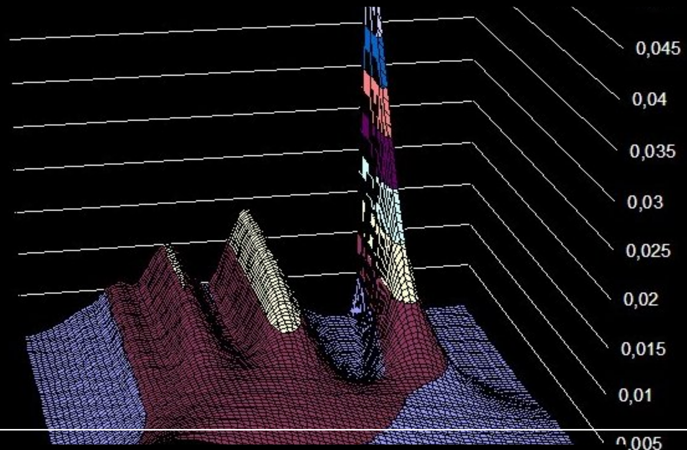
Masterclass: Implementing affine jump diffusion models at the trading desk



Marcello Minenna - Paolo Verzella
Montecarlo – Derivatives and Risk Management Europe 2006

Syllabus of the presentation

- Review of Fourier Methods in Option Pricing
- Calibration and Performance
- Greek derivation
- Greek Behavior of New FT-Q

Syllabus of the presentation

- Review of Fourier Methods in Option Pricing
- Calibration and Performance
- Greek derivation
- Greek Behavior of New FT-Q

Review of Fourier Methods in Option Pricing – theory

European Call Maturity T Terminal Spot Price S_T

In AJD models Call Price can be expressed in a form close to the canonical Black – Scholes - Merton style

$$C_t = S_t P_1(\Theta) - Ke^{-r\tau} P_2(\Theta)$$

where

$$P_1(\Theta), P_2(\Theta) = \Pr(\ln S_T \geq \ln[K])$$

under different martingale measures

$$P_1(\Theta), P_2(\Theta) = \Pr(\ln S_T \geq \ln[K])$$

under different martingale measures

can be

determined by using the Levy's inversion formula, i.e.:

$$\Pr(\ln S_T \geq \ln[K]) = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-i\phi\ln[K]}\tilde{f}_j(\phi)}{i\phi}\right]d\phi$$

$$\Pr(\ln S_T \geq \ln[K]) = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left[\frac{e^{-i\phi\ln[K]}\tilde{f}_j(\phi)}{i\phi}\right]d\phi$$

requires

a close formula for the Characteristic Function of the log – terminal price, i.e.:

$$\tilde{f}_T(\phi) = E\left[e^{i\phi\ln S_T}\right]$$

$$\tilde{f}_T(\phi) = E\left[e^{i\phi\ln S_T}\right]$$

has

a closed formula for AJD models

Example of derivation for Heston Model

$$dS_t = \mu S_t dt + \sqrt{v_t} S_t dz_t^{(1)}$$

$$dv_t = \kappa[\theta - v_t]dt + \sigma\sqrt{v_t}dz_t^{(2)}$$

Example of derivation for Heston Model

PDE Derivation for portfolio replication

$$f = f(S, v, t)$$

$$\begin{aligned} df = {} & \frac{\partial f}{\partial S}\left(\mu S dt + \sqrt{v} S dz_1\right) + \frac{\partial f}{\partial t} dt + \frac{\partial f}{\partial v}\left[\kappa(\theta - v)\, dt + \sigma\sqrt{v} dz_2\right] + \\ & + \frac{1}{2}\frac{\partial^2 f}{\partial S^2}\left(v S^2 dt\right) + \frac{1}{2}\frac{\partial^2 f}{\partial v^2}\left(\sigma^2 v dt\right) + \frac{\partial^2 f}{\partial S \partial v}\left(S \sigma \rho_{1,2} v\right) dt \end{aligned}$$

Example of derivation for Heston Model

PDE Derivation for portfolio replication

$$\pi = f_1 - \Delta_1 f_0 - \Delta_0 S$$

the coefficients Δ_1, Δ_0 are chosen in order to vanish any randomness of the portfolio

$$\begin{aligned} d\pi = {} & \frac{\partial f_1}{\partial t} dt + \frac{1}{2}\frac{\partial^2 f_1}{\partial S^2}\left(v S^2 dt\right) + \frac{\partial f_1}{\partial v}\left[\kappa(\theta - v)\, dt\right] + \frac{1}{2}\frac{\partial^2 f_1}{\partial v^2}\left(\sigma^2 v dt\right) + \\ & + \frac{\partial^2 f_1}{\partial S \partial v}\left(S \sigma \rho_{1,2} v\right) dt - \frac{\partial f_1/\partial v}{\partial f_0/\partial v}\frac{\partial f_0}{\partial t} dt - \\ & - \frac{\partial f_1/\partial v}{\partial f_0/\partial v}\frac{\partial f_0}{\partial v}\left[\kappa(\theta - v)\, dt\right] - \frac{\partial f_1/\partial v}{\partial f_0/\partial v}\frac{1}{2}\frac{\partial^2 f_0}{\partial S^2}\left(v S^2 dt\right) - \frac{\partial f_1/\partial v}{\partial f_0/\partial v}\frac{1}{2}\frac{\partial^2 f_0}{\partial v^2}\left(\sigma^2 v dt\right) - \\ & - \frac{\partial f_1/\partial v}{\partial f_0/\partial v}\frac{\partial^2 f_0}{\partial S \partial v}\left(S \sigma \rho_{1,2} v\right) dt \end{aligned}$$

Example of derivation for Heston Model

PDE Derivation for portfolio replication

no arbitrage hypothesis $d\pi = r\pi dt$

$$\frac{-r f_0 + \frac{\partial f_0}{\partial t} + \frac{\partial f_0}{\partial S} r S + \frac{1}{2}\frac{\partial^2 f_0}{\partial S^2} v S^2 + \frac{1}{2}\frac{\partial^2 f_0}{\partial v^2}\sigma^2 v + \frac{\partial^2 f_0}{\partial S \partial v} S \sigma \rho_{1,2} v - \left[\kappa(\theta - v)\right]\partial f_0/\partial v}{\partial f_0/\partial v} =$$

$$= \frac{-r f_1 + \frac{\partial f_1}{\partial t} + \frac{\partial f_1}{\partial S} r S + \frac{1}{2}\frac{\partial^2 f_1}{\partial S^2} v S^2 + \frac{1}{2}\frac{\partial^2 f_1}{\partial v^2}\sigma^2 v + \frac{\partial^2 f_1}{\partial S \partial v} S \sigma \rho_{1,2} v - \left[\kappa(\theta - v)\right]\partial f_1/\partial v}{\partial f_1/\partial v}$$

$$\frac{-r f + \frac{\partial f}{\partial t} + \frac{\partial f}{\partial S} r S + \frac{1}{2}\frac{\partial^2 f}{\partial S^2} v S^2 + \frac{1}{2}\frac{\partial^2 f}{\partial v^2}\sigma^2 v + \frac{\partial^2 f}{\partial S \partial v} S \sigma \rho_{1,2} v - \left[\kappa(\theta - v)\right]\partial f/\partial v}{\partial f/\partial v} = \lambda^*(S, v, t)$$

Example of derivation for Heston Model

PDE specification for the pricing of a Call option:

$$-rC + \frac{\partial C}{\partial t} + \frac{\partial C}{\partial S} r S + \frac{1}{2}\frac{\partial^2 C}{\partial S^2} v S^2 + \frac{1}{2}\frac{\partial^2 C}{\partial v^2}\sigma^2 v + \frac{\partial C}{\partial S \partial v} S \sigma \rho_{1,2} v + \frac{\partial C}{\partial v}\left[\kappa(\theta - v) - \lambda^*(S, v, t)\right] = 0$$

$$C(S, v, t = T) = \max(0, S_T - K)$$

Example of derivation for Heston Model

PDE Shift into the forward space

$$\tilde{C}(x,v,\tau) = e^{r\tau} C(x,v,\tau) = e^{r(t-t)} C(S,v,t,T)$$

$$-\frac{\partial \tilde{C}}{\partial \tau} + r\frac{\partial \tilde{C}}{\partial x} + \frac{1}{2}\frac{\partial^2 \tilde{C}}{\partial v^2}(\sigma^2 v) + \frac{\partial^2 \tilde{C}}{\partial x \partial v}(v\sigma\rho_{1,2}) + \frac{1}{2}\left(\frac{\partial^2 \tilde{C}}{\partial x^2} - \frac{\partial \tilde{C}}{\partial x}\right)v + \frac{\partial \tilde{C}}{\partial v}\left[\kappa(\theta - v) - \tilde{\lambda} v\right] = 0$$

$$\tilde{C}(x_\tau, v_\tau, \tau = 0) = \max(0, e^{x_{\tau=0}} - K)$$

Example of derivation for Heston Model

PDE Shift into Black-Scholes-Merton space

$$C_t(S,v,t,T) = S_t P_1(S,v,t,T) - K e^{-r(T-t)} P_2(S,v,t,T)$$

$$\tilde{C}_t(x,v,\tau) = e^{x_\tau} P_1(x,v,\tau) - K P_2(x,v,\tau)$$

Example of derivation for Heston Model

PDE Shift into Black-Scholes-Merton space

$$-\frac{\partial P_j}{\partial \tau} + \frac{\partial P_j}{\partial x}(r + c_j v) + \frac{1}{2}\frac{\partial^2 P_j}{\partial v^2}(\sigma^2 v) + \frac{\partial^2 P_j}{\partial x \partial v}(v\sigma\rho_{1,2}) + \frac{1}{2}\frac{\partial^2 P_j}{\partial x^2}v + \frac{\partial P_j}{\partial v}(a - b_j v) = 0$$

$$P_j(x_\tau, v_\tau, \tau = 0) = 1_{(x_\tau \geq \ln K)}$$

where $c_1 = \frac{1}{2}, \quad c_2 = -\frac{1}{2}, \quad a = \kappa\theta, \quad b_1 = \kappa + \tilde{\lambda} - \rho_{1,2}\sigma, \quad b_2 = \kappa + \tilde{\lambda}$

by using Feynman Cac formula….

