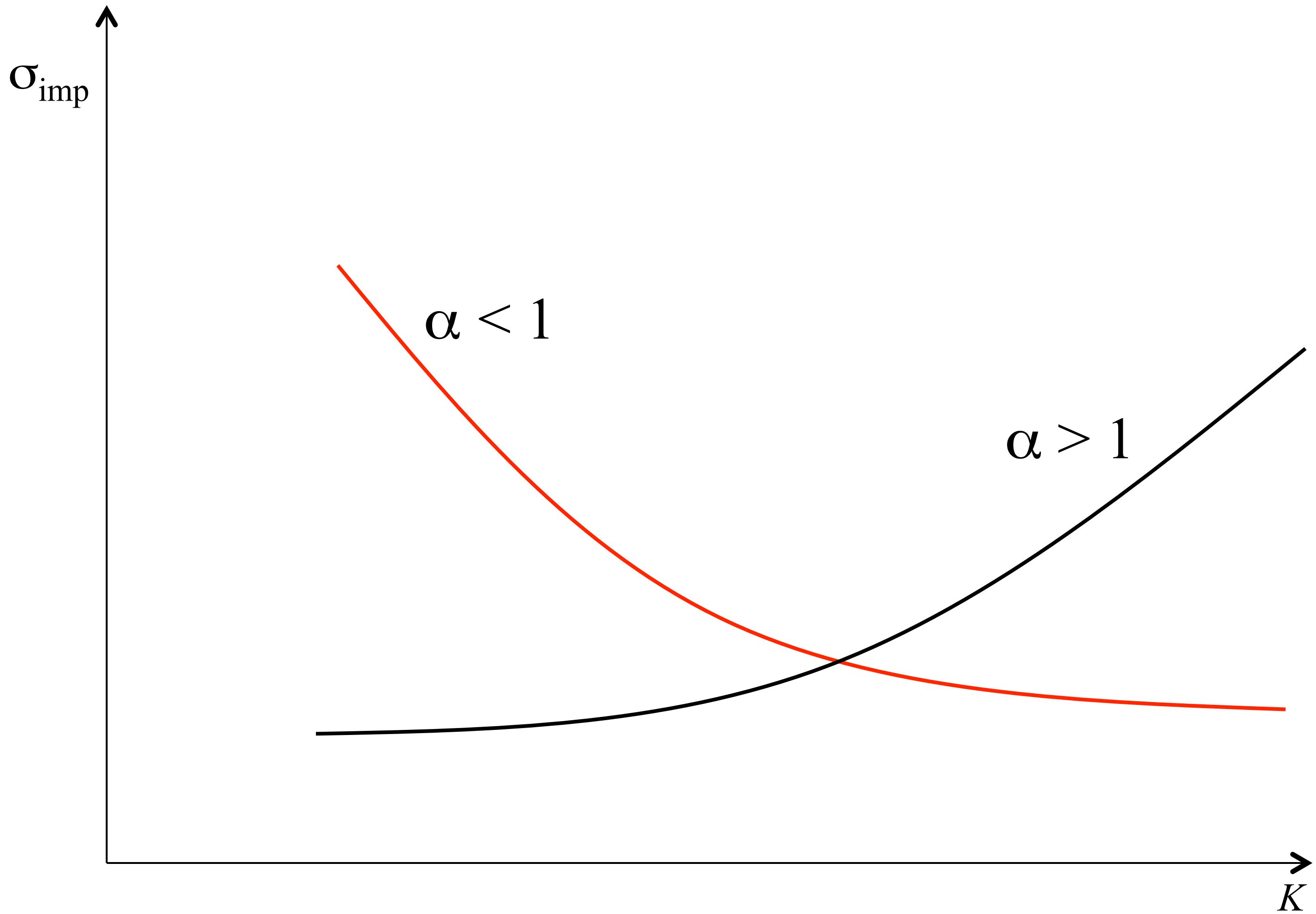
When α = 1 the model is Black-Scholes

When α > 1 volatility rises as stock price rises

When α < 1 volatility falls as stock price rises

European options can be value analytically in terms of the cumulative non-central chi square distribution

CEV Models Implied Volatilities



Mixed Jump Diffusion Model

Robert Merton produced a pricing formula when the asset price follows a diffusion process overlaid with random jumps

$$dX_t = (r - q - \lambda k)X_t dt + \sigma X_t dW_t + X_t dp_t$$

dp_t is the random jump (size of the jump is drawn from some distribution for example log normal)

k is the expected size of the jump $\lambda\, dt$ is the probability that a jump occurs in the next interval of length dt

