Introduction to Monte Carlo in Finance

4 - Beyond Black and Scholes

Giovanni Della Lunga

WORKSHOP IN QUANTITATIVE FINANCE

Bologna - May 2-3, 2019

Outline

1. Beyond Black and Scholes
   - Square-Root Diffusion: the CIR Model
   - The Heston Model
2. The Hull and White Model

Subsection 1

Square-Root Diffusion: the CIR Model

CIR Model

- In this section, we consider the stochastic short rate model MCIR85 of Cox- Ingersoll-Ross which is given by the SDE:

$$\boxed{dr_t = \kappa_r(\theta_r - r_t)dt + \sigma_r\sqrt{r_t}dZ_t} \tag{1}$$

- To simulate the short rate model, it has to be discretized. To this end, we divide the given time interval $[0, T]$ in equidistant sub-intervals of length t such that now $t \in \{0, \Delta t, 2\Delta t, \ldots, T\}$, i.e. there are $M + 1$ points in time with $M = T/t$.
- The exact transition law of the square-root diffusion is known. Consider the general square- root diffusion process

$$dx_t = \kappa(\theta - x_t)dt + \sigma\sqrt{x_t}dZ_t \tag{2}$$

CIR Model

- It can be show that x_t, given x_s with $s = t - \Delta t$, is distributed according to

$$x_t = \frac{\sigma^2(1 - e^{-\kappa \Delta t})}{4\kappa} \chi_d'^2 \left(\frac{4^{-\kappa \Delta t}}{\sigma^2(1 - e^{-\kappa \Delta t})} x_s \right)$$

where $\chi_d'^2$ denotes a non-central chi-squared random variable with

$$d = \frac{4\theta\kappa}{\sigma^2}$$

degrees of freedom and non-centrality parameter

$$l = \frac{4^{-\kappa \Delta t}}{\sigma^2(1 - e^{-\kappa \Delta t})} x_s$$

CIR Model

- For implementation purposes, it may be convenient to sample a chi-squared random variable χ_d^2 instead of a non-central chi-squared one, $\chi_d'^2$.
- If $d > 1$, the following relationship holds true

$$\chi_d'^2(l) = (z + \sqrt{l})^2 + \chi_{d-1}^2$$

  where z is an independent standard normally distributed random variable.
- Similarly, if $d \leq 1$, one has

$$\chi_d'^2(l) = \chi_{d+2N}^2$$

  where N is now a Poisson-distributed random variable with intensity $l/2$. For an algorithmic representation of this simulation scheme refer to Glasserman, p. 124.

CIR Model

- In the next slide the function which generate paths according to the CIR models without approximations.
- This function returns a NumPy array containing the simulated paths.
- The input parameters are:
  - x0: (float) initial value
  - kappa: (float) mean-reversion factor
  - theta: (float) long-run mean
  - sigma: (float) volatility factor
  - T: (float) final date/time horizon
  - M: (int) number of time steps
  - I: (int) number of paths

CIR Model

```
def CIR_generate_paths_exact(x0, kappa, theta, sigma, T, M, I):

    dt     = T / M
    x      = np.zeros((M + 1, I), dtype=np.float)
    x[0]   = x0
    xh     = np.zeros_like(x)
    xh[0]  = x0
    ran    = np.random.standard_normal((M + 1, I))

    d = 4 * kappa * theta / sigma ** 2
    c = (sigma ** 2 * (1 - math.exp(-kappa * dt))) / (4 * kappa)
    if d > 1:
        for t in xrange(1, M + 1):
            l = x[t - 1] * math.exp(-kappa * dt) / c
            chi = np.random.chisquare(d - 1, I)
            x[t] = c * ((ran[t] + np.sqrt(l)) ** 2 + chi)
    else:
        for t in xrange(1, M + 1):
            l = x[t - 1] * math.exp(-kappa * dt) / c
            N = np.random.poisson(l / 2, I)
            chi = np.random.chisquare(d + 2 * N, I)
            x[t] = c * chi
    return x
```