characteristics of the probability measure P_j at a generic time τ:

$$P_j(x_\tau, v_\tau, \tau) = P_j(x_{\tau=0} \geq \ln K \mid x_\tau, v_\tau)$$

Example of derivation for Heston Model

PDE Shift into Fourier space

by using the Levy's inversion formula…

$$P_j(x_{\tau=0} \geq \ln K \mid x_\tau, v_\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \frac{e^{-i\xi \ln K}}{i\xi} \tilde{f}_j(x_\tau, v_\tau, \tau = 0, \xi \mid x_\tau, v_\tau)\, d\xi$$

$$-\frac{\partial \tilde{f}_j}{\partial \tau} + \frac{\partial \tilde{f}_j}{\partial x}(r + c_j v) + \frac{1}{2}\frac{\partial^2 \tilde{f}_j}{\partial v^2}(\sigma^2 v) + \frac{\partial^2 \tilde{f}_j}{\partial v \partial x}(v\sigma\rho_{1,2}) + \frac{\partial^2 \tilde{f}_j}{\partial x^2}v + \frac{\partial \tilde{f}_j}{\partial v}[a - b_j v]$$

$$\tilde{f}_j(x_\tau, v_\tau, \tau = 0, \xi) = e^{i\xi x_{\tau=0}}$$

Example of derivation for Heston Model

PDE Shift into ODE space

by using the solution: $\tilde{f}_j(x_\tau, v_\tau, \tau = 0, \xi | x_\tau, v_\tau) = e^{\left(C_\tau^{(j)} + D_\tau^{(j)} v_t + i\xi x_\tau\right)}$

$$
\begin{aligned}
\frac{\partial C_j}{\partial \tau} &= ri\xi + aD_j \\
\frac{\partial D_j}{\partial \tau} &= c_j i\xi + \frac{1}{2}D_j^2\sigma^2 + i\xi D_j \sigma \rho_{1,2} - \frac{1}{2}\xi^2 - b_j D_j \\
C_0^{(j)} &= 0 \\
D_0^{(j)} &= 0
\end{aligned}
$$

Example of derivation for Heston Model

ODE Solutions

$$C_j = ri\xi(T-t) - \frac{2a}{\sigma^2}\left(\alpha_2(T-t) + \ln\frac{\frac{\alpha_2}{\alpha_1}e^{d(T-t)} - 1}{\frac{\alpha_2}{\alpha_1} - 1}\right)$$

$$D_j = -\frac{2\alpha_2}{\sigma^2}\frac{1 - e^{d(T-t)}}{1 - \frac{\alpha_2}{\alpha_1}e^{d(T-t)}}$$

$$
\begin{aligned}
d &= \sqrt{(\rho_{1,2}\sigma\xi i - b_j)^2 - \sigma^2(2c_j\xi i - \xi^2)} \\
\alpha_1 &= \frac{\rho_{1,2}\sigma\xi i - b_j + d}{2}, \\
\alpha_2 &= \frac{\rho_{1,2}\sigma\xi i - b_j - d}{2}
\end{aligned}
$$

Example of derivation for Heston Model

PRICING

$$C_t = S_t P_1 - Ke^{-r(T-t)}P_2$$

$$P_j = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \Re\left\{\frac{e^{-i\xi\ln K}}{i\xi} e^{\left[C_\tau^{(j)} + D_\tau^{(j)} v_t + i\xi[\ln S_t + r(T-t)]\right]}\right\} d\xi$$

with:

$$C_j = ri\xi(T-t) - \frac{2a}{\sigma^2}\left(\alpha_2(T-t) + \ln\frac{\frac{\alpha_2}{\alpha_1}e^{d(T-t)} - 1}{\frac{\alpha_2}{\alpha_1} - 1}\right)$$

$$D_j = -\frac{2\alpha_2}{\sigma^2}\frac{1 - e^{d(T-t)}}{1 - \frac{\alpha_2}{\alpha_1}e^{d(T-t)}}$$

$$d = \sqrt{(\rho_{1,2}\sigma\xi i - b_j)^2 - \sigma^2(2c_j\xi i - \xi^2)}$$

$$\alpha_1 = \frac{\rho_{1,2}\sigma\xi i - b_j + d}{2}, \quad \alpha_2 = \frac{\rho_{1,2}\sigma\xi i - b_j - d}{2}$$

$$
\begin{aligned}
c_{1/2} &= \pm\tfrac{1}{2} \\
a &= \kappa\theta \\
b_1 &= \kappa + \tilde{\lambda} - \rho_{1,2}\sigma \\
b_2 &= \kappa + \tilde{\lambda}
\end{aligned}
$$

How to compute: C_t

I

Quadrature Algorithm FT – Q for $P_1(\Theta), P_2(\Theta)$

a — **Old FT - Q**

b — **New FT - Q**

II

Fast Fourier Transform FFT

Algorithms Valuation Criteria

STABILITY

The algorithm is defined stable if and only if it "closes" the quadrature scheme

spanning the pricing formula on a vast area of the parameters set. → gives → a "reasonable" result different from a NaN value

Algorithms Valuation Criteria

ACCURACY

The algorithm is defined accurate if and only if

pricing the Call under the Black - Scholes - Merton model via:

- Old FT - Q
- New FT - Q
- FFT

⟷ pricing Call via standard Black – Scholes – Merton

up to 10^{-3} precision

Algorithms Valuation Criteria

SPEED

The algorithm is defined fast with respect to the results of the FFT algorithm

⟷

a set of 4100 prices along the strike

$P_1(\Theta), P_2(\Theta)$ ⟷ through ⟷ Quadrature Algorithm Old FT - Q

High Order Newton Cotes Algorithm

Up to 8th

⟷

$$C_t = S_t P_1(\Theta) - Ke^{-r\tau} P_2(\Theta)$$

$P_1(\Theta), P_2(\Theta)$ ⟺ through ⟺ Quadrature Algorithm
Old FT - Q

Pros (+)	Cons (-)
ACCURACY	STABILITY SPEED

$P_1(\Theta), P_2(\Theta)$ ⟺ through ⟺ Quadrature Algorithm
New FT - Q

In order to overcome the cited problems of Old FT – Q:

- Gauss - Lobatto Quadrature Algorithm
- Re-adjustment of $\widetilde{f}_T(\phi) = E\left[e^{i\phi \ln S_T}\right]$

⇕

$$C_t = S_t P_1(\Theta) - Ke^{-r\tau} P_2(\Theta)$$

$P_1(\Theta), P_2(\Theta)$ ⟺ through ⟺ Quadrature Algorithm
New FT - Q

In order to overcome the cited problems of Old FT – Q:

- Gauss - Lobatto Quadrature Algorithm
- Re-adjustment of $\widetilde{f}_T(\phi) = E\left[e^{i\phi \ln S_T}\right]$

⇕

$$C_t = S_t P_1(\Theta) - Ke^{-r\tau} P_2(\Theta)$$

- Basic Gauss - Lobatto Quadrature Formula

⇕

$$\int_{-1}^{1} f(x)\,dx \approx w_1 f(-1) + w_N f(1) + \sum_{i=2}^{N-1} w_i f(x_i)$$

$$w_i = \frac{2}{N(N-1)\left[P_{N-1}(x_i)\right]^2}$$

LIMITED
to the interval (-1,1)

$$w_1 = w_N = \frac{2}{N(N-1)}$$

The Gautschi - Gander extension (2000)

ENHANCE

The Gauss Lobatto formula

They develop a GL recursive adaptive algorithm for a generic interval

The Gautschi - Gander extension (2000)

$$\int_\alpha^\beta f(x)\,dx \approx h\left\{w_1 f(\alpha) + w_N f(\beta) + \sum_{i=2}^{N-1} w_i\left[f(m + x_i h)\right]\right\}$$

$$w_i = \frac{2}{N(N-1)\left[P_{N-1}(x_i)\right]^2}$$

$$w_1 = w_N = \frac{2}{N(N-1)}$$

$$h = \frac{1}{2}(\beta - \alpha)$$

$$m = \frac{1}{2}(\alpha + \beta)$$

$P_1(\Theta), P_2(\Theta)$ through Quadrature Algorithm New FT - Q

In order to overcome the cited problems of Old FT – Q:

