```
<< "C:\\Documents and Settings\\ocroissant\\My Documents\\NumericalIntegration.m"
```

```
<< "C:\\Documents and Settings\\ocroissant\\My Documents\\BS.m"
```

```
<< "C:\\Documents and Settings\\ocroissant\\My Documents\\Heston\\Heston.m"
```

```
Needs["PlotLegends`"]
```

The Super Biheston Spread Risks Model

Introduction

βoptimal

BiSabr, NormalHeston are common ways to represent a spread risk.The main problem is to have a consistent way to :
1) - take into account the smile of the underlyings inside an arbitrage free framework,
2) - have additional flexibility to introduce the notion of correlation smile that can be adjusted to market data
3) - be useable up to 40 years maturity options

The final test of acceptance will be of course the stability of the hedge ratios

In addition of these requirements some "nice to have" feature can also be considered:
4) - The ability to fit several maturities for the options, so allowing to have a real diffusion model, that can be used to price american deals
5)- The ability to do fit several matiurities in an autonomous way, that mean without using ime dependant coefficients, so improving forward smile behaviour, which is important for american deals
6) - The ability to have "closed form" formula for other type of options: Digitals on the underlyings, Min-Max options, Double condition options,
7) - The ability to have additional flexibility to represent smile behaviour of the tail of the underlying beyond the usual 4 parameters framework

The Super BiHeston Framework fullfill all the requirements from 1) to 7) at a certain cost (computation time and complexity) but is very likely the only one to do so

Specification of the model

notation

M : mean reversion
Σ_∞ : Long term variance
ρ : correlation underlying - variance
Σ : variance initial value
γ : frequency (Fourier parameters)
Y : Log of Underlying initial value
β : Relative vol of vol (convergence : $\beta > 1$)
τ : maturity

Variance Process

$$d\Sigma_t = (M(\Sigma_t - \Sigma_\infty) + (\Sigma_t - \Sigma_\infty) M^*)\, dt + \sqrt{\Sigma_t}\, dW_t\, Q + Q^*\, dW_t^* \left(\sqrt{\Sigma_t}\right)^*$$

Bru shows the β>1 is enough to insure non explosion of the model

$$\Omega\Omega^* = \beta\, Q^*\, Q$$

where

$$\Omega\Omega^* = -M\,\Sigma_\infty - \Sigma_\infty\, M^*$$

$$\Sigma_t = \begin{pmatrix} (\Sigma^{11})_t & (\Sigma^{12})_t \\ (\Sigma^{12})_t & (\Sigma^{22})_t \end{pmatrix}$$ 3 process of variance

we note :

$$\sqrt{\Sigma_t} = \sigma_t = \begin{pmatrix} (\sigma^{11})_t & (\sigma^{12})_t \\ (\sigma^{21})_t & (\sigma^{22})_t \end{pmatrix} \text{ such that } \Sigma_t = \sigma_t\, \sigma_t^*$$

In the 1 dim case, we compute the vol of vol by :

$$Q = \sqrt{-M\,\Sigma_\infty} \big/ \beta$$

Variance Process Parameters

Initial value of the variance : $\Sigma_0 = \begin{pmatrix} (\Sigma^{11})_0 & (\Sigma^{12})_0 \\ (\Sigma^{21})_0 & (\Sigma^{22})_0 \end{pmatrix}$

Vol of vol : $Q = \begin{pmatrix} \nu_{11} & \nu_{12} \\ \nu_{21} & \nu_{22} \end{pmatrix}$

Mean reverting speed : $M = \begin{pmatrix} \kappa_{11} & \kappa_{12} \\ \kappa_{21} & \kappa_{22} \end{pmatrix}$

Long term variance : $\Sigma_\infty = \begin{pmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{pmatrix}$

M definite negative
Q inversible
Σ_0 definite positive

Σ_∞ definite positive

Asset Returns

$$\text{Vect}\left[dS_{i,t}\right] = \sqrt{\Sigma_t}\, dZ_t$$

Covariance asset - asset volatility

$$Z_{i,t} = \text{Tr}[W_t R_i^*] + B_{i,t}\sqrt{1 - \text{Tr}[R_i R_i^*]} \quad i \in \{1, 2\}$$

such that $\|R\| \le 1$ with W_t et B_t independant

we show that affinity of the process implies that $\{\rho_1, \rho_2\}$

such that $R_1 = \begin{pmatrix} \rho_1 & \rho_2 \\ 0 & 0 \end{pmatrix}$ et $R_2 = \begin{pmatrix} 0 & 0 \\ \rho_1 & \rho_2 \end{pmatrix}$

Risk Geometry

We use Einstein summation convenrtions

soit $\sigma = \sqrt{\Sigma}$

Underlying 1 Variance : projector P1

$$\text{Var}[Y_1]\, dt = <\sigma_{1\,k}\, dZ_k, \sigma_{1\,m}\, dZ_m \ge \sigma_{1\,k}\, \sigma_{1\,m} < dZ_k, dZ_m \ge \sigma_{1\,k}\, \sigma_{1\,m}\, \delta_{km}\, dt = \sigma_{1\,k}\, \sigma_{1\,k}\, dt = \Sigma_{11}\, dt$$

Variance of Underlying 1 Variance :

we see that it does not work if we do not symmetrize Σ_{ij} dynamics : we can not get $\sigma_{1\,m}\, \sigma_{1\,m}$ in factor

on a $\Sigma_{ij} = \sigma_{ik}\, \sigma_{jk}$

$$dY_i = (\,..\,)\, dt + \sigma_{ik}\, dZ_k$$

$$d\Sigma_{jk} = (\,..\,)\, dt + \sigma_{jm}\, dW_{mn}\, Q_{nk} + Q_{mj}\, dW_{nm}\, \sigma_{kn}$$

$$< dZ_k, dW_{mn} > = R_{k,mn}\, dt = \delta_{km}\, (\delta_{n\,1}\, \rho_1 + \delta_{n2}\, \rho_2)\, dt$$

donc

$$\begin{aligned}
\text{Var}[\Sigma_{11}]\, dt = {} & < \sigma_{1\,m}\, dW_{mn}\, Q_{n1} + Q_{m1}\, dW_{nm}\, \sigma_{1\,n}, \sigma_{1\,u}\, dW_{uv}\, Q_{v1} + Q_{u1}\, dW_{vu}\, \sigma_{1\,v} > = \\
& < \sigma_{1\,m}\, dW_{mn}\, Q_{n1}, \sigma_{1\,u}\, dW_{uv}\, Q_{v1} > + \\
& < \sigma_{1\,m}\, dW_{mn}\, Q_{n1}, Q_{u1}\, dW_{vu}\, \sigma_{1\,v} > + \\
& < Q_{m1}\, dW_{nm}\, \sigma_{1\,n}, \sigma_{1\,u}\, dW_{uv}\, Q_{v1} > + \\
& < Q_{m1}\, dW_{nm}\, \sigma_{1\,n}, Q_{u1}\, dW_{vu}\, \sigma_{1\,v} >
\end{aligned}$$

$$= \sigma_{1\,m}\, Q_{n1}\, \sigma_{1\,u}\, Q_{v1} < dW_{mn}, dW_{uv} > + \sigma_{1\,m}\, Q_{n1}\, Q_{u1}\, \sigma_{1\,v} < dW_{mn}, dW_{vu} > + Q_{m1}\, \sigma_{1\,n}\, \sigma_{1\,u}\, Q_{v1} < dW_{nm}, dW_{uv} > + Q_{m1}\, \sigma_{1\,n}\, Q_{u1}\, \sigma_{1\,v} < dW_{nm}, dW_{vu} >$$

$$\begin{aligned}
= {} & \sigma_{1\,m}\, Q_{n1}\, \sigma_{1\,u}\, Q_{v1}\, \delta_{mu}\, \delta_{nv} + \\
& \sigma_{1\,m}\, Q_{n1}\, Q_{u1}\, \sigma_{1\,v}\, \delta_{mv}\, \delta_{nu} + \\
& Q_{m1}\, \sigma_{1\,n}\, \sigma_{1\,u}\, Q_{v1}\, \delta_{nu}\, \delta_{mv} + \\
& Q_{m1}\, \sigma_{1\,n}\, Q_{u1}\, \sigma_{1\,v}\, \delta_{nv}\, \delta_{mu}
\end{aligned}$$

$$= \sigma_{1\,m}\, Q_{n1}\, \sigma_{1\,m}\, Q_{n1} + \sigma_{1\,m}\, Q_{n1}\, Q_{n1}\, \sigma_{1\,m} + Q_{m1}\, \sigma_{1\,n}\, \sigma_{1\,n}\, Q_{m1} + Q_{m1}\, \sigma_{1\,n}\, Q_{m1}\, \sigma_{1\,n}$$

$$= \sigma_{1\,m}\, \sigma_{1\,m}\, (Q_{n1}\, Q_{n1} + Q_{n1}\, Q_{n1} + Q_{m1}\, Q_{m1} + Q_{m1}\, Q_{m1}) = 4\, \Sigma_{11}\, \left(Q_{11}{}^2 + Q_{12}{}^2\right)$$