CIR Model

- The exactness comes along with a relatively high computational burden which may, however, be justified by higher accuracy due to faster convergence. Although the computational burden per simulated value of xt may be quite high with the exact scheme, the possible reduction in time steps and simulation paths may more than compensate for this.
- We also consider an Euler discretization of the square-root diffusion, a possible discretization is given by

$$x_t = \left| x_s + \kappa(\theta - x_s)\Delta t + \sigma\sqrt{x_s}\sqrt{\Delta t} z_t \right|$$

  with z_t standard normal.
- While x_t cannot reach zero with the exact scheme if the Feller condition $2\kappa\theta > \sigma^2$ is met, this is not the case with the Euler scheme.
- Therefore, we take the absolute value on the right hand side for x_t.

CIR Model

```
def CIR_generate_paths_approx(x0, kappa, theta, sigma, T, M, I):

    dt     = T / M
    x      = np.zeros((M + 1, I), dtype=np.float)
    x[0]   = x0
    xh     = np.zeros_like(x)
    xh[0] = x0
    ran    = np.random.standard_normal((M + 1, I))

    for t in xrange(1, M + 1):
        xh[t] = (xh[t - 1] + kappa * (theta - np.maximum(0, xh[t - 1]))
                        * dt + np.sqrt(np.maximum(0, xh[t - 1]))
                        * sigma * ran[t] * math.sqrt(dt))
        x[t] = np.maximum(0, xh[t])
    return x
```