λk is the expected return from jumps

Simulating a Jump Process

In each time step

- Sample Poisson process to determine the number of jumps
- Sample to determine the size of each jump

Jumps and the Smile

Jumps have a big effect on the implied volatility of short term options

They have a much smaller effect on the implied volatility of long term options

Time Varying Volatility

The variance rate substituted into BSM should be the average variance rate

- Suppose the volatility is σ_1 for the first year and σ_2 for the second and third
- Total accumulated variance at the end of three years is $\sigma_1{}^2 + 2\sigma_2{}^2$

The 3-year average volatility is given by

$$3\bar{\sigma}^2 = \sigma_1^2 + 2\sigma_2^2; \quad \bar{\sigma} = \sqrt{\frac{\sigma_1^2 + 2\sigma_2^2}{3}}$$

Mean Reverting Ornstein-Uhlenbeck Process

$$dX_t = \theta\,(\mu - X_t)dt + \sigma\, dW_t$$

$$X_{t_n} = X_{t_{n-1}} + \theta(\mu - X_{t_{n-1}})\Delta t + \sigma\varepsilon_n\sqrt{\Delta t}$$

Let **X_0 = 0**, the initial value of the process.
Let $\theta=1$, $\mu=0$, $\sigma=1$
N = 250, number of time steps.
T = 1, the final time measured in years
Δt = T/N = 0.004 is a size of a time step.
n is an index of a timestep **t_n = n T/N**, $0 \le n \le N$