Covariance entre l' underlying i et la variance de l' underlying 1 :

$$\mathrm{Covar}[Y_i, \Sigma_{11}]\, dt = \langle \sigma_{ik}\, dZ_k ,\ \sigma_{1\,u}\, dW_{uv}\, Q_{v1} + Q_{u1}\, dW_{vu}\, \sigma_{1\,v} \rangle =$$
$$\langle \sigma_{ik}\, dZ_k ,\ \sigma_{1\,u}\, dW_{uv}\, Q_{v1} \rangle + \langle \sigma_{ik}\, dZ_k ,\ Q_{u1}\, dW_{vu}\, \sigma_{1\,v} \rangle =$$
$$\sigma_{ik}\, \sigma_{1\,u}\, Q_{v1} \langle dZ_k ,\ dW_{uv} \rangle + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v} \langle dZ_k ,\ dW_{vu} \rangle =$$
$$\sigma_{ik}\, \sigma_{1\,u}\, Q_{v1}\, R_{k,uv} + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v}\, R_{k,v,u} =$$

Affinité

Il est necessaire que l covariance entre Y_i et Σ_{11} soit lineaire en Σ_{11} (et ou Y_i mais ici , cela n ' apparait pas)

donc il faut que

$\sigma_{ik}\, \sigma_{1\,u}\, Q_{v1}\, R_{k,uv} + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v}\, R_{k,v,u}$ soit une fonction de $\Sigma_{1\,i}$

c ' est a dire $Q_{v1}\, R_{k,uv}$ doit etre null pour k et u different

et en plus pour k = u ,

on doit avoir le meme coefficient quelque soit u , appelons le ρ_k

Covariance

donc

$$= \sigma_{ik}\, \sigma_{1\,u}\, Q_{v1}\, (\delta_{ku}\, (\delta_{v\,1}\, \rho_1 + \delta_{v2}\, \rho_2)) + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v}\, (\delta_{kv}\, (\delta_{u\,1}\, \rho_1 + \delta_{u2}\, \rho_2)) =$$
$$\sigma_{ik}\, \sigma_{1\,u}\, Q_{v1}\, (\delta_{ku}\, (\delta_{v\,1}\, \rho_1)) + \sigma_{ik}\, \sigma_{1\,u}\, Q_{v1}\, (\delta_{ku}\, (\delta_{v2}\, \rho_2)) + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v}\, (\delta_{kv}\, (\delta_{u\,1}\, \rho_1)) + \sigma_{ik}\, Q_{u1}\, \sigma_{1\,v}\, (\delta_{kv}\, (\delta_{u2}\, \rho_2))$$

$$= \sigma_{ik}\, \sigma_{1\,k}\, Q_{11}\, \rho_1 + \sigma_{ik}\, \sigma_{1\,k}\, Q_{21}\, \rho_2 + \sigma_{ik}\, Q_{11}\, \sigma_{1\,k}\, \rho_1 + \sigma_{ik}\, Q_{21}\, \sigma_{1\,k}\, \rho_2$$

$$= \Sigma_{1\,i}\, Q_{11}\, \rho_1 + \Sigma_{1\,i}\, Q_{21}\, \rho_2 + \Sigma_{1\,i}\, Q_{11}\, \rho_1 + \Sigma_{1\,i}\, Q_{21}\, \rho_2 = 2\, \Sigma_{1\,i}\, (Q_{11}\, \rho_1 + Q_{21}\, \rho_2)$$

Correspondance Heston / BiHeston (Risque)

$$dY_1 = (\,..\,)\, dt + \sqrt{\Sigma_1}\, dP_1$$

$$d\Sigma_1 = (\,..\,)\, dt + \nu_1 \sqrt{\Sigma_1}\, dQ_1$$

On a donc les equations suivantes :

$$\mathrm{Var}[dY_1] = \Sigma_1\, dt$$

donc c' est la meme normalization de la vol

$\mathrm{Var}[d\Sigma_1] = \nu_1^2\,\Sigma_1\,dt$

$\mathrm{Covar}[dY_1, d\Sigma_1] = \rho_{s1}\,\nu_1\,\Sigma_1$

on a donc

$$\rho_{s1}\,\nu_1 = \frac{Q_{11}\,\rho_1 + Q_{21}\,\rho_2}{2}$$

$$\rho_{s2}\,\nu_2 = \frac{Q_{12}\,\rho_1 + Q_{22}\,\rho_2}{2}$$

$$\frac{\nu_1^2}{4} = Q_{11}^2 + Q_{21}^2$$

$$\frac{\nu_2^2}{4} = Q_{12}^2 + Q_{22}^2$$

Correspondance Heston / BiHeston (Drift)

On a

$\Omega\,\Omega^* = \beta\,Q^*\,Q = \beta\,\left(Q_{11}^2 + Q_{21}^2\right)$

le drift de $d\Sigma_{11}$ est : $\left(\beta\,\left(Q_{11}^2 + Q_{21}^2\right) + 2\,(M_{11}\,\Sigma_{11} + M_{21}\,\Sigma_{12})\right)\,dt$

celui de $d\Sigma_1$ est $\lambda_1\,(\Sigma_{\infty 1} - \Sigma_1)\,dt$

donc on a les equations :

$$\beta\,\left(Q_{11}^2 + Q_{21}^2\right) + 2\,M_{21}\,\Sigma_{12} = \lambda_1\,\Sigma_{\infty 1}$$

$$2\,M_{11} = -\lambda_1$$

$$\beta\,\left(Q_{12}^2 + Q_{22}^2\right) + 2\,M_{12}\,\Sigma_{21} = \lambda_2\,\Sigma_{\infty 2}$$

$$M_{12} = \frac{\lambda_2\,\Sigma_{\infty 2} - \beta\,\left(Q_{12}^2 + Q_{22}^2\right)}{2\,\Sigma_{12}}$$

$$M_{21} = \frac{\lambda_1\,\Sigma_{\infty 1} - \beta\,\left(Q_{21}^2 + Q_{11}^2\right)}{2\,\Sigma_{12}}$$

$$2\,M_{22} = -\lambda_2$$

mais on n'oublie pas que $\Sigma_{21} = \Sigma_{12}$

On a donc , au total :

$$M = \begin{pmatrix} \frac{-\lambda_1}{2} & \frac{\lambda_2\,\Sigma_{\infty 2} - \beta\,\left(Q_{12}^2 + Q_{22}^2\right)}{2\,\Sigma_{12}} \\ \frac{\lambda_1\,\Sigma_{\infty 1} - \beta\,\left(Q_{21}^2 + Q_{11}^2\right)}{2\,\Sigma_{12}} & \frac{-\lambda_2}{2} \end{pmatrix}$$

Transformé de fourier du payoff Vanille

We start computing the option for K > 0, then we will use conversion formula to deduce the case K < 0

But the payoff does not belong to a suitable space of functions so we decompose the payoff in two and formulate a fourier transformation theory with specific shift in the fourier plane for each part :

```
PayOff[x1_, x2_, K_, α_, β_] := Max[(α ⅇ^x1 - β ⅇ^x2 - K), 0]
```

PayOff[x1, x2, K] == PayOff_1[x1, x2, K] + PayOff_2[x1, x2, K]

ou PayOff_ 1[x1, x2, K] = Max$\left[\left(e^{x1} - e^{x2} - K\right), 0\right] 1_{x2>0}$ et PayOff_ 2[x1, x2, K] = Max$\left[\left(e^{x1} - e^{x2} - K\right), 0\right] 1_{x2<0}$

CompleteFourierPayOffDroite implements the fourier transform of PayOff_ 1

CompleteFourierPayOffGauche implements the fourier transform of PayOff_ 2

The Fourier Transform of PayOff[x1, x2, K] is therefore the sum of both

The case K == 0 is much simpler. It is handled separatly.