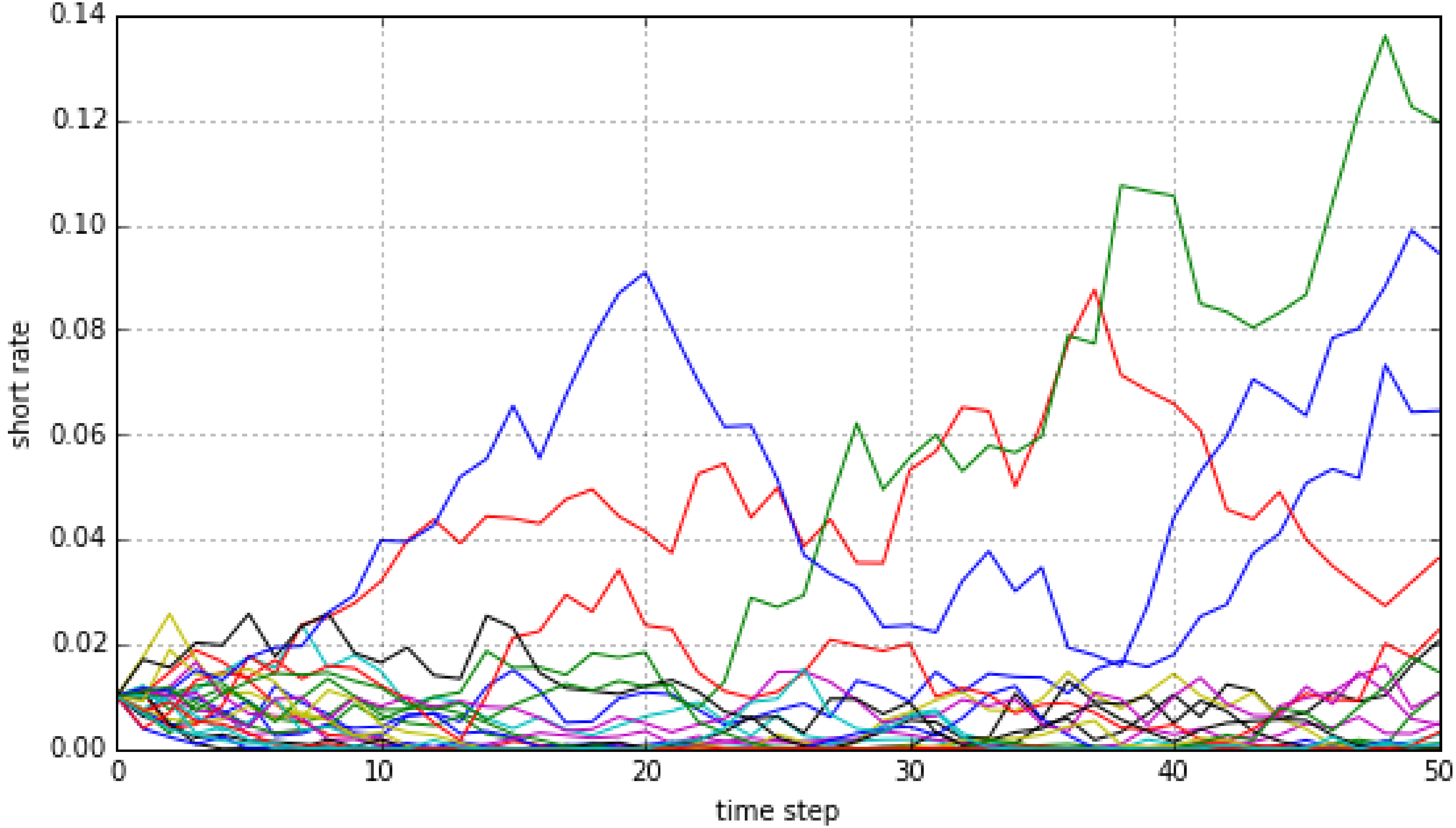
CIR Model

Let's generate some paths...

```
r0, kappa_r, theta_r, sigma_r = [0.01, 0.1, 0.03, 0.2]
T  = 2.0   # time horizon
M  = 50    # time steps
dt = T / M
I  = 50000   # number of MCS paths
np.random.seed(50000)  # seed for RNG

r = CIR_generate_paths_exact(r0,kappa_r,theta_r,sigma_r,T,M,I)
```

CIR Model


short rate
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
10
20
30
40
50
time step

CIR Model: Pricing ZCB

- The present value of the ZCB in the CIR model takes the form:

$$B_0(T) = b_1(T)e^{-b_2(T)r_0}$$

where

$$b_1(T) = \left[\frac{2\gamma\, exp((\kappa_r+\gamma)T/2)}{2\gamma+(\kappa_r+\gamma)(e^{\gamma T}-1)}\right]^{\frac{2\kappa_r\theta_r}{\sigma_r^2}}$$

$$b_2(T) = \frac{2(e^{\gamma T}-1)}{2\gamma+(\kappa_r+\gamma)(e^{\gamma T}-1)}$$

$$\gamma = \sqrt{\kappa_r^2+2\sigma_r^2}$$

- Now we simulate the CIR Model and derive MCS estimates for Zero-Coupon Bond (ZCB) at different points in time.

CIR Model: Pricing ZCB

- Since we know these value in closed form in the CIR Model, we have a natural benchmark to check accuracy of the MCS implementation.
- A MC estimator for the value of the ZCB at t is derived as follows.
- Consider a certain path i of the I simulated paths for the short rate process with time grid $t \in \{0, \Delta t, 2\Delta t, \ldots, T\}$.
- We discount the terminal value of the ZCB, i.e. 1, step-by-step backward. For $t < T$ and $s = t - \Delta t$ we have