- Gauss - Lobatto Quadrature Algorithm
- Re-adjustment of $\tilde{f}_T(\phi) = E\left[e^{i\phi \ln S_T}\right]$

$$C_t = S_t P_1(\Theta) - K e^{-r\tau} P_2(\Theta)$$

Example of re-adjstment for Heston Model

$$C_t = S_t P_1 - K e^{-r(T-t)} P_2$$

$$P_j = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \Re\left\{\frac{e^{-i\xi \ln K}}{i\xi}\, e^{\left[C_\tau^{(j)} + D_\tau^{(j)} v_t + i\xi[\ln S_t + r(T-t)]\right]}\right\} d\xi$$

with:

$$C_j = ri\xi(T-t) - \frac{2a}{\sigma^2}\left(\alpha_2(T-t) + \ln\frac{\frac{\alpha_2}{\alpha_1}e^{d(T-t)} - 1}{\frac{\alpha_2}{\alpha_1} - 1}\right)$$

$$D_j = -\frac{2\alpha_2}{\sigma^2}\frac{1 - e^{d(T-t)}}{1 - \frac{\alpha_2}{\alpha_1}e^{d(T-t)}}$$

$$d = \sqrt{(\rho_{1,2}\sigma\xi i - b_j)^2 - \sigma^2(2c_j\xi i - \xi^2)}$$

$$\alpha_1 = \frac{\rho_{1,2}\sigma\xi i - b_j + d}{2}, \quad \alpha_2 = \frac{\rho_{1,2}\sigma\xi i - b_j - d}{2}$$

$$c_{1/2} = \pm\tfrac{1}{2}$$
$$a = \kappa\theta$$
$$b_1 = \kappa + \tilde{\lambda} - \rho_{1,2}\sigma$$
$$b_2 = \kappa + \tilde{\lambda}$$

Example of re-adjstment for Heston Model

$$C_j = ri\xi(T-t) - \frac{2a}{\sigma^2}\left(\alpha_2(T-t) + \ln\frac{\frac{\alpha_2}{\alpha_1}e^{d(T-t)} - 1}{\frac{\alpha_2}{\alpha_1} - 1}\right)$$

$$D_j = -\frac{2\alpha_2}{\sigma^2}\frac{1 - e^{d(T-t)}}{1 - \frac{\alpha_2}{\alpha_1}e^{d(T-t)}}$$

$$C_j = ri\xi\tau - \frac{a}{\sigma^2}\left(\rho_{1,2}\sigma\xi i - b_j + d\right)\tau - \frac{a}{\sigma^2}2\ln\left(1 - \frac{\left(1-e^{-d\tau}\right)\left(\rho_{1,2}\sigma\xi i - b_j + d\right)}{2d}\right)$$

$$D_j = \frac{\left(2c_j\xi i - \xi^2\right)\left(1 - e^{-d\tau}\right)}{2d - \left(\rho_{1,2}\sigma\xi i - b_j + d\right)\left(1 - e^{-d\tau}\right)}$$

Review of Fourier Methods in Option Pricing – practice

$P_1(\Theta), P_2(\Theta)$ ⇔ through ⇔ **Quadrature Algorithm**
New FT - Q

Pros (+)	**Cons (-)**
STABILITY	SPEED
ACCURACY	

Review of Fourier Methods in Option Pricing – practice

C_t ⇔ through ⇔ **Fast Fourier Trasform**
FFT

Cooley - Tukey algorithm

$$\omega(n) = \sum_{j=1}^{N} e^{-i\frac{2\pi}{N}(j-1)(n-1)} f_j = \sum_{j=1}^{\frac{N}{2}} e^{-i\frac{2\pi}{N}(2j-1)(n-1)} f_{2j} + \sum_{j=1}^{\frac{N}{2}} e^{-i\frac{2\pi}{N}2j(n-1)} f_{2j+1}$$

Review of Fourier Methods in Option Pricing – practice

C_t ⇔ through ⇔ **Fast Fourier Trasform**
FFT

Cooley - Tukey algorithm

Applied to the equivalent formula via a recombinant FFT parameters

$$C_t = \frac{e^{-\alpha \ln K}}{\pi}\int_0^{\infty} e^{-i\phi\ln K}\,\widetilde{f}_j(\phi)\,d\phi$$

for ATM

Review of Fourier Methods in Option Pricing – practice

C_t ⇔ through ⇔ Fast Fourier Trasform FFT

Pros (+)

SPEED
FASTER
(up to 40 times the quadrature algorithms)

Cons (-)

STABILITY
* The formula must be changed arbitrarily according to Option moneyness

ACCURACY
** the recombinant FFT parameters must be changed according to the choice of the pricing models

Syllabus of the presentation

- Review of Fourier Methods in Option Pricing
- Calibration and Performance
- Greek derivation
- Greek Behaviour of New FT-Q

The Calibration Procedure and Performance

$$SSE_t = \min_{v(t),\Phi} \sum_{n=1}^{N} \left[C_{Market}(S_t) - C_{AJD}(S_t) \right]^2$$

I → Quadrature Algorithm FT - Q

a → Old FT - Q

b → New FT - Q

II → Fast Fourier Trasform FFT

The Calibration Procedure and Performance

$SSE_t = \min_{v(t),\Phi} \sum_{n=1}^{N} \left[C_{Market}(S_t) - C_{AJD}(S_t) \right]^2$ ⇔ through ⇔ Quadrature Algorithm Old FT - Q

Pros (+)

Cons (-)

STABILITY
ACCURACY
SPEED

The Calibration Procedure and Performance

$$SSE_t = \min_{v(t),\Phi} \sum_{n=1}^{N} \left[C_{Market}(S_t) - C_{AJD}(S_t) \right]^2$$ ⟺ through ⟺ Fast Fourier Trasform FFT

Pros (+)	Cons (-)
SPEED	STABILITY * ACCURACY **

CONSOB

The Calibration Procedure and Performance

$$SSE_t = \min_{v(t),\Phi} \sum_{n=1}^{N} \left[C_{Market}(S_t) - C_{AJD}(S_t) \right]^2$$ ⟺ through ⟺ Quadrature Algorithm New FT - Q

Pros (+)	Cons (-)
STABILITY ACCURACY SPEED	

CONSOB

The Calibration Procedure and Performance

By keeping in mind that only New FT-Q is stable and accurate, some figures on speed

Original Option Pricing Formulas are used

	Heston Model	Merton Model	BCC Model
FFT	7.26 sec.	10.54 sec.	18.33 sec.
NEW FT - Q	55.12 sec.	66.48 sec.	110.39 sec.
OLD FT – Q	390.41 sec.	454.76 sec.	722.1 sec.

By now, the speed of Fourier Trasform method is closer than ever to the FFT calibration time

CONSOB

The Calibration Procedure and Performance

Calibration Performances using
Option Readjusted Pricing Formulas
where available

	Heston Model	Merton Model	BCC Model
FFT	7.24 sec.	10.54 sec.	18.32 sec.
NEW FT - Q	23.13 sec.	66.48 sec.	48.7 sec.
OLD FT – Q	331.6 sec.	454.76 sec.	688.5 sec.

CONSOB

Syllabus of the presentation

- Review of Fourier Methods in Option Pricing
- Calibration Procedure and Performance
- Greek derivation
- Greek Behaviour of New FT-Q

Greek derivation

European Call   Maturity T Terminal Spot Price S_T

In AJD models Greeks can be derived by using the following equivalences

$$S_t\frac{\partial P_1}{\partial S_t}+K\frac{\partial P_1}{\partial K}=0 \qquad S_t\frac{\partial P_2}{\partial S_t}+K\frac{\partial P_2}{\partial K}=0$$

$$\frac{\partial^2 P_1}{\partial S_t\partial K}=\frac{\partial^2 P_1}{\partial K\partial S_t} \qquad \frac{\partial^2 P_2}{\partial S_t\partial K}=\frac{\partial^2 P_2}{\partial K\partial S_t}$$

$$S_t\frac{\partial P_1}{\partial S_t}-e^{-r(T-t)}K\frac{\partial P_2}{\partial S_t}=0$$

$$P_1=\frac{\partial C_t}{\partial S_t}$$

$$\frac{\partial C_t}{\partial K}=-e^{-r(T-t)}P_2$$

Greek derivation

Example of derivation for Heston Model

$$\Delta_C=P_1$$

$$\Gamma_C=\frac{\partial P_1}{\partial S_t}$$

$$\mathcal{V}_C=S_t\frac{\partial P_1}{\partial v_t}-Ke^{-r\tau}\frac{\partial P_2}{\partial v_t}$$

$$\rho_C=K\tau e^{-r\tau}P_2$$

$$\begin{aligned}\Theta_C = & -\tfrac{\partial P_1}{\partial S}\left(\tfrac{1}{2}vS^2\right)-\tfrac{\partial P_1}{\partial v}S\left[\sigma\rho_{1,2}v+\left[\kappa\left(\theta-v\right)-\lambda v\right]\right]-\tfrac{\partial^2 P_1}{\partial v^2}\left(\tfrac{1}{2}S\sigma^2 v\right)- \\ & -Ke^{-r\tau}\left[rP_2-\tfrac{1}{2}\sigma^2 v\tfrac{\partial^2 P_2}{\partial v^2}-\tfrac{\partial P_2}{\partial v}\left[\kappa\left(\theta-v\right)-\lambda v\right]\right]\end{aligned}$$

$$\mathfrak{V}_C=S_t\frac{\partial^2 P_1}{\partial v_t^2}-Ke^{-r\tau}\frac{\partial^2 P_2}{\partial v_t^2}$$

Syllabus of the presentation

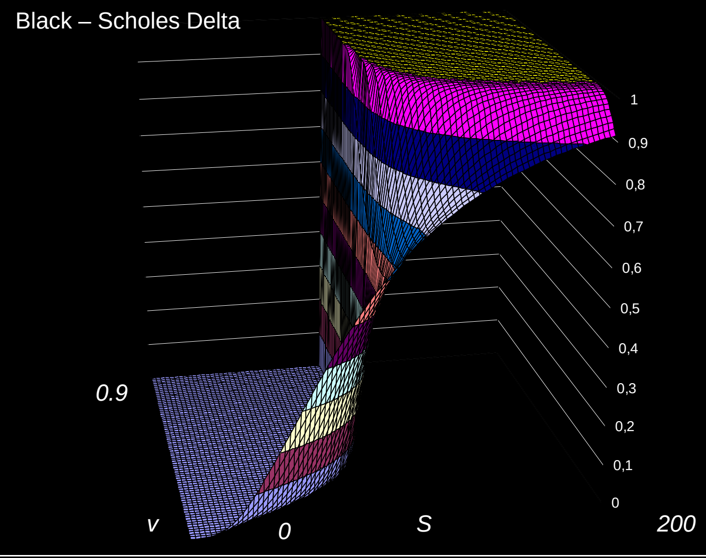
- Review of Fourier Methods in Option Pricing
- Calibration Procedure and Performance
- Greek derivation
- Greek Behaviour of New FT-Q