The computations are done in annex 1

```
CompleteFourierPayOffDroite[k1_, k2_, K_] :=
  K^(ⅈ k1) / (k1 (-ⅈ + k1) k2 (-ⅈ + k2)) (K^(ⅈ k2+1) / Gamma[-ⅈ k1] (
     (1 + ⅈ k2) Gamma[1 + ⅈ k2] (Gamma[-ⅈ (k1 + k2)]) +
      ⅈ k2 Gamma[2 + ⅈ k2] (Gamma[-ⅈ (-ⅈ + k1 + k2)])) -
    (ⅈ k2 Hypergeometric2F1[-ⅈ k1, 1 + ⅈ k2, 2 + ⅈ k2, -1/K] +
     K (1 + ⅈ k2) Hypergeometric2F1[-ⅈ k1, ⅈ k2, 1 + ⅈ k2, -1/K]))
```

```
CompleteFourierPayOffDroite[k1_, k2_, K_, α_, β_] :=
 -((k1 (α - β) + ⅈ β) Hypergeometric2F1[-ⅈ k1, -ⅈ (-ⅈ + k1 + k2), -ⅈ (k1 + k2), -K]) /
    (k1 (-ⅈ + k1) (-ⅈ + k1 + k2)) -
  (K (ⅈ + k1 (-1 + α)) ⅈ Hypergeometric2F1[-ⅈ k1, -ⅈ (k1 + k2), -ⅈ (ⅈ + k1 + k2), -K]) /
    (ⅈ (k1 + k2) k1 (-ⅈ + k1))
```

```
{CompleteFourierPayOffDroite[1.1, 2.2, 0.1],
 CompleteFourierPayOffDroite[1.1, 2.2, 0.1, 1, 1]}
```

{0.180133 - 0.089615 ⅈ, 0.180133 - 0.089615 ⅈ}

```
CompleteFourierPayOffDroite[k1_, k2_] := 1 / (k1 (-ⅈ + k1) (1 + ⅈ k1 + ⅈ k2))
```

```
CompleteFourierPayOffDroite[k1_, k2_, α_, β_] := -(k1 (α - β) + ⅈ β) / (k1 (-ⅈ + k1) (-ⅈ + k1 + k2))
```

```
{CompleteFourierPayOffDroite[1.17, 2.2], CompleteFourierPayOffDroite[1.17, 2.2, 1, 1]}
```

{0.13256 - 0.0859277 ⅈ, 0.13256 - 0.0859277 ⅈ}

$$\text{CompleteFourierPayOffGauche[k1\_, k2\_, K\_]} := \frac{K^{\text{i} k1}}{k1\,(-\text{i}+k1)\,k2\,(-\text{i}+k2)}\left(\text{i}\;k2\,\text{Hypergeometric2F1}\left[-\text{i}\,k1,\,1+\text{i}\,k2,\,2+\text{i}\,k2,\,-\frac{1}{K}\right]+K\,(1+\text{i}\,k2)\,\text{Hypergeometric2F1}\left[-\text{i}\,k1,\,\text{i}\,k2,\,1+\text{i}\,k2,\,-\frac{1}{K}\right]\right)$$

$$\text{CompleteFourierPayOffGauche[k1\_, k2\_, K\_, }\alpha\text{\_, }\beta\text{\_]} := \frac{1}{(-1-\text{i}\,k1)\,k1\,k2\,(-\text{i}+k2)}K^{\text{i} k1}\left(k2\,(-\text{i}\,k1\,(\alpha-\beta)+\beta)\,\text{Hypergeometric2F1}\left[-\text{i}\,k1,\,1+\text{i}\,k2,\,2+\text{i}\,k2,\,-\frac{1}{K}\right]+K\,(-\text{i}+k2)\,(1-\text{i}\,k1\,(-1+\alpha))\,\text{Hypergeometric2F1}\left[-\text{i}\,k1,\,\text{i}\,k2,\,1+\text{i}\,k2,\,-\frac{1}{K}\right]\right)$$

$$\text{CompleteFourierPayOffGauche[k1\_, k2\_, K\_, }\alpha\text{\_, }\beta\text{\_]} := \frac{1}{(-1-\text{i}\,k1)\,k1\,k2\,(-\text{i}+k2)}\left(k2\,(-\text{i}\,k1\,(\alpha-\beta)+\beta)\left(\frac{(K)^{1+\text{i} k1+\text{i} k2}\,\text{Gamma}[-1-\text{i}\,k1-\text{i}\,k2]\,\text{Gamma}[2+\text{i}\,k2]}{\text{Gamma}[-\text{i}\,k1]}+\frac{(1+\text{i}\,k2)\;\text{Hypergeometric2F1}[-\text{i}\,k1,\,-1-\text{i}\,k1-\text{i}\,k2,\,-\text{i}\,k1-\text{i}\,k2,\,-K]}{1+\text{i}\,k1+\text{i}\,k2}\right)+(-\text{i}+k2)\;(1-\text{i}\,k1\,(-1+\alpha))\left(\frac{(K)^{1+\text{i} k1+\text{i} k2}\,\text{Gamma}[-\text{i}\,k1-\text{i}\,k2]\,\text{Gamma}[1+\text{i}\,k2]}{\text{Gamma}[-\text{i}\,k1]}+\frac{(K)\;\text{i}\,k2\;\text{Hypergeometric2F1}[-\text{i}\,k1,\,-\text{i}\,k1-\text{i}\,k2,\,1-\text{i}\,k1-\text{i}\,k2,\,-K]}{\text{i}\,k1+\text{i}\,k2}\right)\right)$$

$$\text{CompleteFourierPayOffGauche[k1\_, k2\_, K\_, }\alpha\text{\_, }\beta\text{\_]} := \text{Module}\Big[\Big\{a0 = (K)^{1+\text{i} k1+\text{i} k2},\; g1 = \text{Gamma}[-1-\text{i}\,k1-\text{i}\,k2],\; g2 = \text{Gamma}[1+\text{i}\,k2],\; g3 = \text{Gamma}[-\text{i}\,k1]\Big\},\; \frac{1}{(-1-\text{i}\,k1)\,k1\,k2\,(-\text{i}+k2)}\left(k2\,(-\text{i}\,k1\,(\alpha-\beta)+\beta)\left(\frac{a0\,g1\,g2\,(1+\text{i}\,k2)}{g3}+\frac{(1+\text{i}\,k2)\;\text{Hypergeometric2F1}[-\text{i}\,k1,\,-1-\text{i}\,k1-\text{i}\,k2,\,-\text{i}\,k1-\text{i}\,k2,\,-K]}{1+\text{i}\,k1+\text{i}\,k2}\right)+(-\text{i}+k2)\;(1-\text{i}\,k1\,(-1+\alpha))\left(\frac{a0\,g1\,(-1-\text{i}\,k1-\text{i}\,k2)\,g2}{g3}+\frac{(K)\;\text{i}\,k2\;\text{Hypergeometric2F1}[-\text{i}\,k1,\,-\text{i}\,k1-\text{i}\,k2,\,1-\text{i}\,k1-\text{i}\,k2,\,-K]}{\text{i}\,k1+\text{i}\,k2}\right)\right)\Big]$$

```
{CompleteFourierPayOffGauche[1.17, 2.2, 0.1],
 CompleteFourierPayOffGauche[1.17, 2.2, 0.1, 1, 1]}
```

{-0.158254 + 0.0845554 i, -0.158254 + 0.0845554 i}

$$\text{CompleteFourierPayOffGauche[k1\_, k2\_]} := \frac{-1}{k1\,(-\text{i}+k1)\,(1+\text{i}\,k1+\text{i}\,k2)}$$

```
CompleteFourierPayOffGauche[k1_, k2_, α_, β_] := (k1 (α - β) + ⅈ β) / (k1 (-ⅈ + k1) (-ⅈ + k1 + k2))
```

```
{CompleteFourierPayOffGauche[1.17, 2.2], CompleteFourierPayOffGauche[1.17, 2.2, 1, 1]}
```

```
{-0.13256 + 0.0859277 ⅈ, -0.13256 + 0.0859277 ⅈ}
```

When the strike get higher than 1 , we can use the following rule :

```
RuleHyper = {Hypergeometric2F1[a_, b_, c_, z_] →
    (Gamma[b - a] Gamma[c]) / (Gamma[b] Gamma[c - a]) (-z)^-a Hypergeometric2F1[a, a - c + 1, a - b + 1, 1/z] +
     (Gamma[a - b] Gamma[c]) / (Gamma[a] Gamma[c - b]) (-z)^-b Hypergeometric2F1[b, b - c + 1, b - a + 1, 1/z]};
```

Laplace transform of the propagator of the process

by Feynman Kac :

$$L = TR\left[(\Omega\,\Omega^{*} + M\,\Sigma + \Sigma\,M)\,D + \frac{\Sigma\,D\,Q^{*}\,Q\,D}{2}\right] + \frac{1}{2}\nabla_Y\Sigma\ \nabla_Y{}^{*} + Tr[D\,Q^{*}\,\rho\,\nabla_Y\,\Sigma] - \frac{1}{2}\,Vec[\Sigma_{ii}]\,\nabla_Y$$

the security price verifies :

$$\begin{cases} -\frac{\partial P}{\partial t} = L\,p \\ p[0] = \left(S_1\,e^{Y_1} - S_2\,e^{Y_2} - K\right)^{+} \end{cases}$$

then

$$E\left[\left(S_1\,e^{Y_{1,T}} - S_2\,e^{Y_{2,T}} - K\right)^{+}\right] = \frac{S_1\,S_2}{(2\,\pi)^2}\int_{i\,\gamma-\infty}^{i\,\gamma+\infty} e^{-2\,i\,(Y_1\,\gamma_1 + Y_2\,\gamma_2)}\,q[-i\,\gamma_1, -i\,\gamma_2]\,d\gamma_1\,d\gamma_2$$

we assume for non explosion purposes (see Bru 1987)

$$Q\,Q^{*} = \frac{-2\,M\,\Sigma inf}{\beta}$$

The laplace transform of the probability transition is affine with coefficients A and c
$q[\gamma_1, \gamma_2] == E[\exp[<\gamma, Y>]] == \exp\left[Tr[A.\Sigma] + \gamma^{T}.Y + c\right]$

we can show (see Fonseca 2007)

$$\begin{cases} \frac{\partial c[\tau]}{\partial \tau} = M\,\Theta\,A[\tau] \\ \frac{\partial A[\tau]}{\partial \tau} = \frac{Q\,Q^{*}}{2}\,A[\tau]^2 + A[\tau]\,M + M^{*}\,A[\tau] + \gamma\,\rho^{*}\,Q\,A[\tau] + A[\tau]\,Q^{*}\,\rho\,\gamma^{*} + \frac{\gamma}{:} \end{cases}$$

that we can solve by linearizing the flow when we double the space dimension.