$$B_{s,i} = B_{t,i} e^{-\frac{r_t + r_s}{2}\Delta t}$$

- The MC estimator of the ZCB value at t is

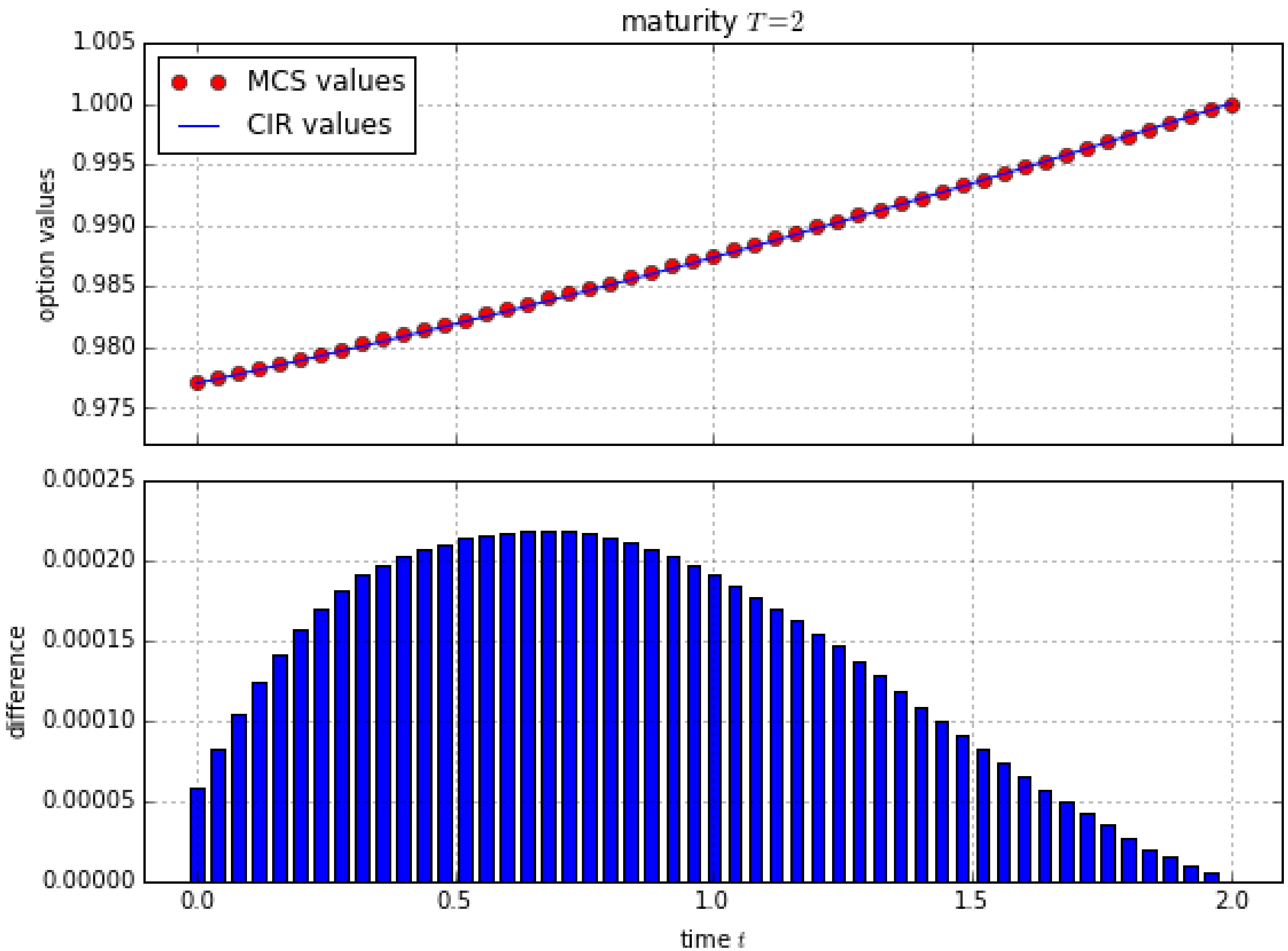
$$B_t^{MC} = \frac{1}{I}\sum_{i=1}^{I} B_{t,i}$$