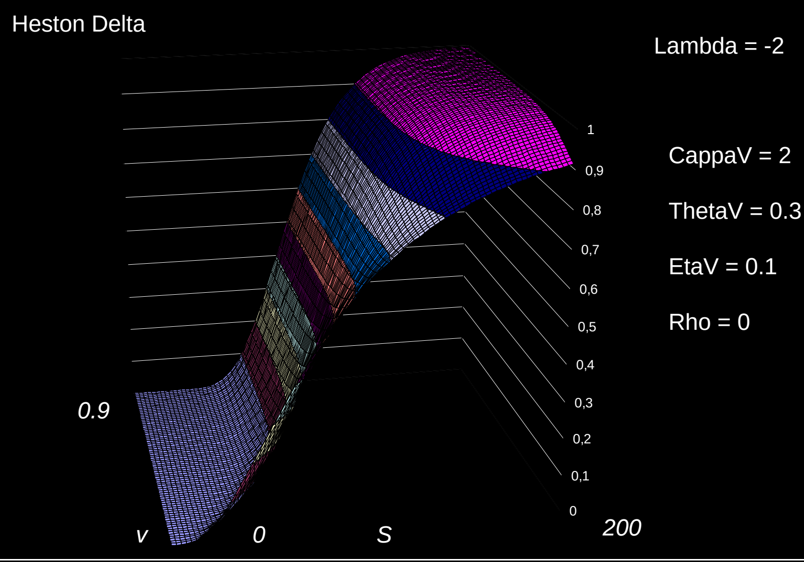
Greek behaviour of new FT-Q

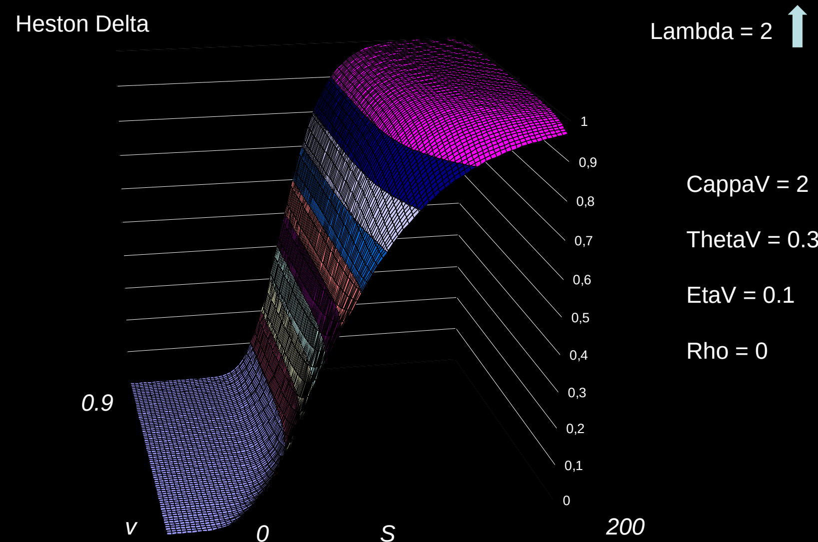
An impressive methodology to test Stability of the New FT – Quadrature algorithm is to compute Greeks

Infact, in an AJD setting the Greeks are available in closed form

So, an extended spanning of the AJD Greeks on the parameters set is useful to assess models and test Stability