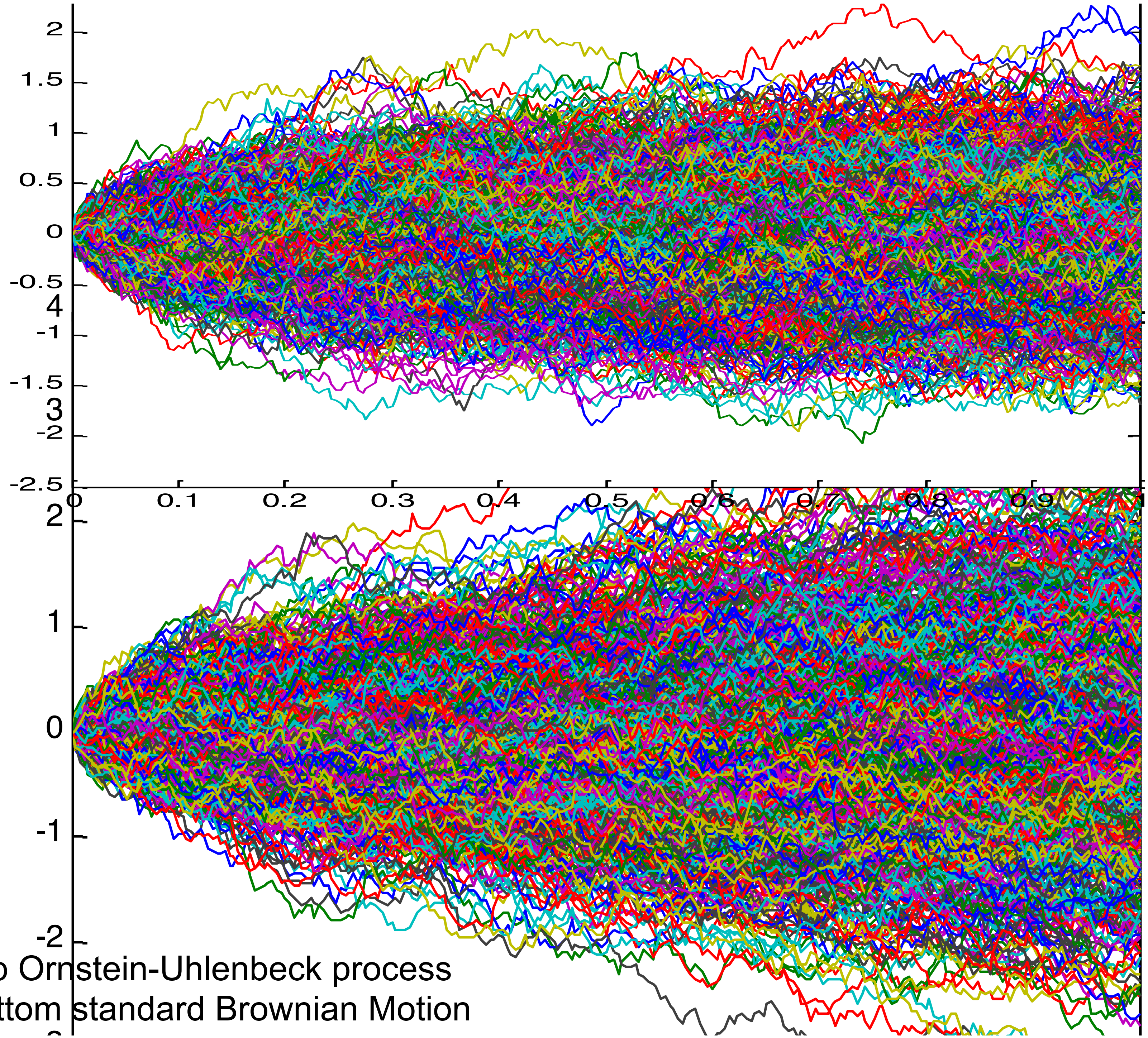
$$dX_t = -X_t\, dt + dW_t$$

$$X_{t_n} = X_{t_{n-1}} - X_{t_{n-1}}\Delta t + \varepsilon_n\sqrt{\Delta t}$$

```matlab
M=1000;     %number of trajectories
N=250;      %Number of steps in one trajectory
X0=0;       %initial point
T=1;        %Final Time in years in trajectory
dt=T/N;     %time step
Sqrtdt=sqrt(dt);
Sigma=1; q=1;       %q is the same as theta, m same as mu
m=0; %level of Ornstein-Uhlenbeck process to which reverts
%X(j,:) j-th trajectory of Ornstein-Uhlenbeck process
%            dXt=q*(m-Xt)*dt+Sigma*dWt
X(1:M,1)=X0;      % Initial value  X(j,1)=X0 for all j=1:M
  %in Matlab array index starts with 1 and not 0 as in C++

for j=1:M  %generate M traject.of Ornstein-Uhlenbeck proces
 for i = 2:N+1  %generate j-th trajectory
  X(j,i)=X(j,i-1) + q*(m-X(j,i-1))*dt +Sigma*randn*Sqrtdt;
 end
end

t=0:dt:T;          %creating time array for plotting
plot(t,X(:,:));  %Plotting graph of trajectories
```



Top Ornstein-Uhlenbeck process
Bottom standard Brownian Motion

For mean-reverting Ornstein-Uhlenbeck process with parameters μ, σ, θ that starts at Xo at t=0 the probability distribution at time t is

$$p(x,t) = \frac{\sqrt{2\theta}}{\sqrt{2\pi}\sigma\sqrt{(1-e^{-2\theta t})}} e^{-\frac{(x-\mu-(x_0-\mu)e^{-\theta t})^2}{2\sigma^2(1-e^{-2\theta t})/(2\theta)}}$$

Which is normal distribution with mean $\mu + (x_0 - \mu)e^{-\theta t}$

and standard deviation $\sigma\sqrt{(1-e^{-2\theta t})/(2\theta)}$

As $t \rightarrow +\infty$ this distribution stabilizes and becomes

$$p_{stable}(x) = \frac{\sqrt{2\theta}}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2/(2\theta)}}$$

In mean-reverting Ornstein-Uhlenbeck process standard deviation is not growing as $\sigma\sqrt{t}$ but as

$$\sigma\sqrt{(1-e^{-2\theta\, t})/(2\theta)}$$

So as $t \to +\infty$ standard deviation stabilizes at $\sigma / \sqrt{2\theta}$

and mean $\mu + (x_0 - \mu)e^{-\theta\, t}$ stabilizes at μ

$$p_{stable}(x) = \frac{\sqrt{2\theta}}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2/(2\theta)}}$$

Stochastic Volatility Models

$$dX_t = (r - q)X_t dt + \sqrt{V_t} X_t dW_{1t}$$

$$dV_t = a(V_L - V_t)dt + bV_t^{\alpha} dW_{2t}$$

V_t is the variance of the first process that is driven by the second process

V_L is a long term average

W_{1t} and W_{2t} are two correlated Brownian motions with correlation ρ

When V_t and X_t are uncorrelated a European option price is the Black-Scholes-Merton price integrated over the distribution of the average variance rate

$$X_n - X_{n-1} = (r-q)X_{n-1}\Delta t + \sqrt{V_{n-1}}X_{n-1} \quad e_{1n}\sqrt{\Delta t}$$

$$V_n - V_{n-1} = a(V_L - V_{n-1})\Delta t + bV_{n-1}^{\alpha} \quad e_{2n}\sqrt{\Delta t}$$