Solution of a Riccati

We want to solve : $dA = (A\,a2\,A + a1\,A + A\,a1s + a0)\,dt$

let $A[\tau] = F^{-1} G$ soit $G = F A$

Then $d[F A] = dF\, A + F\, dA = dG$

$dA = (A\, a2\, A + a1\, A + A\, a1s + a0)\, dt$

$dG = F\ dA + dF\, A = (F\, A\, a2\, A + F\, a1\, A + F\, A\, a1s + F\, a0)\, dt + dF\ A$

$dG = (G\, a2\, A + F\, a1\, A + G\, a1s + F\, a0)\, dt + dF\ A$

We split:

$(G\, a1s + F\, a0)\, dt = dG$

$(G\, a2 + F\, a1)\, dt + dF = 0$

then $dF = (-\ F\, a1 - G\, a2)\, dt$ et $dG = (G\, a1s\ +\ F\, a0)\, dt$

therefore $d\,(F, G) = (F, G) \cdot \begin{pmatrix} -a1 & a0 \\ -a2 & a1s \end{pmatrix} dt$

and we can use matrix exponentiation to deduce the solution of the system

Computation of A

$\{F, G\} \cdot \begin{pmatrix} -a1 & a0 \\ -a2 & a1s \end{pmatrix}$

{-a1 F - a2 G, a0 F + a1s G}

$a2 = \frac{Q\, Q^*}{2}$; $a1 = M^* + \gamma\, \rho^*\, Q$; $a1s = M + Q^*\, \rho\, \gamma^*$; $a0 = \frac{\gamma\, \gamma^*}{2}$

initial condition A[0] = 0 soit F = 1 et G = 0

then $\{F, G\} = \{1, 0\} \cdot \mathrm{Exp}\left[\begin{pmatrix} -a1 & a0 \\ -a2 & a1s \end{pmatrix}\right]$

then $\begin{pmatrix} F \\ G \end{pmatrix} = \mathrm{Transpose}\left[\mathrm{Exp}\left[\begin{pmatrix} -a1 & a0 \\ -a2 & a1s \end{pmatrix}\right]\right] \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

then $\begin{pmatrix} F \\ G \end{pmatrix} = \mathrm{Exp}\left[\mathrm{Transpose}\left[\begin{pmatrix} -a1 & a0 \\ -a2 & a1s \end{pmatrix}\right]\right] \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \mathrm{Exp}\left[\begin{pmatrix} -a1 & -a2 \\ a0 & a1s \end{pmatrix}\right] \cdot \begin{pmatrix} 1 \\ 0 \end{pmatrix}$

{{a11, a12}, {a21, a22}} . {1, 0}

{a11, a21}

$A = a11^{-1}\, a21$

Computation of c

$dc = \mathrm{Tr}\left[-\beta^2\, QQ^*\, A\right] dt$

but

$dG = (G\, a1s\ +\ F\, a0)\, dt = \left(F\, \frac{\gamma\, \gamma}{2} + G\left(\frac{M + Q^*\, \rho\, \gamma^*}{2}\right)\right) dt$

$dF = (-\ F\, a1\ -\ G\, a2)\, dt = -F\left(\frac{M^* + \gamma\, R\, Q}{2}\right) - G\, \frac{Q^*\, Q}{2}$

where $a2 = \frac{Q\,Q^*}{2}$; $a1 = M^* + \gamma\,\rho^*\,Q$; $a1s = M + Q^*\,\rho\,\gamma^*$; $a0 = \frac{\gamma\,\gamma^*}{2}$

we get:

$$G = -2\left(\frac{dF}{dt} + F\left(\frac{M^* + \gamma\,R\,Q}{2}\right)\right)\,(Q^*\,Q)^{-1}$$

we multiplies by F

$$A = F^{-1}\,G = -2\left(F^{-1}\,\frac{dF}{dt} + \left(\frac{M^* + \gamma\,R\,Q}{2}\right)\right)\,(Q^*\,Q)^{-1}$$

then

$$dc = -2\,\beta^2\,\mathrm{Tr}\left[\left(F^{-1}\,\frac{dF}{d\tau} + \left(\frac{M^* + \gamma\,R\,Q}{2}\right)\right)\right]\,d\tau$$

Using trace properties

$$c = -2\,\beta^2\,\mathrm{Tr}\left[\left(\mathrm{Log}[F] + \left(\frac{M^* + \gamma\,R\,Q}{2}\right)\tau\right)\right]$$

Handling of the gap at the money

Due to integration difficulty , the call put conversion relationship which is used to compute the K < 0 cases can lead to discontinuities of the smile at the money and if we handle the discontinuity, to discontinuity of the derivatives. To take care of them, we start from the T = 0 relationship that should have no discontinuity :

$$(S1 - S2 - K)^+ = (S1 - S2 - K) + (S2 - S1 - (-K))^+$$

Obviously this relationship is continuous and has the same continuous derivatives in K > 0

$$\left(e^{x1} - e^{x2} - K\right)^+ = \left(e^{x1} - e^{x2}\right)^+ - \left(e^{x2} - e^{x1}\right)^+ - K + \left(e^{x2} - e^{x1} - (-K)\right)^+$$

let $f[x1, x2, K] \equiv$

$$1_{K>0}\left(e^{x1} - e^{x2} - K\right)^+ + 1_{K<0}\left(\left(e^{x1} - e^{x2}\right)^+ - \left(e^{x2} - e^{x1}\right)^+ - K + \left(e^{x2} - e^{x1} - (-K)\right)^+\right)$$

we have

$$f[x1, x2, K] == \left(e^{x1} - e^{x2} - K\right)^+ \quad \text{for all } K \in \mathbb{R}$$

but we use the definition of F to compute its fourier transform

Theory of special fourier operators to handle the K > 0 cases

Let $\mathcal{F}_{\lambda1,\lambda2}$ the special fourier transform that regularize with λ1 and λ2 the functions like $g[x1, x2] = \left(e^{x1} - e^{x2} - K\right)^+$ and make them convergent :

$$\mathcal{F}_{\lambda1,\lambda2}[g][k1, k2] = \begin{pmatrix} \int_{-\infty}^{\infty}\left(e^{i\,(k1 + i\,\lambda1)\,x1 + i\,(k2 - i\,\lambda2)\,x2}\,1_{x2<0}\right) g[x1, x2]\,dx1\,dx2 \\ \int_{-\infty}^{\infty}\left(e^{i\,(k1 + i\,\lambda1)\,x1 + i\,(k2 + i\,\lambda2)\,x2}\,1_{x2>0}\right) g[x1, x2]\,dx1\,dx2 \end{pmatrix}$$

We have the inverse fourier transform :

$$\mathcal{G}_{\lambda1,\lambda2}\left[\begin{pmatrix} h1 \\ h2 \end{pmatrix}\right][x1, x2] = \frac{1}{2\,\pi}\int_{-\infty}^{\infty}\left(e^{-i\,(k1 + i\,\lambda1)\,x1 - i\,(k2 - i\,\lambda2)\,x2}\right) h1[k1, k2]\,dk1\,dk2 +$$

$$\frac{1}{2\pi}\int_{-\infty}^{\infty}\left(e^{-i\,(k1+i\,\lambda 1)\,x1-i\,(k2+i\,\lambda 2)\,x2}\right)h2[k1,\,k2]\,dk1\,dk2$$

that we can justify by saying that we
analyze separately the $1_{x2<0}$ part and the $1_{x2>0}$ part

we can check that

$\mathcal{G}_{\lambda 1,\lambda 2}\circ\mathcal{F}_{\lambda 1,\lambda 2}[g] = g$

let

$g_c[K] = \left(e^{x1}-e^{x2}-K\right)^{+}$

$g_p[K] = \left(e^{x1}-e^{x2}\right)^{+}-\left(e^{x2}-e^{x1}\right)^{+}-K+\left(e^{x2}-e^{x1}-(-K)\right)^{+}$