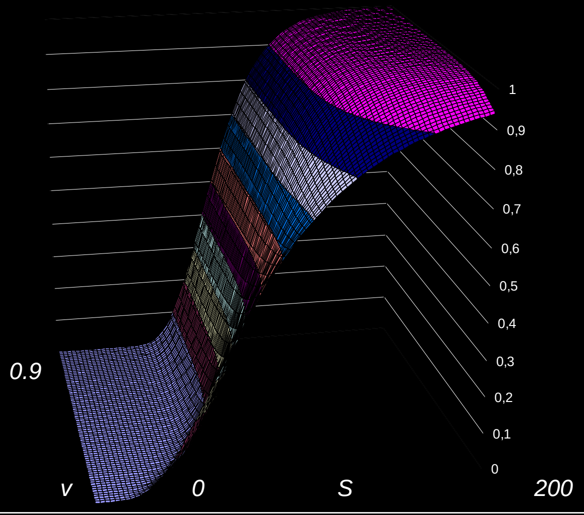






Heston Delta

Rho = -1



CappaV = 2

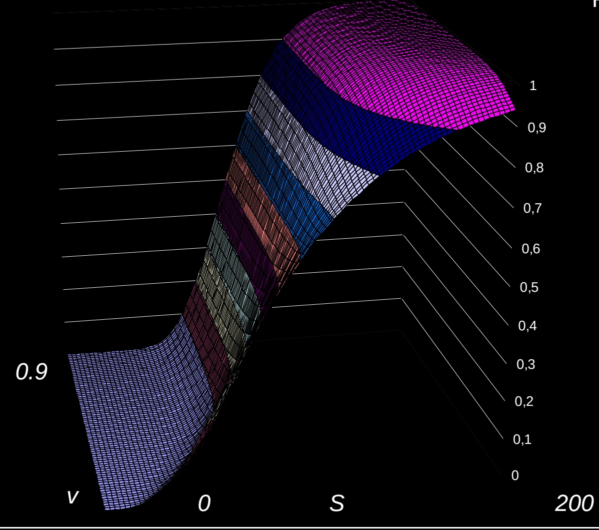
ThetaV = 0.3

EtaV = 0.1

Lambda = 0

Heston Delta

Rho = 1



CappaV = 2

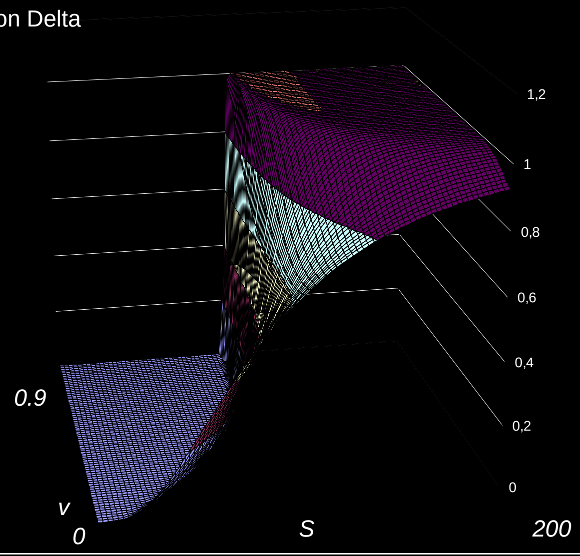
ThetaV = 0.3

EtaV = 0.1

Lambda = 0

Merton Delta

LambdaJ = 0

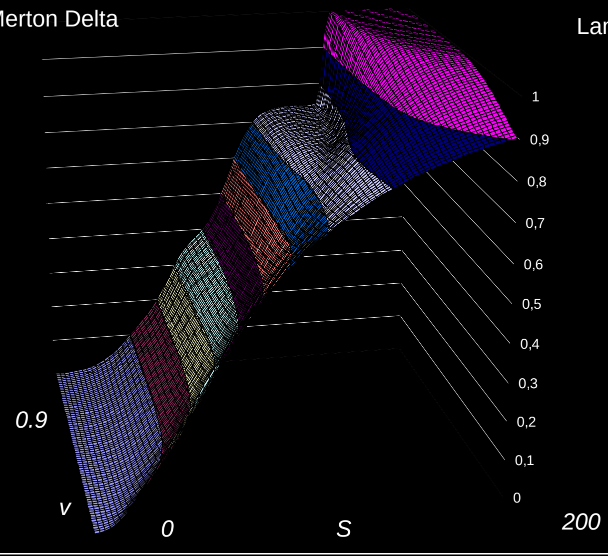


EtaJ = 0

MuJ= 0

Merton Delta

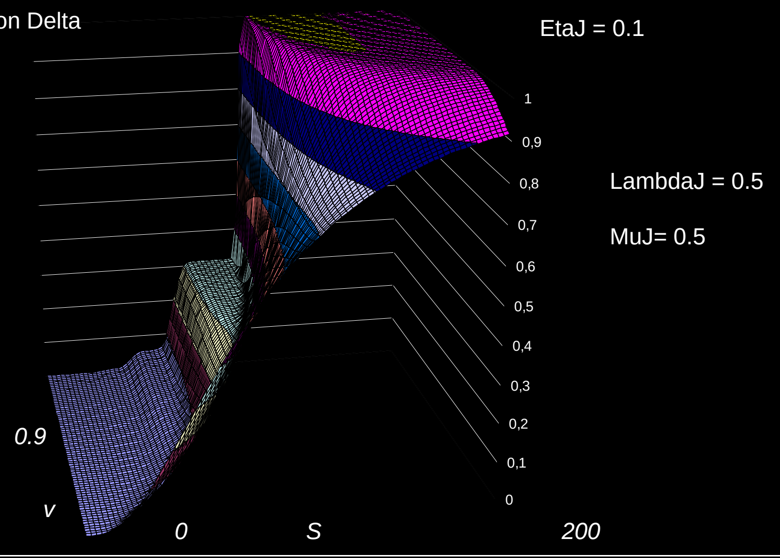
LambdaJ = 1.8



EtaJ = 0.1

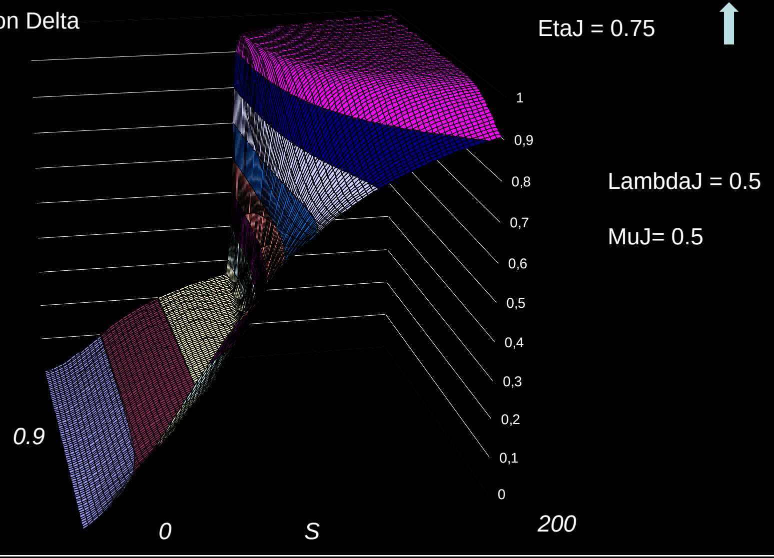
MuJ= 0.5

Merton Delta



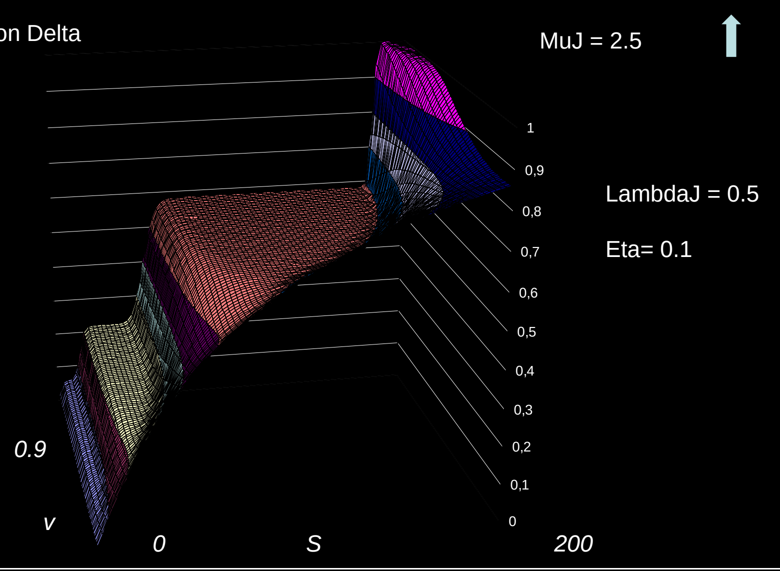
CONSOB

Merton Delta



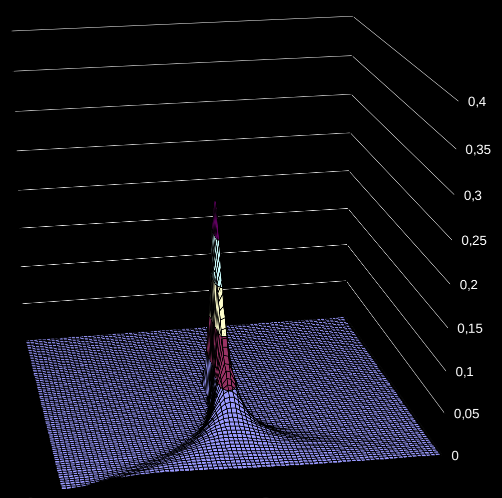
CONSOB

Merton Delta



CONSOB

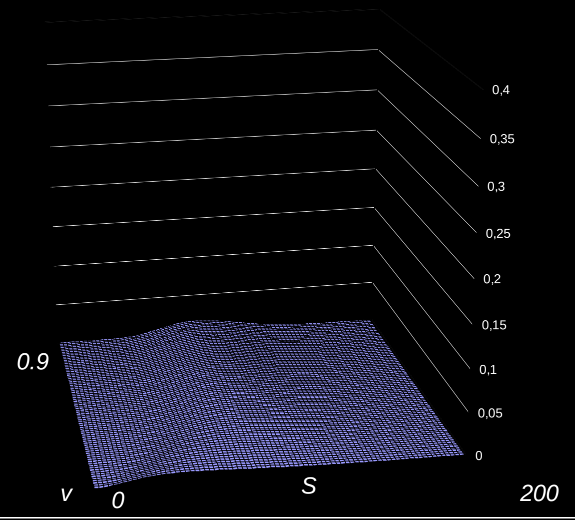
Black – Scholes Gamma



CONSOB

Heston Gamma