CIR Model: Pricing ZCB

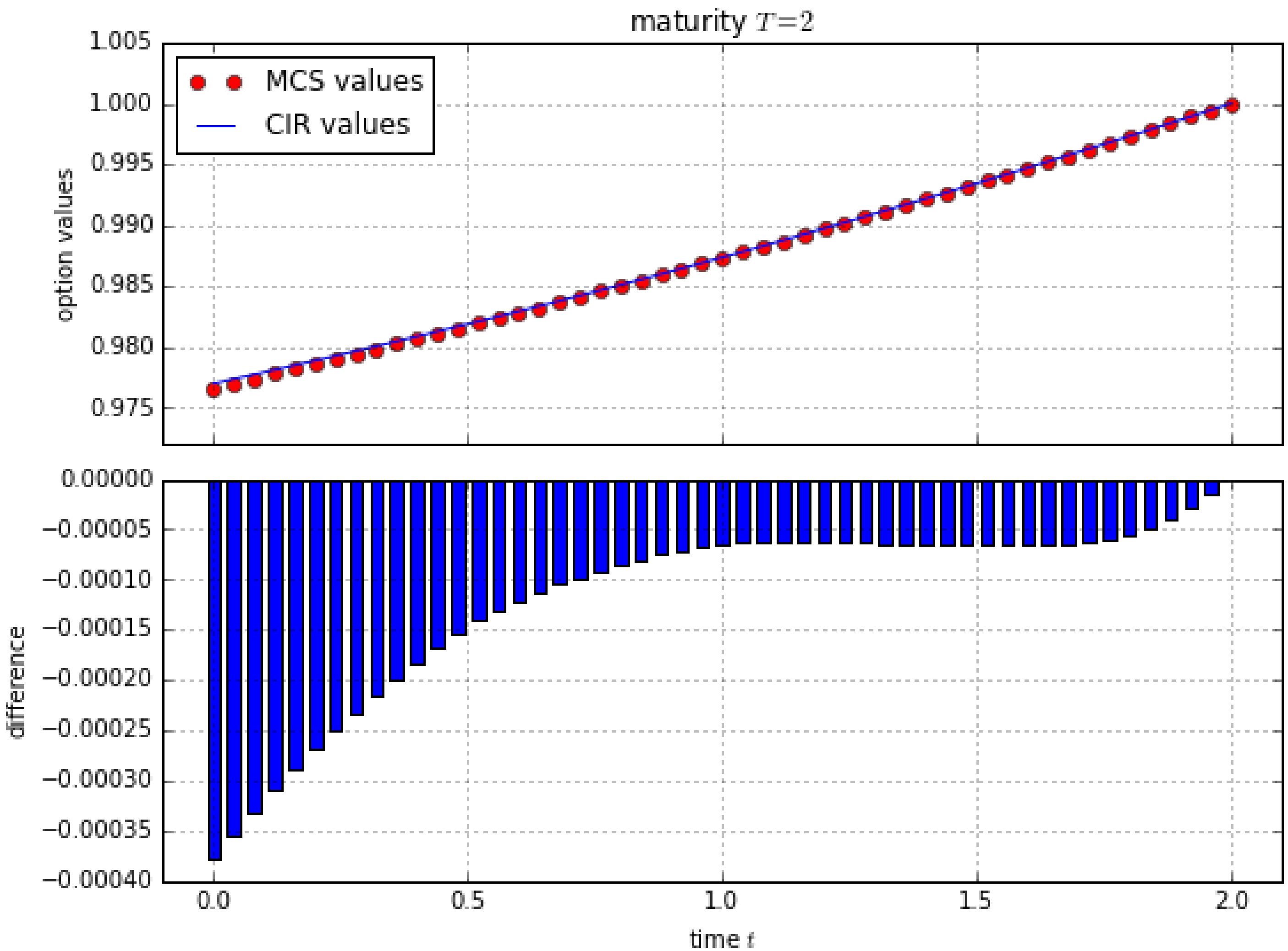
```
def CIR_generate_paths(r0, kappa_r, theta_r, sigma_r, T, M, I, x_disc):
    if x_disc is 'exact':
        return CIR_generate_paths_exact(r0, kappa_r, theta_r, sigma_r, T, M, I
    else:
        return CIR_generate_paths_approx(r0, kappa_r, theta_r, sigma_r, T, M, I

def zcb_estimator(M=50, x_disc='exact'):
    dt = T / M
    r = CIR_generate_paths(r0, kappa_r, theta_r, sigma_r, T, M, I, x_disc)
    zcb = np.zeros((M + 1, I), dtype=np.float)
    zcb[-1] = 1.0  # final value
    for t in range(M, 0, -1):
        zcb[t - 1] = zcb[t] * np.exp(-(r[t] + r[t - 1]) / 2 * dt)
    return np.sum(zcb, axis=1) / I
```