$g_f = \left(e^{x1}-e^{x2}\right)^{+}-\left(e^{x2}-e^{x1}\right)^{+}$

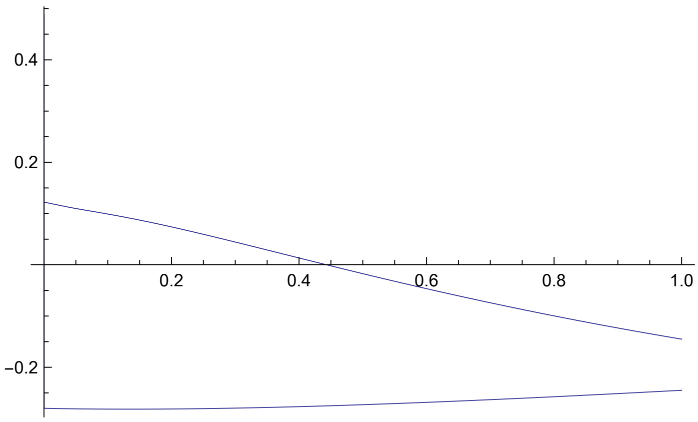
having analytic closed forms for $\mathcal{F}_{\lambda 1,\lambda 2}[g_c]$:

$\mathcal{F}_{\lambda 1,\lambda 2}[g_c] ==$

$$\left(\begin{array}{l} \frac{K^{i\,k1}\left(i\ k2\,\text{Hypergeometric2F1}\left[-i\,k1,1+i\,k2,2+i\,k2,-\frac{1}{K}\right]+K\,(1+i\,k2)\,\text{Hypergeometric2}\right.}{k1\,(-i+k1)\,k2\,(-i+k2)} \\ \frac{K^{i\,k1}}{k1\,(-i+k1)\,k2\,(-i+k2)}\left(\frac{K^{i\,k2+1}}{\text{Gamma}[-i\,k1]}\left((1+i\,k2)\,\text{Gamma}[1+i\,k2]\,(\text{Gamma}[-i\,(k1+k2)])+\right.\right. \\ \quad\left(i\ k2\,\text{Hypergeometric2F1}\left[-i\,k1,\,1+i\,k2,\,2+i\,k2,\,-\frac{1}{K}\right]+K\,(1+i\,k2)\,\text{Hyperge}\right. \end{array}\right)$$

$$\mathcal{F}_{\lambda 1,\lambda 2}[g_f] == \left(\begin{array}{l} -\frac{1}{k1\,(-i+k1)\,(1+i\,k1+i\,k2)}\ \ /.\ \{k1\to k1+i\,\lambda 1,\ k2\to k2-i\,\lambda 2\} \\ \frac{1}{k1\,(-i+k1)\,(1+i\,k1+i\,k2)}\ \ /.\ \{k1\to k1+i\,\lambda 1,\ k2\to k2+i\,\lambda 2\} \end{array}\right)$$

```
Fλ1λ2gc[k1_, k2_, λ1_, λ2_, K_] :=
 {(K^(ⅈ (k1+ⅈ λ1)) (ⅈ (k2 - ⅈ λ2) Hypergeometric2F1[-ⅈ (k1 + ⅈ λ1),
        1 + ⅈ (k2 - ⅈ λ2), 2 + ⅈ (k2 - ⅈ λ2), -1/K] + K (1 + ⅈ (k2 - ⅈ λ2))
       Hypergeometric2F1[-ⅈ (k1 + ⅈ λ1), ⅈ (k2 - ⅈ λ2), 1 + ⅈ (k2 - ⅈ λ2), -1/K])) /
   ((k1 + ⅈ λ1) (-ⅈ + (k1 + ⅈ λ1)) (k2 - ⅈ λ2) (-ⅈ + (k2 - ⅈ λ2))),
  1/((k1 + ⅈ λ1) (-ⅈ + (k1 + ⅈ λ1)) (k2 + ⅈ λ2) (-ⅈ + (k2 + ⅈ λ2)))
   K^(ⅈ (k1+ⅈ λ1)) (1/Gamma[-ⅈ (k1 + ⅈ λ1)] K^(ⅈ (k2+ⅈ λ2)+1)
       ((1 + ⅈ (k2 + ⅈ λ2)) Gamma[1 + ⅈ (k2 + ⅈ λ2)] Gamma[-ⅈ ((k1 + ⅈ λ1) + (k2 + ⅈ λ2))] +
         ⅈ (k2 + ⅈ λ2) Gamma[2 + ⅈ (k2 + ⅈ λ2)] Gamma[-ⅈ (-ⅈ + (k1 + ⅈ λ1) + (k2 + ⅈ λ2))]) -
      (ⅈ (k2 + ⅈ λ2) Hypergeometric2F1[-ⅈ (k1 + ⅈ λ1), 1 + ⅈ (k2 + ⅈ λ2),
          2 + ⅈ (k2 + ⅈ λ2), -1/K] + K (1 + ⅈ (k2 + ⅈ λ2))
         Hypergeometric2F1[-ⅈ (k1 + ⅈ λ1), ⅈ (k2 + ⅈ λ2), 1 + ⅈ (k2 + ⅈ λ2), -1/K]))}

Fλ1λ2gf[k1_, k2_, λ1_, λ2_] :=
 {-1/((k1 + ⅈ λ1) (-ⅈ + (k1 + ⅈ λ1)) (1 + ⅈ (k1 + ⅈ λ1) + ⅈ (k2 - ⅈ λ2))),
  1/((k1 + ⅈ λ1) (-ⅈ + (k1 + ⅈ λ1)) (1 + ⅈ (k1 + ⅈ λ1) + ⅈ (k2 + ⅈ λ2)))}

Module[{k1 = 1, k2 = 1, λ1 = 1.1, λ2 = 1.2}, Fλ1λ2gf[k1, k2, λ1, λ2]]
{0.122192+ 0.266569 ⅈ, -0.280064 - 0.0173219 ⅈ}

Module[{k1 = 1, k2 = 1, λ1 = 1.1, λ2 = 1.2}, Fλ1λ2gc[k1, k2, λ1, λ2, 0.0001]]
{0.122168+ 0.266592 ⅈ, -0.280067 - 0.0173439 ⅈ}

Module[{k1 = 1, k2 = 1, λ1 = 1.1, λ2 = 1.2},
 Plot[Re[Fλ1λ2gc[k1, k2, λ1, λ2, K]], {K, 0, 1}]]
```



on a bien convergence

$\lim_{K\to 0} \mathcal{F}_{\lambda1,\lambda2}[g_c[K]] = \mathcal{F}_{\lambda1,\lambda2}[g_f]$

Differentiability at K = 0

$$\frac{\partial g_c[K]}{\partial K} = \frac{\partial\left(e^{x1}-e^{x2}-K\right)^+}{\partial K} = \frac{\partial\left(\left(e^{x1}-e^{x2}-K\right)\theta\left[e^{x1}-e^{x2}-K\right]\right)}{\partial K} =$$

$$\frac{\partial\left(e^{x1}-e^{x2}-K\right)}{\partial K}\theta\left[e^{x1}-e^{x2}-K\right] + \left(e^{x1}-e^{x2}-K\right)\frac{\partial\left(\theta\left[e^{x1}-e^{x2}-K\right]\right)}{\partial K} =$$

$$-\theta\left[e^{x1}-e^{x2}-K\right] - \left(e^{x1}-e^{x2}-K\right)\delta\left[e^{x1}-e^{x2}-K\right]$$

assume two different definitions of f ,for K>0 and K<0 :

f[K] = θ[K] fp[K] + (1 - θ[K]) fn[K] = fn[K] + θ[K] (fp[K] - fn[K])

$$\frac{\partial f[K]}{\partial K} = \frac{\partial fn[K]}{\partial K} + \theta[K]\frac{\partial(fp[K]-fn[K])}{\partial K} + (fp[K]-fn[K])\times\delta[K]$$

we have fn[x1, x2, K] = ΔS - K + fp[x2, x1, -K]

So

$$\frac{\partial f[K]}{\partial K} = -1 - \frac{\partial fp[x2, x1, -K]}{\partial K} + \theta[K]\frac{\partial(fp[x1, x2, K]+1+fp[x2, x1, -K])}{\partial K} +$$
$$(fp[x1, x2, K] - (\Delta S - K + fp[x2, x1, -K]))\times\delta[K]$$

$$\frac{\partial f[K]}{\partial K} = \theta[K]\frac{\partial fp[x1, x2, K]}{\partial K} - (1-\theta[K])\frac{\partial fp[x2, x1, -K]}{\partial K} +$$
$$(fp[x1, x2, K] - (\Delta S - K + fp[x2, x1, -K]))\times\delta[K]$$

$$\frac{\partial f[K]}{\partial K} = 1_{K>0}\frac{\partial fp[x1, x2, K]}{\partial K} - 1_{K<0}\frac{\partial fp[x2, x1, -K]}{\partial K} +$$
$$(fp[x1, x2, K] - (\Delta S - K + fp[x2, x1, -K]))\times\delta[K]$$

let suppose ΔS = fp[x1, x2, 0] - fp[x2, x1, 0]

then

$$\left.\frac{\partial f[x1, x2, K]}{\partial K}\right|_{0+} = \left.\frac{\partial f[x1, x2, K]}{\partial K}\right|_{0-} + \left(\frac{\partial fp[x1, x2, 0]}{\partial K} - \frac{\partial fp[x2, x1, 0]}{\partial K}\right)$$