Lambda = -2



CappaV = 2

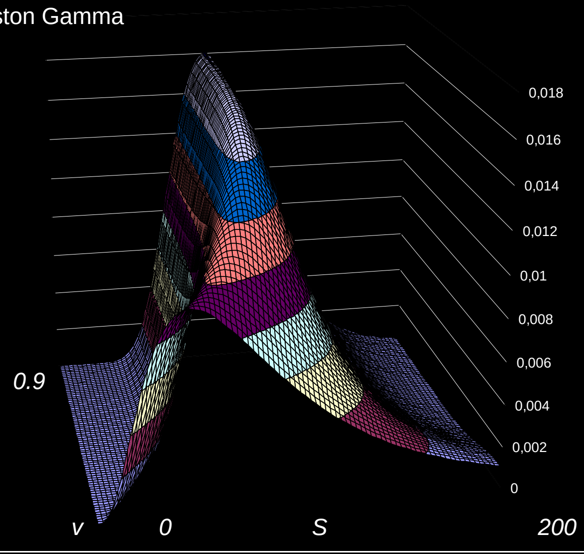
ThetaV = 0.3

EtaV = 0.1

Rho = 0

Heston Gamma

Lambda = -2



CappaV = 2

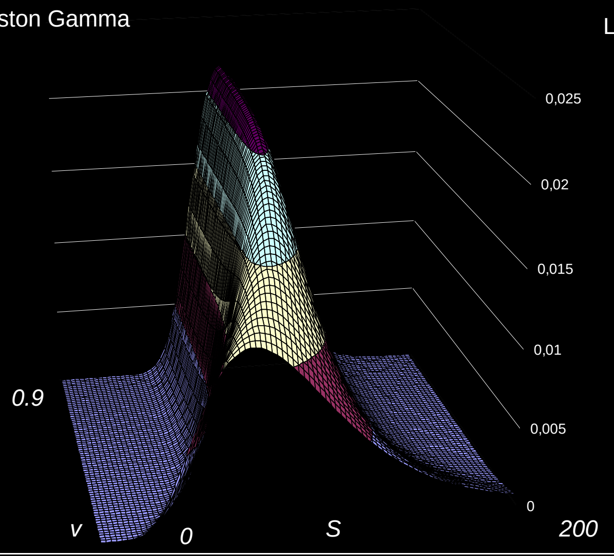
ThetaV = 0.3

EtaV = 0.1

Rho = 0

Heston Gamma

Lambda = 2



CappaV = 2

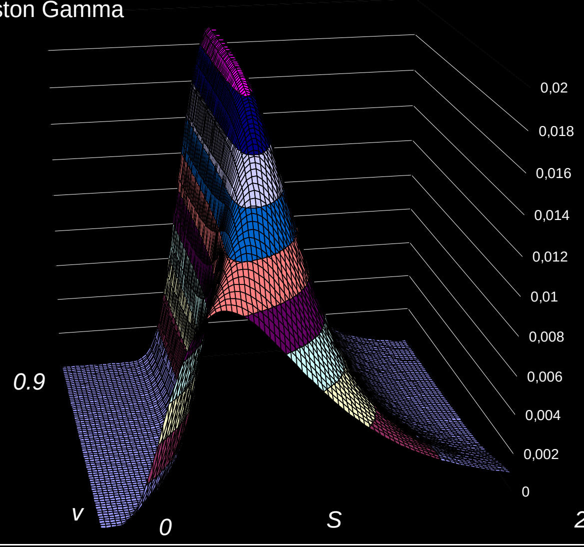
ThetaV = 0.3

EtaV = 0.1

Rho = 0

Heston Gamma

Rho = -1



CappaV = 2

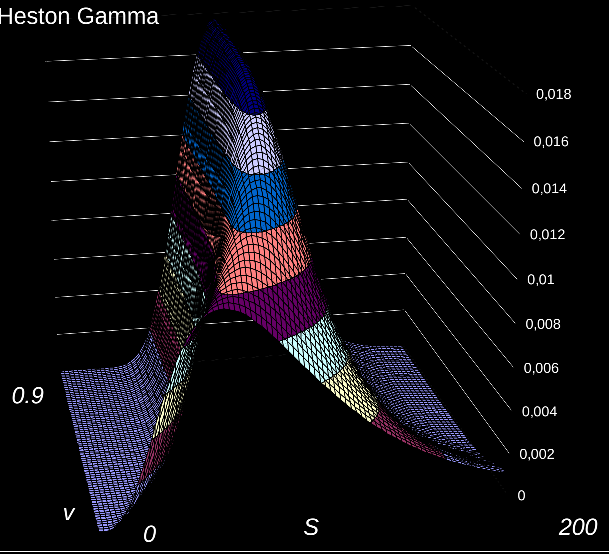
ThetaV = 0.3

EtaV = 0.1

Lambda = 0

Heston Gamma

Rho = 1



CappaV = 2

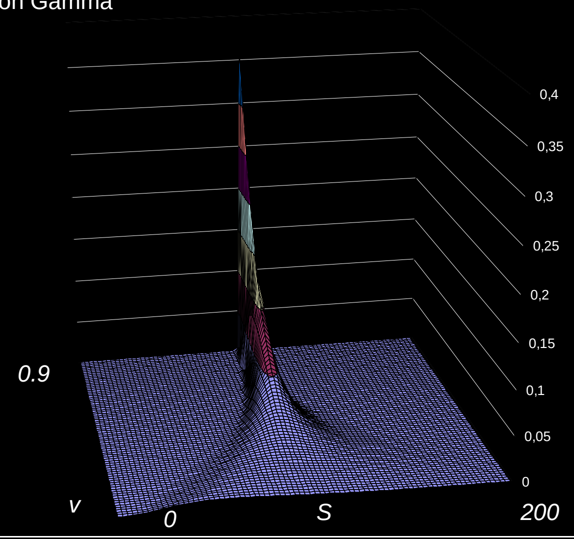
ThetaV = 0.3

EtaV = 0.1

Lambda = 0

Merton Gamma

LambdaJ = 0

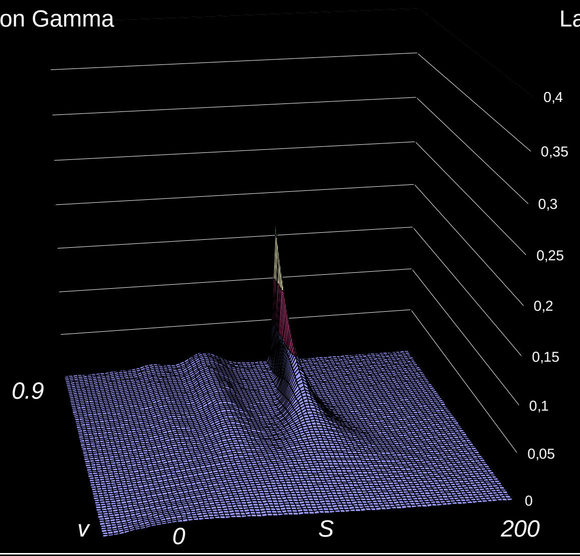


EtaJ = 0

MuJ= 0

Merton Gamma

LambdaJ = 0.9 ↑

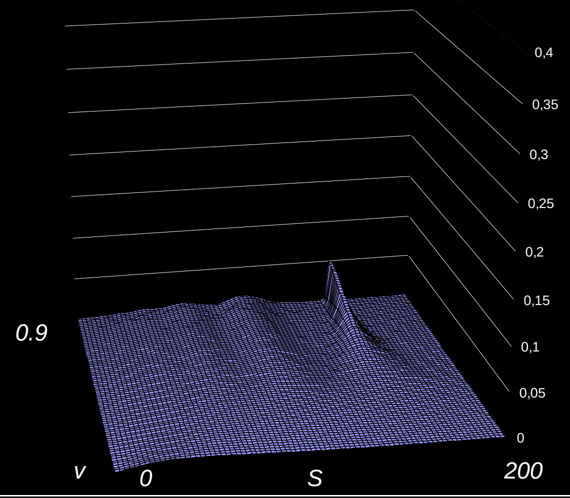


EtaJ = 0.1