CIR Model: Pricing ZCB



CIR Model: Pricing ZCB



Notebook

- **GitHub :** polyhedron-gdl;
- **Notebook :** n06_mcs_cir;

Subsection 2

The Heston Model

The Heston Model

- Stochastic volatility models are those in which the variance of a stochastic process is itself randomly distributed.
- The models assumes that the underlying security's volatility is a random process, governed by state variables such as the price level of the underlying security, the tendency of volatility to revert to some long-run mean value, and the variance of the volatility process itself, among others.
- Stochastic volatility models are one approach to resolve a shortcoming of the Black–Scholes model.
- In particular this model cannot explain long-observed features of the implied volatility surface such as volatility smile and skew, which indicate that implied volatility does tend to vary with respect to strike price and expiry.

The Heston Model

By assuming that the volatility of the underlying price is a stochastic process rather than a constant, it becomes possible to model derivatives more accurately.

- Heston model
- CEV model
- SABR volatility model
- GARCH model

The Heston Model

- In this section we are going to consider the stochastic volatility model MH93 with constant short rate.
- This section values European call and put options in this model by MCS.
- As for the ZCB values, we also have available a semi-analytical pricing formula which generates natural benchmark values against which to compare the MCS estimates.

The Heston Model

- The basic Heston model assumes that S_t, the price of the asset, is determined by a stochastic process:

$$dS_t = \mu S_t\,dt + \sqrt{\nu_t}S_t\,dW_t^S$$

where ν_t, the instantaneous variance, is a CIR process:

$$d\nu_t = \kappa(\theta - \nu_t)\,dt + \xi\sqrt{\nu_t}\,dW_t^\nu$$

and dW_t^S, dW_t^ν are Wiener process with correlation ρ, or equivalently, with covariance ρdt.

The Heston Model

The parameters in the above equations represent the following:

- μ is the rate of return of the asset.
- θ is the *long variance*, or long run average price variance; as t tends to infinity, the expected value of ν_t tends to θ.
- κ is the rate at which ν_t reverts to θ.
- ξ is the volatility of the volatility, or *vol of vol*, and determines the variance of ν_t.

If the parameters obey the following condition (known as the Feller condition) then the process ν_t is strictly positive

$$2\kappa\theta > \xi^2$$

The Heston Model

- The correlation introduces a new problem dimension into the discretization for simulation purposes.
- To avoid problems arising from correlating normally distributed increments (of S) with chi-squared distributed increments (of v), we will in the following only consider Euler schemes for both the S and v process.
- This has the advantage that the increments of v become normally distributed as well and can therefore be easily correlated with the increments of S.

The Heston Model

- we consider two discretization schemes for S and seven discretization schemes for v.
- For S we have the simple Euler discretization scheme (with $s = t - \Delta t$)

$$S_t = S_s \left(e^{r\Delta t} + \sqrt{v_t}\sqrt{\Delta t} z_t^1 \right)$$

As an alternative we consider the exact log Euler scheme

$$S_t = S_s e^{(r - v_t/2)\Delta t + \sqrt{v_t}\sqrt{\Delta t} z_t^1}$$

This one is obtained by considering the dynamics of $\log S_t$ and applying Ito's lemma to it.