but we have $fp[x1, x2, K] = g_c[K]$

$$\frac{\partial g_c[K]}{\partial K} = \frac{\partial\left(e^{x1}-e^{x2}-K\right)^+}{\partial K} = \frac{\partial\left(\left(e^{x1}-e^{x2}-K\right)\theta\left[e^{x1}-e^{x2}-K\right]\right)}{\partial K} =$$

$$\frac{\partial\left(e^{x1}-e^{x2}-K\right)}{\partial K}\theta\left[e^{x1}-e^{x2}-K\right] + \left(e^{x1}-e^{x2}-K\right)\frac{\partial\left(\theta\left[e^{x1}-e^{x2}-K\right]\right)}{\partial K} =$$

$$-\theta\left[e^{x1}-e^{x2}-K\right] - \left(e^{x1}-e^{x2}-K\right)\delta\left[e^{x1}-e^{x2}-K\right]$$

and

$$\left(\frac{\partial fp[x1, x2, 0]}{\partial K} - \frac{\partial fp[x2, x1, 0]}{\partial K}\right) =$$
$$-\theta\left[e^{x1}-e^{x2}\right] - \left(e^{x1}-e^{x2}\right)\delta\left[e^{x1}-e^{x2}\right] - \left(-\theta\left[e^{x2}-e^{x1}\right] - \left(e^{x2}-e^{x1}\right)\delta\left[e^{x2}-e^{x1}\right]\right)$$

let $\Delta S = e^{x1} - e^{x2}$

$$\left(\frac{\partial fp[x1, x2, 0]}{\partial K} - \frac{\partial fp[x2, x1, 0]}{\partial K}\right) = -\theta[\Delta S] - (\Delta S)\,\delta[\Delta S] - (-\theta[-\Delta S] - (-\Delta S)\,\delta[-\Delta S]) =$$
$$-\theta[\Delta S] - (\Delta S)\,\delta[\Delta S] - (-(1-\theta[\Delta S]) - (-\Delta S)\,\delta[\Delta S]) =$$
$$-\theta[\Delta S] - \Delta S\delta[\Delta S] - (-1 + \theta[\Delta S] + \Delta S\,\delta[\Delta S]) = -\theta[\Delta S] - \Delta S\delta[\Delta S] + 1 - \theta[\Delta S] - \Delta S\,\delta[\Delta S] = 1$$

```
f[ΔS_] := -HeavisideTheta[ΔS] - ΔS DiracDelta[ΔS]
```

```
Simplify[f[ΔS] - f[-ΔS]]
```

```
HeavisideTheta[-ΔS] - HeavisideTheta[ΔS]
```

$$= 1 - 2\,\theta[\Delta S]$$

```
f[S_, K_] := Max[S - K, 0]
```

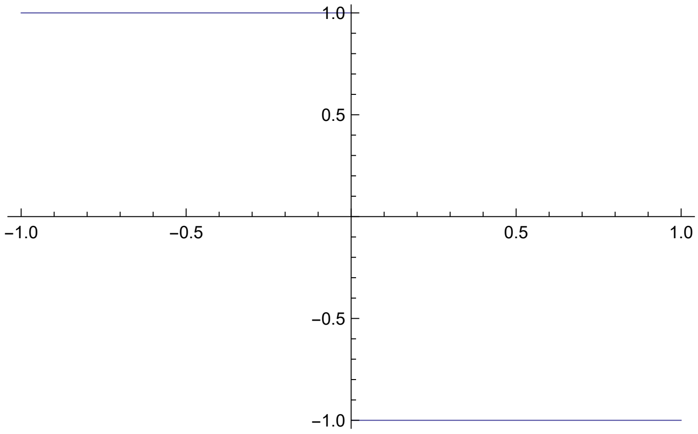
```
h = (D[f[S, K], K] - D[f[-S, K], K] ) /. K → 0
```

$$\begin{cases} -1 & -S < 0 \end{cases} - \begin{cases} -1 & S < 0 \end{cases}$$

```
FullForm[h]
```

```
Plus[Piecewise[List[List[-1, Less[Times[-1, S], 0]]], 0],
 Times[-1, Piecewise[List[List[-1, Less[S, 0]]], 0]]]
```

```
Plot[h, {S, -1, 1}]
```



Implementation

Therefore we can write a first version of the propagator

```
SuperBiHestonLaplaceTransform[M_, {{θ11_, θ12_}, {θ12_, θ22_}},
  ρ_, {{Σ11_, Σ12_}, {Σ12_, Σ22_}}, γ_, β_, τ_] :=
 Module[{H, EXPH, A11, A21, A, Q, ν11, ν12, ν21, ν22,
    c, Σ = (Σ11 Σ12
            Σ12 Σ22), ΣinfM = (θ11 θ12
                                θ12 θ22),
    Placement1 = {{1, 0, 0, 0}, {0, 1, 0, 0}}, Placement2 = {{0, 0, 1, 0}, {0, 0, 0, 1}}},
   Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM. Transpose[M])/2)] / β;

   H = Transpose[Placement1] . ((- (M + Transpose[Q].Outer[Times, ρ, γ]))/2).Placement1 +
     Transpose[Placement1] .(-Transpose[Q]. Q / 2).Placement2 +
     Transpose[Placement2] .
      (1/2 (Outer[Times, γ, γ] - DiagonalMatrix[γ] )).Placement1 +
     Transpose[Placement2] . (Transpose[(M + Transpose[Q].Outer[Times, ρ, γ])/2]).
      Placement2 ;
   EXPH = MatrixExp[τ H];
   A11 = {{EXPH[[1, 1]], EXPH[[1, 2]]}, {EXPH[[2, 1]], EXPH[[2, 2]]}};
   A21 = {{EXPH[[3, 1]], EXPH[[3, 2]]}, {EXPH[[4, 1]], EXPH[[4, 2]]}};
   A = Inverse[A11] .A21;
   c = -2 β^2 (Tr[MatrixLog2[A11] + τ/2 (Transpose[M] + Outer[Times, γ, ρ].Q) ]);
   (*c=-β/2 Tr[Log[A22]+τ (Transpose[M]+Outer[Times,γ,ρ].Q)];*)
   Exp[Tr[A . Σ] + c]
  ]
```

the problem of this version is that Q is recomputed insde the procedure that will be called at evry node ($\gamma 1$, $\gamma 2$), so we optimize the procedure by externalizing the computation of Q and precomputing all the matrix algebra :

```
SuperBiHestonLaplaceTransformReduced[
  {{M11_, M12_}, {M21_, M22_}}, {{Q11_, Q12_}, {Q21_, Q22_}},
  {ρ1_, ρ2_}, {{Σ11_, Σ12_}, {Σ21_, Σ22_}}, {γ1_, γ2_}, β_, τ_] :=
 Module[{EXPH = MatrixExp[
      τ {{1/2 (-M11 - Q11 γ1 ρ1 - Q21 γ1 ρ2), 1/2 (-M12 - Q11 γ2 ρ1 - Q21 γ2 ρ2),
          1/2 (-Q11^2 - Q21^2), 1/2 (-Q11 Q12 - Q21 Q22)}, {1/2 (-M21 - Q12 γ1 ρ1 - Q22 γ1 ρ2),
          1/2 (-M22 - Q12 γ2 ρ1 - Q22 γ2 ρ2), 1/2 (-Q11 Q12 - Q21 Q22), 1/2 (-Q12^2 - Q22^2)},
         {1/2 × (-1 + γ1) γ1, (γ1 γ2)/2, 1/2 (M11 + Q11 γ1 ρ1 + Q21 γ1 ρ2),
          1/2 (M21 + Q12 γ1 ρ1 + Q22 γ1 ρ2)}, {(γ1 γ2)/2, 1/2 × (-1 + γ2) γ2,
          1/2 (M12 + Q11 γ2 ρ1 + Q21 γ2 ρ2), 1/2 (M22 + Q12 γ2 ρ1 + Q22 γ2 ρ2)}}], δ},
  δ = -EXPH[[1, 2]] × EXPH[[2, 1]] + EXPH[[1, 1]] × EXPH[[2, 2]];
  Exp[1/δ (EXPH[[2, 2]] × EXPH[[3, 1]] Σ11 - EXPH[[1, 2]] × EXPH[[4, 1]] Σ11 -
       EXPH[[2, 1]] × EXPH[[3, 1]] Σ12 + EXPH[[2, 2]] × EXPH[[3, 2]] Σ12 +
       EXPH[[1, 1]] × EXPH[[4, 1]] Σ12 - EXPH[[1, 2]] × EXPH[[4, 2]] Σ12 -
       EXPH[[2, 1]] × EXPH[[3, 2]] Σ22 + EXPH[[1, 1]] × EXPH[[4, 2]] Σ22) -
    2 β^2 (Log[δ] + τ/2 ( M11 + M22 + Q11 γ1 ρ1 + Q12 γ2 ρ1 + Q21 γ1 ρ2 + Q22 γ2 ρ2))]
 ]
```