MuJ = 0.5

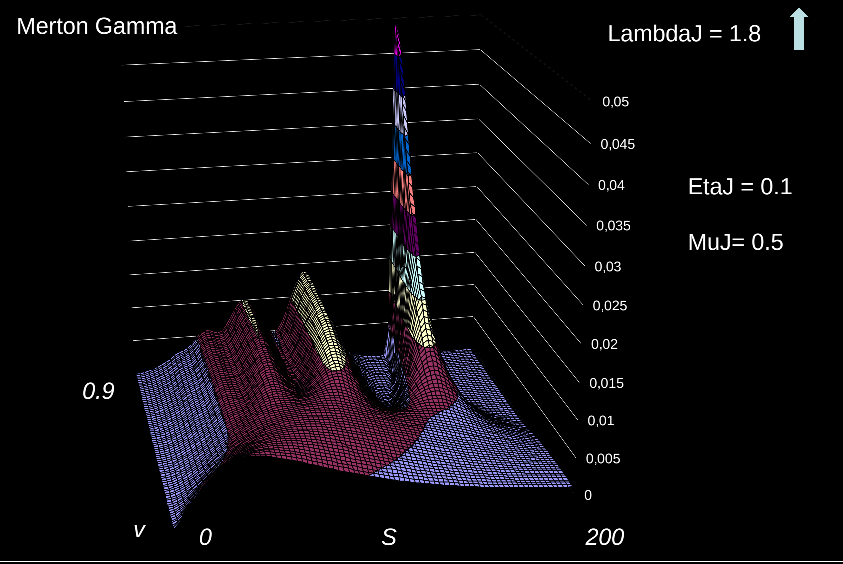
Merton Gamma

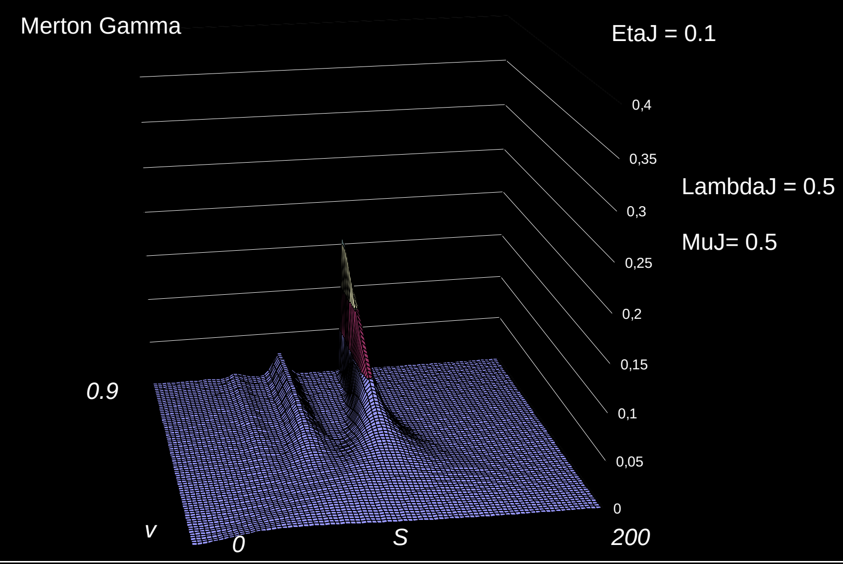
LambdaJ = 1.8 ↑

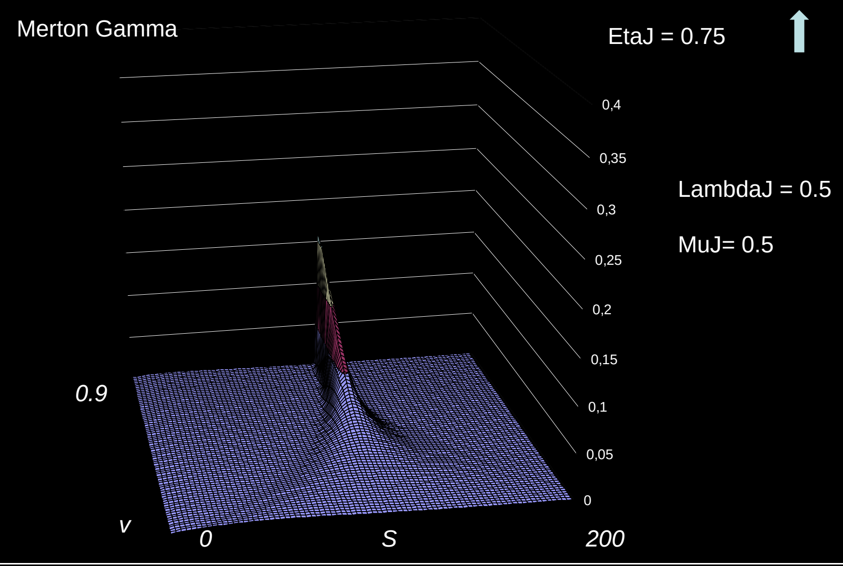


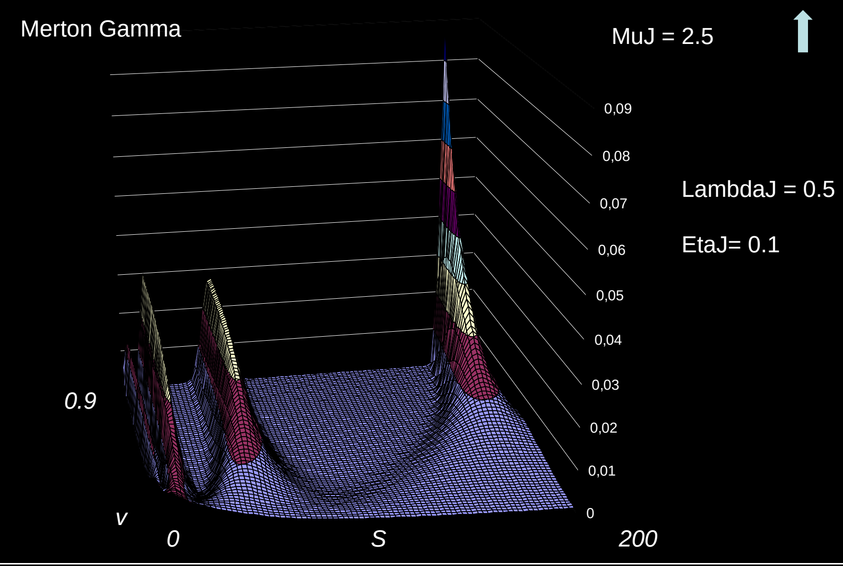
EtaJ = 0.1

MuJ = 0.5

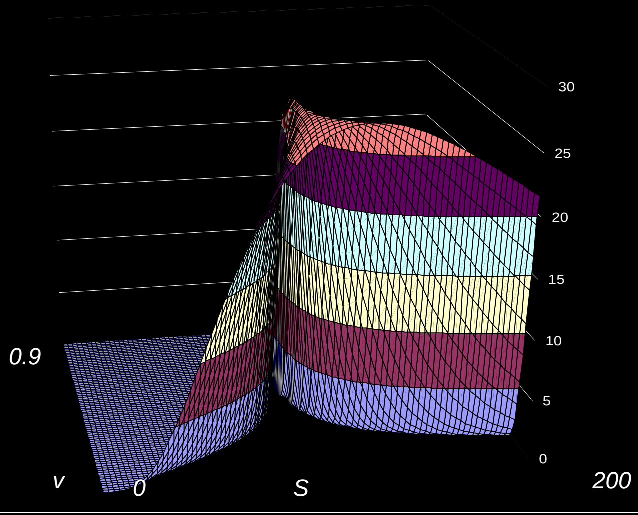

Merton Gamma
LambdaJ = 1.8
EtaJ = 0.1
MuJ= 0.5
0,05
0,045
0,04
0,035
0,03
0,025
0,02
0,015
0,01
0,005
0
0.9
v
0
S
200


Merton Gamma
EtaJ = 0.1
LambdaJ = 0.5
MuJ= 0.5
0,4
0,35
0,3
0,25
0,2
0,15
0,1
0,05
0
0.9
v
0
S
200


Merton Gamma
EtaJ = 0.75
LambdaJ = 0.5
MuJ= 0.5
0,4
0,35
0,3
0,25
0,2
0,15
0,1
0,05
0
0.9
v
0
S
200

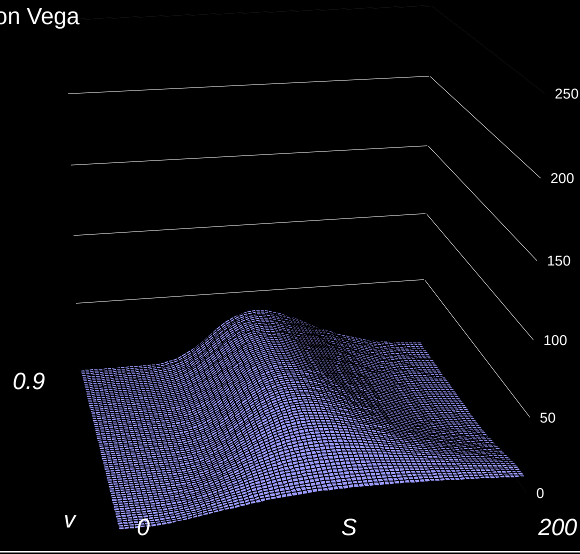

Merton Gamma
MuJ = 2.5
LambdaJ = 0.5
EtaJ= 0.1
0,09
0,08
0,07
0,06
0,05
0,04
0,03
0,02
0,01
0
0.9
v
0
S
200