The Heston Model

- These schemes can be combined with any of the following Euler schemes for the square-root diffusion ($x^+ = \max[0, x]$):
- **Full Truncation**

$$\tilde{x}_t = \tilde{x}_s + \kappa(\theta - \tilde{x}_s^+)\Delta t + \sigma\sqrt{\tilde{x}_s^+}\sqrt{\Delta t}z_t, \quad x_t = \tilde{x}_t^+$$

- **Partial Truncation**

$$\tilde{x}_t = \tilde{x}_s + \kappa(\theta - \tilde{x}_s)\Delta t + \sigma\sqrt{\tilde{x}_s^+}\sqrt{\Delta t}z_t, \quad x_t = \tilde{x}_t^+$$

- **Truncation**

$$x_t = \max\left[0, \tilde{x}_s + \kappa(\theta - \tilde{x}_s)\Delta t + \sigma\sqrt{\tilde{x}_s}\sqrt{\Delta t}z_t\right]$$

- **Reflection**

$$\tilde{x}_t = |\tilde{x}_s| + \kappa(\theta - |\tilde{x}_s|)\Delta t + \sigma\sqrt{|\tilde{x}_s|}\sqrt{\Delta t}z_t, \quad x_t = |\tilde{x}_t|$$

The Heston Model

- **Hingham-Mao**

$$\tilde{x}_t = \tilde{x}_s + \kappa(\theta - \tilde{x}_s)\Delta t + \sigma\sqrt{|\tilde{x}_s|}\sqrt{\Delta t}z_t, \quad x_t = |\tilde{x}_t|$$

- **Simple Reflection**

$$\tilde{x}_t = \left| \tilde{x}_s + \kappa(\theta - \tilde{x}_s)\Delta t + \sigma\sqrt{\tilde{x}_s}\sqrt{\Delta t}z_t \right|$$

- **Absorption**

$$\tilde{x}_t = \tilde{x}_s^+ + \kappa(\theta - \tilde{x}_s^+)\Delta t + \sigma\sqrt{\tilde{x}_s^+}\sqrt{\Delta t}z_t, \quad x_t = \tilde{x}_t^+$$

- In the literature there are a lot of tests and numerical studies available that compare efficiency and precision of different discretization schemes.

Notebook

- **GitHub :** polyhedron-gdl;
- **Notebook :** n07_mcs_heston;

Outline

1. Beyond Black and Scholes
   - Square-Root Diffusion: the CIR Model
   - The Heston Model
2. The Hull and White Model

Hull-White Model

- Described by the SDE for the short rate

$$dr = (\theta(t) - ar)dt + \sigma dw \tag{3}$$

- See Brigo-Mercurio ...
- Our version simplified: a and σ constant;
- AKA Extended Vasicek (Note: $r(t)$ is Gaussian);
- θ determined uniquely by term structure;

Hull-White Model: Solving for r(t)

$$d(e^{at}r) = e^{at}dr + ae^{at}rdt = \theta(t)e^{at} + e^{at}\sigma dw$$

integrating both sides we obtain

$$e^{at}r(t) = r(0) + \int_0^t \theta(s)e^{as}ds + \sigma \int_0^t e^{as}dw(s)$$

simplify

$$r(t) = r(0)e^{-at} + \int_0^t \theta(s)e^{-a(t-s)}ds + \sigma \int_0^t e^{-a(t-s)}dw(s)$$

Hull-White Model: Solving for P(t,T)

- $P(t, T) = V(t, r(t))$ where V solves the PDE

$$V_t + (\theta(t) - ar)V_r + \frac{1}{2}\sigma^2 V_{rr} - rV = 0$$

- Final-time condition $V(T, r) = 1$ for all r at $t = T$;
- Ansatz:

$$V = A(t, T)e^{-B(t,T)r(t)}$$

- A and B must satisfy:

$$A_t - \theta(t)AB + \frac{1}{2}\sigma^2(AB)^2 = 0, \quad and \quad B_t - aB + 1 = 0$$

- Final-time conditions

$$A(T, T) = 1 \quad and \quad B(T, T) = 0$$

Hull-White Model: Solving for P(t,T)

- B independent of θ so

$$B(t,T) = \frac{1}{a}\left(1 - e^{-a(T-t)}\right) \tag{4}$$

- Solving for A requires integration of θ

$$A(t,T) = exp\left[-\int_t^T \theta(s)B(s,T)ds - \frac{\sigma^2}{2a^2}\left(B(t,T) - T + t\right) - \frac{\sigma^2}{4a}B(t,T)^2\right]$$