the same procedure but where' underlying 1 is echanged with underlying 2

```
SuperBiHestonLaplaceTransformReduced2[
  {{M22_, M21_}, {M12_, M11_}}, {{Q22_, Q21_}, {Q12_, Q11_}},
  {ρ2_, ρ1_}, {{Σ22_, Σ21_}, {Σ12_, Σ11_}}, {γ1_, γ2_}, β_, τ_] :=
 Module[{EXPH = MatrixExp[
      τ {{1/2 (-M11 - Q11 γ1 ρ1 - Q21 γ1 ρ2), 1/2 (-M12 - Q11 γ2 ρ1 - Q21 γ2 ρ2),
          1/2 (-Q11^2 - Q21^2), 1/2 (-Q11 Q12 - Q21 Q22)}, {1/2 (-M21 - Q12 γ1 ρ1 - Q22 γ1 ρ2),
          1/2 (-M22 - Q12 γ2 ρ1 - Q22 γ2 ρ2), 1/2 (-Q11 Q12 - Q21 Q22), 1/2 (-Q12^2 - Q22^2)},
         {1/2 × (-1 + γ1) γ1, γ1 γ2/2, 1/2 (M11 + Q11 γ1 ρ1 + Q21 γ1 ρ2),
          1/2 (M21 + Q12 γ1 ρ1 + Q22 γ1 ρ2)}, {γ1 γ2/2, 1/2 × (-1 + γ2) γ2,
          1/2 (M12 + Q11 γ2 ρ1 + Q21 γ2 ρ2), 1/2 (M22 + Q12 γ2 ρ1 + Q22 γ2 ρ2)}}], δ},
  δ = -EXPH[[1, 2]] × EXPH[[2, 1]] + EXPH[[1, 1]] × EXPH[[2, 2]];
  Exp[1/δ (EXPH[[2, 2]] × EXPH[[3, 1]] Σ11 - EXPH[[1, 2]] × EXPH[[4, 1]] Σ11 -
       EXPH[[2, 1]] × EXPH[[3, 1]] Σ12 + EXPH[[2, 2]] × EXPH[[3, 2]] Σ12 +
       EXPH[[1, 1]] × EXPH[[4, 1]] Σ12 - EXPH[[1, 2]] × EXPH[[4, 2]] Σ12 -
       EXPH[[2, 1]] × EXPH[[3, 2]] Σ22 + EXPH[[1, 1]] × EXPH[[4, 2]] Σ22) -
    2 β^2 (Log[δ] + τ/2 ( M11 + M22 + Q11 γ1 ρ1 + Q12 γ2 ρ1 + Q21 γ1 ρ2 + Q22 γ2 ρ2))]
 ]
```

In annex , we write the 1 dimension case, and compare it with the known Heston formula. it tests and verifies the matrix exponential method for solving the riccati equation

Computation of the integrand

Naive integrand

following the preceding theory we can write a first version of the integrand

```
SuperBiHestonVanillaIntegrand[K_, τ_, M_, Σinf_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 Module[{x1 = Y[[1]], x2 = Y[[2]]}, Re[
    SuperBiHestonLaplaceTransform[M, Σinf, ρ, Σ, {-i (ω1 + i λ1), -i (ω2 - i λ2)}, β, τ]
     e^(-i x1 ( ω1+i λ1)-i x2 ( ω2-i λ2)) CompleteFourierPayOffGauche[ω1 + i λ1, ( ω2 - i λ2), K] +
     SuperBiHestonLaplaceTransform[M, Σinf, ρ, Σ,
       {-i (ω1 + i λ1), -i (ω2 + i λ2)}, β, τ] e^(-i x1 ( ω1+i λ1)-i x2 ( ω2+i λ2))
      CompleteFourierPayOffDroite[ω1 + i λ1, ( ω2 + i λ2), K]]] /; K ≠ 0
```

```
SuperBiHestonVanillaSymetrizedIntegrand[
  K_, τ_, M_, Σinf_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 (SuperBiHestonVanillaIntegrand[K, τ, M, Σinf, ρ, Σ, Y, β, λ1, λ2, ω1, ω2] +
   SuperBiHestonVanillaIntegrand[K, τ, M, Σinf, ρ, Σ, Y, β, λ1, λ2, -ω1, ω2])
```

But this version recomputes the cholesky matrix every time the SuperBiHestonLaplaceTransform function is called, and we want to include the symetrization.
Also we want to adress the K<0 case and the handling of the ATM gap (or not)

Integrand without handling of the at the money gap

```
NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
  K_, τ_, M_, Q_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 Module[{x1 = Y〚1〛, x2 = Y〚2〛, k1 = (ω1 + ⅈ λ1), Sk1 = (-ω1 + ⅈ λ1),
   k2 = (ω2 + ⅈ λ2), Sk2 = (-ω2 + ⅈ λ2), Sk1A = (-ω1 + ⅈ λ1), k1A = (ω1 - ⅈ λ1),
   k2A = (ω2 - ⅈ λ2), α, αA, Symα, SymαA, α2, αA2, Symα2, SymαA2, propagatorDroit,
   SympropagatorDroit, propagatorGauche, SympropagatorGauche, propagatorDroit2,
   SympropagatorDroit2, propagatorGauche2, SympropagatorGauche2},
  Re[If[K > 0,
    α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
    αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
    Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
    SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
    propagatorDroit =
     SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
    SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
    propagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
    SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
    α propagatorDroit CompleteFourierPayOffDroite[k1, k2, K] +
     Symα SympropagatorDroit CompleteFourierPayOffDroite[Sk1, k2, K] +
     αA propagatorGauche CompleteFourierPayOffGauche[k1, k2A, K] +
     SymαA SympropagatorGauche CompleteFourierPayOffGauche[Sk1A, k2A, K],
    If[K < 0,
     α2 = ⅇ^(-ⅈ x2 k1-ⅈ x1 k2);
     αA2 = ⅇ^(-ⅈ x2 k1-ⅈ x1 k2A);
     Symα2 = ⅇ^(-ⅈ x2 Sk1-ⅈ x1 k2);
     SymαA2 = ⅇ^(-ⅈ x2 Sk1A-ⅈ x1 k2A);
     propagatorDroit2 =
      SuperBiHestonLaplaceTransformReduced2[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
     SympropagatorDroit2 = SuperBiHestonLaplaceTransformReduced2[
       M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
     propagatorGauche2 = SuperBiHestonLaplaceTransformReduced2[
       M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
     SympropagatorGauche2 = SuperBiHestonLaplaceTransformReduced2[
       M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
     (α2 (propagatorDroit2 (CompleteFourierPayOffDroite[k1, k2, -K])) +
        Symα2 (SympropagatorDroit2 (CompleteFourierPayOffDroite[Sk1, k2, -K]))) +
      (αA2 propagatorGauche2 ( CompleteFourierPayOffGauche[k1, k2A, -K]) +
        SymαA2 SympropagatorGauche2 (CompleteFourierPayOffGauche[Sk1A, k2A, -K]))
     ,
     α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
```

```
    αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
    Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
    SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
    propagatorDroit =
     SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
    SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
    propagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
    SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
    α propagatorDroit CompleteFourierPayOffDroite[k1, k2] +
     Symα SympropagatorDroit CompleteFourierPayOffDroite[Sk1, k2] +
     αA propagatorGauche CompleteFourierPayOffGauche[k1, k2A] +
     SymαA SympropagatorGauche CompleteFourierPayOffGauche[Sk1, k2A]]]]]
```