Black – Scholes Vega



CONSOB

Heston Vega



Lambda = -2

CappaV = 2

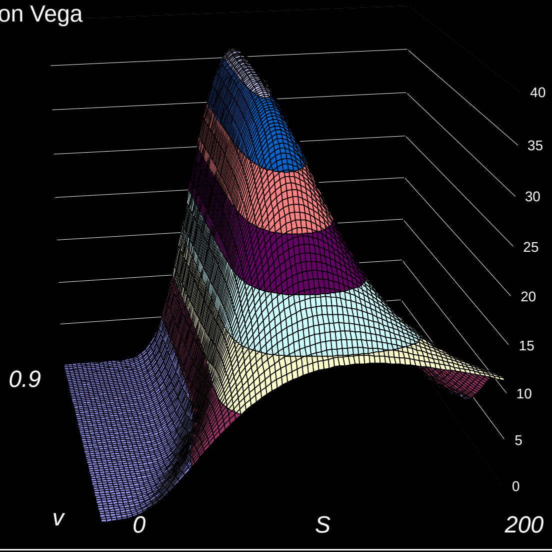
ThetaV = 0.3

EtaV = 0.1

Rho = 0

CONSOB

Heston Vega



Lambda = -2

CappaV = 2

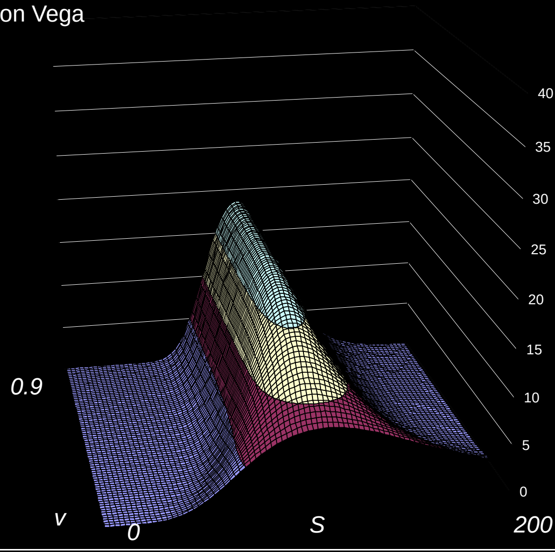
ThetaV = 0.3

EtaV = 0.1

Rho = 0

CONSOB

Heston Vega



Lambda = 2 ⬆

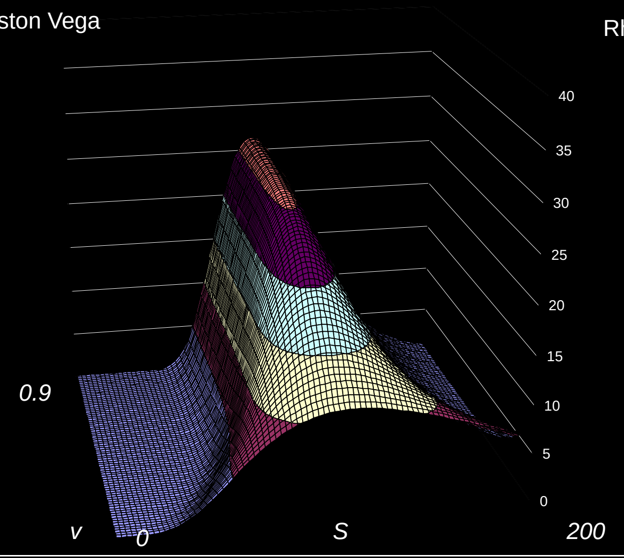
CappaV = 2

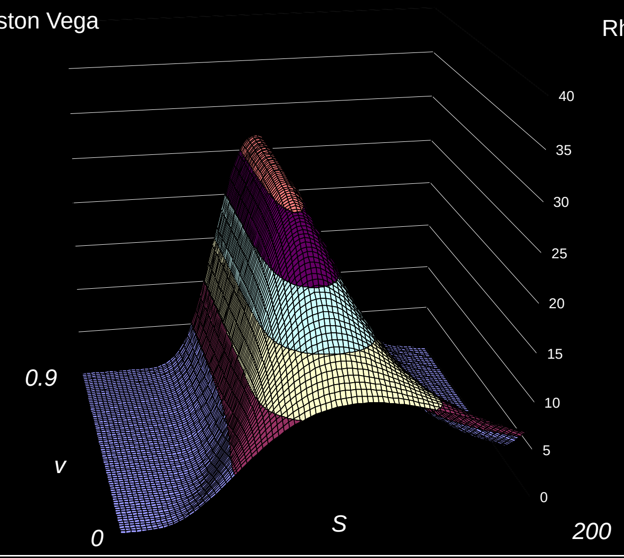
ThetaV = 0.3

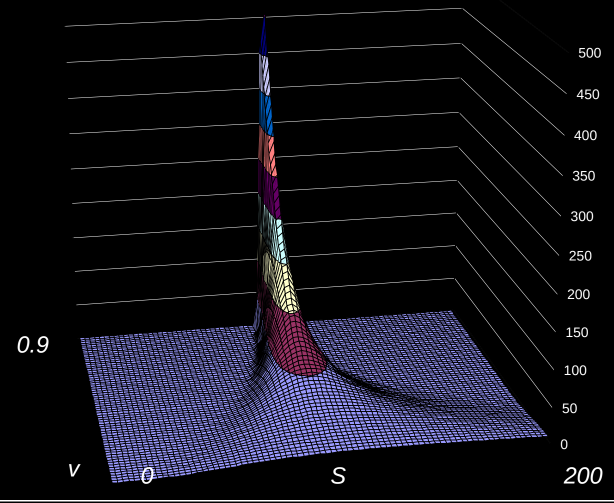
EtaV = 0.1

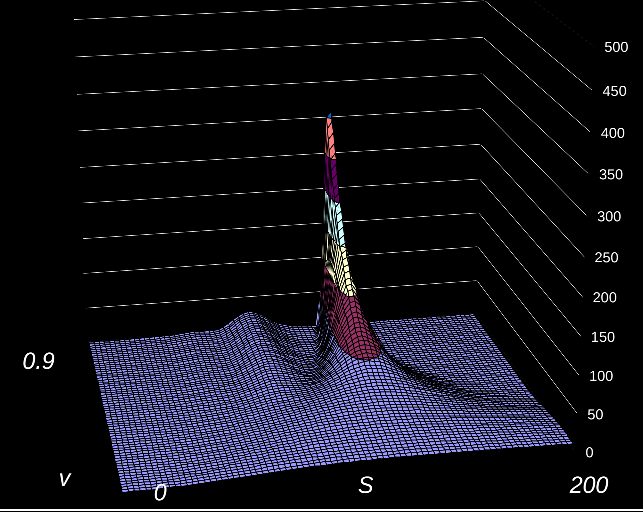
Rho = 0

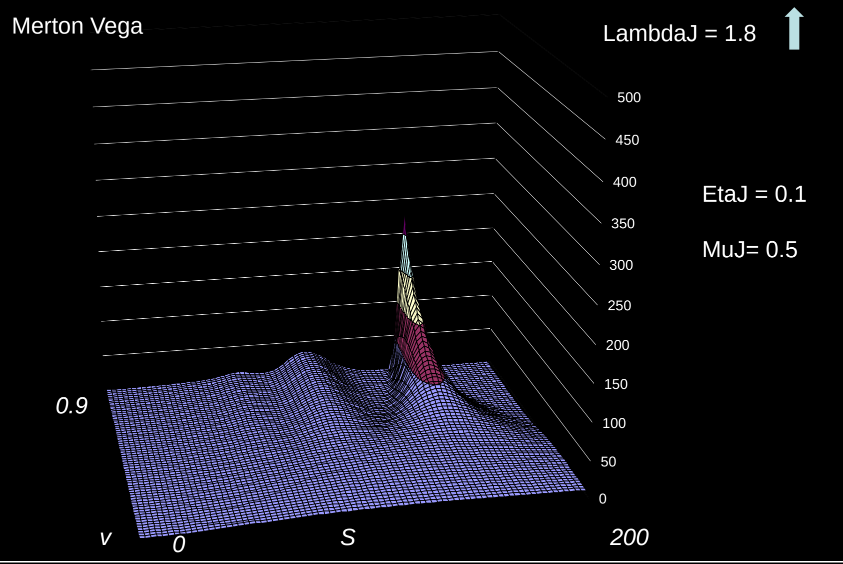
CONSOB

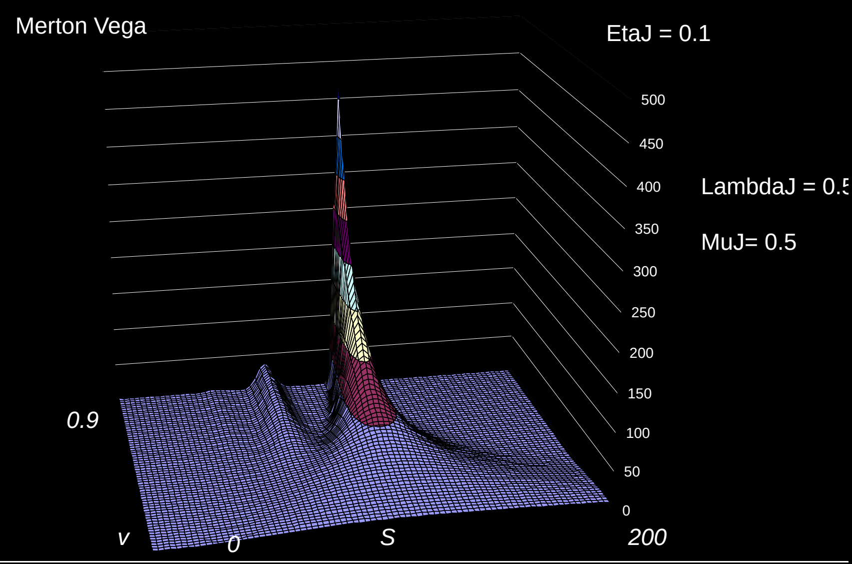

Heston Vega
Rho = -1
CappaV = 2
ThetaV = 0.3
EtaV = 0.1
Lambda = 0
40
35
30
25
20
15
10
5
0
0.9
v
0
S
200

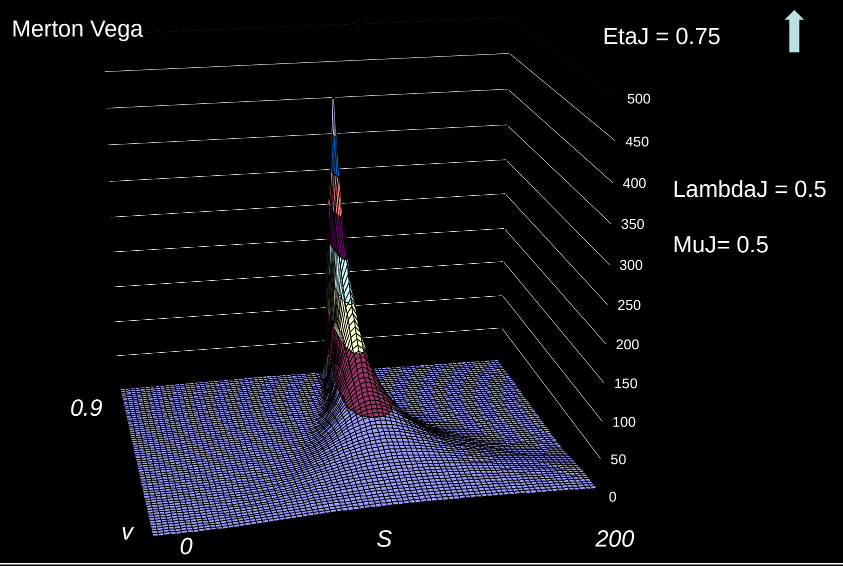

Heston Vega
Rho = 1
CappaV = 2
ThetaV = 0.3
EtaV = 0.1
Lambda = 0
40
35
30
25
20
15
10
5
0
0.9
v
0
S
200

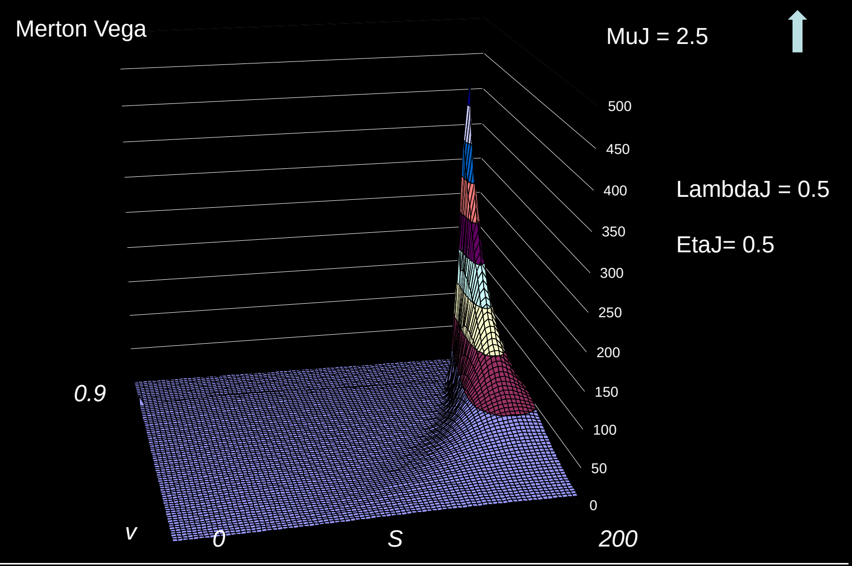

Merton Vega
LambdaJ = 0
EtaJ = 0
MuJ= 0
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200


Merton Vega
LambdaJ = 0.9
EtaJ = 0.1
MuJ = 0.5
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200

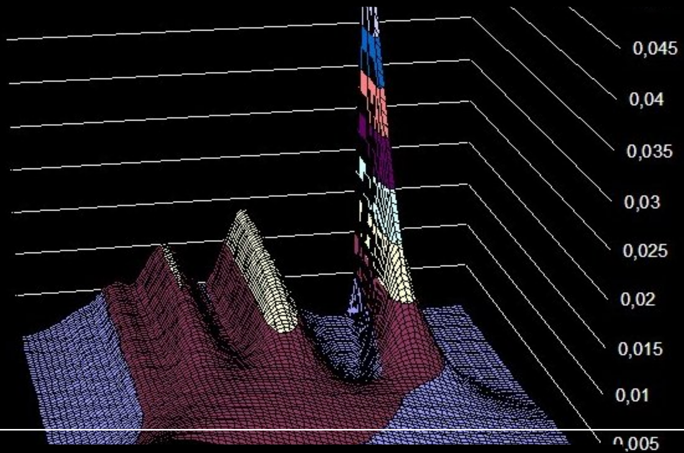

Merton Vega
LambdaJ = 1.8
EtaJ = 0.1
MuJ= 0.5
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200


Merton Vega
EtaJ = 0.1
LambdaJ = 0.5
MuJ= 0.5
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200


Merton Vega
EtaJ = 0.75
LambdaJ = 0.5
MuJ= 0.5
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200


Merton Vega
MuJ = 2.5
LambdaJ = 0.5
EtaJ= 0.5
500
450
400
350
300
250
200
150
100
50
0
0.9
v
0
S
200

Masterclass: Implementing affine jump diffusion models at the trading desk



Marcello Minenna - Paolo Verzella

Montecarlo – Derivatives and Risk Management Europe 2006