We can set initial values and parameters for example:

$$X_0 = 50, V_0 = 0.09, r = 0.02, q = 0.01$$

$$a = 0.1, V_L = 0.06, \alpha = 0.9, b = 0.15$$

To model correlated Brownian motions with correlation ρ we obtain independent normal samples ε_{1n} and ε_{2n} and set

$$e_{1n} = \varepsilon_{1n}$$

$$e_{2n} = \rho\varepsilon_{1n} + \varepsilon_{2n}\sqrt{1-\rho^2}$$

To Obtain Normal Samples

In Excel =NORMSINV(RAND()) gives a random sample from N(0,1)

In matlab randn gives such sample

To Obtain 2 Correlated Normal Samples

Obtain independent normal samples ε_1 and ε_2 and set

$$e_1 = \varepsilon_1$$

$$e_2 = \rho\varepsilon_1 + \varepsilon_2\sqrt{1-\rho^2}$$

There is a procedure known as Cholesky's decomposition when samples are required from more than two normal variables

Stochastic Volatility Models

When V and X are negatively correlated we obtain a downward sloping volatility skew similar to that observed in the market for equities

When V and X are positively correlated the skew is upward sloping. (This pattern is sometimes observed for commodities)

The IVF Model

The implied volatility function model is designed to create a process for the asset price that exactly matches observed option prices. The usual geomeric Brownian motion model

$$dX_t = (r - q)X_t dt + \sigma X_t dW$$

is replaced by

$$dX_t = [r(t) - q(t)]X_t dt + \sigma(X, t)X_t dW_t$$

The Volatility Function

The volatility function that leads to the model matching all European option prices is

$$[\sigma(K,t)]^2 =$$

$$2\frac{\partial c_{mkt}/\partial t + q(t)c_{mkt} + K[r(t)-q(t)]\partial c_{mkt}/\partial K}{K^2(\partial^2 c_{mkt}/\partial K^2)}$$

Strengths and Weaknesses of the IVF Model

The model matches the probability distribution of asset prices assumed by the market at each future time

The models does not necessarily get the joint probability distribution of asset prices at two or more times correct

Monte Carlo Simulation and Options

When used to value European stock options, Monte Carlo simulation involves the following steps:

1. Simulate 1 path for the stock price in a risk neutral world
2. Calculate the payoff from the stock option
3. Repeat steps 1 and 2 many times to get many sample payoffs
4. Calculate mean payoff
5. Discount mean payoff at risk free rate to get an estimate of the value of the option

Sampling Stock Price Movements

In a risk neutral world the process for a stock price is

$$dX_t = (r - q)X_t \, dt + \sigma X_t \, dW_t$$

We can simulate a path by choosing time steps of length Δt and using the discrete version of this

$$X_n - X_{n-1} = (r - q)X_{n-1} \, \Delta t + \sigma X_{n-1} \, \varepsilon_n \sqrt{\Delta t}$$

where ε_n is a random sample from $\mathrm{N}(0,1)$

A More Accurate Approach

Use

$$d \ \ln Xt = \left(r - q - \sigma^{2}/2\right) dt + \sigma \ dWt$$

The discrete version of this is

$$\ln X(t + \Delta t) - \ln X(t) = \left(r - q - \sigma^{2}/2\right) \Delta t + \sigma\varepsilon \sqrt{\Delta t}$$

or

$$\mathrm{X}(t + \Delta t) = X(t) \ e^{\left(r - q - \sigma^{2}/2\right) \Delta t + \sigma \varepsilon \sqrt{\Delta t}}$$

Extensions

When a derivative depends on several underlying variables we can simulate paths for each of them in a risk-neutral world to calculate the values for the derivative

Standard Errors in Monte Carlo Simulation

The standard error of the estimate of the option price is the standard deviation of the discounted payoffs given by the simulation trials divided by the square root of the number of observations.

Application of Monte Carlo Simulation

Monte Carlo simulation can deal with path dependent options, options dependent on several underlying state variables, and options with complex payoffs

It cannot easily deal with American-style options

Determining Greek Letters

For Δ:

1. Make a small change to asset price
2. Carry out the simulation again using the same random number streams
3. Estimate Δ as the change in the option price divided by the change in the asset price

Proceed in a similar manner for other Greek letters

Variance Reduction Techniques

1. Antithetic variable technique (make a second trajectory changing sign of normal samples in trajectory and average the value of derivative over 2 trajectories)
2. Control variate technique
3. Importance sampling
4. Stratified sampling
5. Moment matching
6. Using quasi-random sequences

Control Variate Technique

Value of option A in simulation f_A

Value of option B in the same simulation with same random samples f_B

Value option B using known analytic solution,

$f_{B\ analytic}$

Option A price $= f_A + (f_{B\ analytic} - f_B)$