Hull-White Model: Determining θ

- Determining θ from the term structure at time 0;
- Goal: demonstrate the relation

$$\theta(t) = \frac{\partial f}{f \partial T}(0, t) + af(0, t) + \frac{\sigma^2}{2a}(1 - e^{-2at}) \tag{5}$$

- Recall

$$f(t, T) = -\partial \log P(t, T)/\partial T$$

Hull-White Model: Determining θ

- We have

$$-\log P(0,T) = \int_0^T \theta(s)B(s,T)ds + \frac{\sigma^2}{2a^2}[B(0,T) - T] + \frac{\sigma^2}{4a}B(0,T)^2 + B(0,T)r_0$$

- Differentiating and using that $B(T,T) = 1$ and $\partial_T B - 1 = -aB$ we get

$$f(0,T) = \int_0^T \theta(s)\partial_T B(s,T)ds - \frac{\sigma^2}{2a^2}B(0,T) + \frac{\sigma^2}{2a^2}B(0,T)\partial_T B(0,T) + \partial_T B(0,T)r_0$$

Hull-White Model: Determining θ

- Differentiating again, get:

$$
\begin{aligned}
\partial_T f(0,T) = \theta(T) + \int_0^T \theta(s)\partial_{TT}B(s,T)ds \\
- \frac{\sigma^2}{2a^2}\partial_T B(0,T) \\
+ \frac{\sigma^2}{2a^2}[(\partial_T B(0,T))^2 + B(0,T)\partial_{TT}B(0,T)] \\
+ \partial_{TT}B(0,T)r_0
\end{aligned}
\tag{6}
$$

Hull-White Model: Determining θ

- Combine these equations, and use $a\partial_T B + \partial_{TT} B = 0$;
- Get:

$$af(0,T) + \partial_T f(0,T) = \theta(T) - \frac{\sigma^2}{2a}(aB + \partial_T B) + \frac{\sigma^2}{2a}[aB\partial_T B + (\partial_T B)^2 + B\partial_{TT} B]$$

- Substitute formula for B and simplify to get

$$af(0,T) + \partial_T f(0,T) = \theta(T) - \frac{\sigma^2}{2a}(1 - e^{-2aT})$$

QED

Additive Factor Gaussian Model

- The model is given by dynamics (Brigo-Mercurio p. 143):

$$r(t) = x(t) + \phi(t)$$

where

$$dx(t) = -ax(t)dt + \sigma dW_t \quad x(0) = 0$$

and ϕ is a deterministic shift which is added in order to fit exactly the initial zero coupon curve

Additive Factor Gaussian Model

- So the short rate $r(t)$ is distributed normally with mean and variance given by (Brigo-Mercurio p.144 equations 4.6 with $\eta = 0$)

$$E(r_t|r_s) = x(s)e^{-a(t-s)} + \phi(t)$$

$$Var(r_t|r_s) = \frac{\sigma^2}{2a}\left(1 - e^{-2a(t-s)}\right)$$

where $\phi(T) = f^M(0, T) + \frac{\sigma^2}{2a}\left(1 - e^{-aT}\right)^2$ and $f^M(0, T)$ is the market instantaneous forward rate at time t as seen at time 0.

Additive Factor Gaussian Model

- Model discount factors are calculated as in Brigo-Mercurio (section 4.2):

$$P(t,T) = \frac{P^M(0,T)}{P^M(0,t)} \exp\left(\mathcal{A}(t,T)\right)$$

$$\mathcal{A}(t,T) = \frac{1}{2}\left[V(t,T) - V(0,T) + V(0,t)\right] - \frac{1-e^{-a(T-t)}}{a} x(t)$$

where

$$V(t,T) = \frac{\sigma^2}{a^2}\left[T - t + \frac{2}{a}e^{-a(T-t)} - \frac{1}{2a}e^{-2a(T-t)} - \frac{3}{2a}\right]$$