Integrand with handling of the at the money gap

```
NewSymetrizedSuperBiHestonVanillaReducedIntegrand[
  K_, τ_, M_, Q_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 Module[{x1 = Y[[1]], x2 = Y[[2]], k1 = (ω1 + ⅈ λ1), Sk1 = (-ω1 + ⅈ λ1), k2 = (ω2 + ⅈ λ2),
   Sk2 = (-ω2 + ⅈ λ2), Sk1A = (-ω1 + ⅈ λ1), k1A = (ω1 - ⅈ λ1), k2A = (ω2 - ⅈ λ2),
   α, αA, Symα, SymαA, α2, αA2, Symα2, SymαA2, propagatorDroit,
   SympropagatorDroit, propagatorGauche, SympropagatorGauche, propagatorDroit2,
   SympropagatorDroit2, propagatorGauche2, SympropagatorGauche2},
  Re[If[K > 0,
    α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
    αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
    Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
    SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
    propagatorDroit =
     SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
    SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
    propagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
    SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
      M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
    α propagatorDroit CompleteFourierPayOffDroite[k1, k2, K] +
     Symα SympropagatorDroit CompleteFourierPayOffDroite[Sk1, k2, K] +
     αA propagatorGauche CompleteFourierPayOffGauche[k1, k2A, K] +
     SymαA SympropagatorGauche CompleteFourierPayOffGauche[Sk1A, k2A, K],
    If[K < 0,
     α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
     αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
     Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
     SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
     propagatorDroit =
      SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
```

```
SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
propagatorGauche = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
α2 = ⅇ^(-ⅈ x2 k1-ⅈ x1 k2);
αA2 = ⅇ^(-ⅈ x2 k1-ⅈ x1 k2A);
Symα2 = ⅇ^(-ⅈ x2 Sk1-ⅈ x1 k2);
SymαA2 = ⅇ^(-ⅈ x2 Sk1A-ⅈ x1 k2A);
propagatorDroit2 =
  SuperBiHestonLaplaceTransformReduced2[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
SympropagatorDroit2 = SuperBiHestonLaplaceTransformReduced2[
   M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
propagatorGauche2 = SuperBiHestonLaplaceTransformReduced2[
   M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
SympropagatorGauche2 = SuperBiHestonLaplaceTransformReduced2[
   M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
(α2 (propagatorDroit2 (CompleteFourierPayOffDroite[k1, k2, -K])) +
    Symα2 (SympropagatorDroit2 (CompleteFourierPayOffDroite[Sk1, k2, -K]))) +
  (αA2 propagatorGauche2 ( CompleteFourierPayOffGauche[k1, k2A, -K]) +
    SymαA2 SympropagatorGauche2 (CompleteFourierPayOffGauche[Sk1A, k2A, -K])) +
  (α propagatorDroit - α2 propagatorDroit2) CompleteFourierPayOffDroite[k1, k2] +
  (Symα SympropagatorDroit - Symα2 SympropagatorDroit2)
   CompleteFourierPayOffDroite[Sk1, k2] +
  (αA propagatorGauche - αA2 propagatorGauche2)
   CompleteFourierPayOffGauche[k1, k2A] +
  (SymαA SympropagatorGauche - SymαA2 SympropagatorGauche2)
   CompleteFourierPayOffGauche[Sk1, k2A]


,
α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
propagatorDroit =
  SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
propagatorGauche = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
   M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
α propagatorDroit CompleteFourierPayOffDroite[k1, k2] +
  Symα SympropagatorDroit CompleteFourierPayOffDroite[Sk1, k2] +
  αA propagatorGauche CompleteFourierPayOffGauche[k1, k2A] +
  SymαA SympropagatorGauche CompleteFourierPayOffGauche[Sk1, k2A]]]]]
```

Of course all these functions computes the same things

```
Timing[Module[{S1 = 0.05, S2 = 0.045, K = 0.00001, M1 = -0.01, M2 = -0.02,
   θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3,
   ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2,
   scope1, scope2, nb = 40, printflag = 0, ω1 = 1, ω2 = 1, Σinf, M, Σ, Q},
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  scope1 = 2/(Sqrt[(Σ1+Σ2+θ1+θ2)/4] τ);
  scope2 = 3/(Sqrt[(Σ1+Σ2+θ1+θ2)/4] τ);
  Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)]/β;
  {NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ, M, Q, {ρ1, ρ2},
    {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}}, {Log[S1], Log[S2]}, β, λ1, λ2, ω1, ω2],
   SuperBiHestonVanillaSymetrizedIntegrand[K, τ, M, Σinf,
    {ρ1, ρ2}, Σ, {Log[S1], Log[S2]}, β, λ1, λ2, ω1, ω2]
  }]]
{0.016, {-0.195351, -0.195351}}
```

Final Formula for the computation of the vanilla option

Option brute

with handling of the at the money gap

```
NewSuperBiHestonVanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_, λ1_, λ2_,
  Scope1_, Scope2_, Nb_, printflag_] := NewSuperBiHestonVanillaAux[K,
   τ, M, Σinf, ρ, Σ, S, β, λ1, λ2, Scope1, Scope2, Nb, printflag] /; K ≥ 0
```

```
NewSuperBiHestonVanilla[K_, τ_, M_, Σinf_, ρ_,
  Σ_, S_, β_, λ1_, λ2_, Scope1_, Scope2_, Nb_, printflag_] :=
 NewSuperBiHestonVanillaAux[K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2, Scope1,
    Scope2, Nb, printflag] + S[[1]] - S[[2]] - K /; K < 0
```

```
NewSuperBiHestonVanillaAux[K_, τ_, M_, Σinf_, ρ_,
  Σ_, S_, β_, λ1_, λ2_, Scope1_, Scope2_, Nb_, printflag_] :=
 2/(2 π)^2 Module[{res, Y = {Log[S[[1]]], Log[S[[2]]]}, Q},
   Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
   If[printflag == 1, Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, ρ,
      Σ, S, β, λ1, λ2}, " {Nb,Scope1,Scope2}=", {Nb, Scope1, Scope2}]];
   res = CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
        K, τ, M, Q, ρ, Σ, Y, β, λ1, λ2, #1, #2]) &,
     RiemanCoeffs[Nb, 0, Scope1], RiemanCoeffs[Nb / 2, 0, Scope2]];
   If[printflag == 1, Print["res-aux=", res]];
   res]
```

without handling of the at the money gap

```
NewSuperBiHestonVanilla2[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_, λ1_, λ2_,
  Scope1_, Scope2_, Nb_, printflag_] := NewSuperBiHestonVanillaAux2[K,
   τ, M, Σinf, ρ, Σ, S, β, λ1, λ2, Scope1, Scope2, Nb, printflag] /; K ≥ 0
```

```
NewSuperBiHestonVanilla2[K_, τ_, M_, Σinf_, ρ_,
  Σ_, S_, β_, λ1_, λ2_, Scope1_, Scope2_, Nb_, printflag_] :=
 NewSuperBiHestonVanillaAux2[K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2,
    Scope1, Scope2, Nb, printflag] - K /; K < 0
```

```
NewSuperBiHestonVanillaAux2[K_, τ_, M_, Σinf_, ρ_,
  Σ_, S_, β_, λ1_, λ2_, Scope1_, Scope2_, Nb_, printflag_] :=
 2/(2 π)^2 Module[{res, Y = {Log[S[[1]]], Log[S[[2]]]}, Q},
   Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
   If[printflag == 1, Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, ρ,
      Σ, S, β, λ1, λ2}, " {Nb,Scope1,Scope2}=", {Nb, Scope1, Scope2}]];
   res = CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand[
        K, τ, M, Q, ρ, Σ, Y, β, λ1, λ2, #1, #2]) &,
     RiemanCoeffs[Nb, 0, Scope1], RiemanCoeffs[Nb / 2, 0, Scope2]];
   If[printflag == 1, Print["res-aux=", res]];
   res]
```

Option Interfacée Heston (avec traitement du gap a la monnaie)

Several integration shema are provided

```
NewSuperBiHestonVanilla[{ν1_, ρs1_, χ1_, Σinf1_}, {ν2_, ρs2_, χ2_, Σinf2_},
  {ρ1_, ρ2_}, {ρ12_, ρinf12_} , {Σ1_, Σ2_}, {S1_, S2_}, β_, K_,
  τ_, λ1_, λ2_, z1max_, z2max_, Nb_, Integflag_, printflag_] :=
 NewSuperBiHestonVanillaAux[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
   {ρ1, ρ2}, {ρ12, ρinf12} , {Σ1, Σ2}, {S1, S2}, β, K, τ,
   λ1, λ2, z1max, z2max, Nb, Integflag, printflag]  /; K ≥ 0
```

```
NewSuperBiHestonVanilla[{ν1_, ρs1_, χ1_, Σinf1_}, {ν2_, ρs2_, χ2_, Σinf2_},
  {ρ1_, ρ2_}, {ρ12_, ρinf12_} , {Σ1_, Σ2_}, {S1_, S2_}, β_, K_,
  τ_, λ1_, λ2_, z1max_, z2max_, Nb_, Integflag_, printflag_] :=
 NewSuperBiHestonVanillaAux[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
    {ρ1, ρ2}, {ρ12, ρinf12} , {Σ1, Σ2}, {S1, S2}, β, K, τ, λ1, λ2,
    z1max, z2max, Nb, Integflag, printflag] + S1 - S2 - K  /; K < 0
```

```
NewSuperBiHestonVanillaAux[{ν1_, ρs1_, χ1_, Σinf1_},
  {ν2_, ρs2_, χ2_, Σinf2_}, {ρ1_, ρ2_}, {ρ12_, ρinf12_} , {Σ1_, Σ2_}, {S1_, S2_},
  β_, K_, τ_, λ1_, λ2_, z1max_, z2max_, Nb_, Integflag_, printflag_] :=
 Module[{M, Σinf, Σ, Q, Σ12, Scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ], Scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ],
   result, coeffs1, coeffs2, y0, Y1 = Log[S1], Y2 = Log[S2]},
  Σ12 = Sqrt[Σ1 Σ2] ρ12;
  Q = DetermineQ[ρ1, ρ2, ν1, ν2, ρs1, ρs2, printflag];
  M = DetermineM[Q, β, Σinf1, Σinf2, χ1, χ2, Σ12, printflag];
  Σinf = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Σ12}, {Σ12, Σ2}};
  If[printflag == 1,
   Print["M=", M // MatrixForm, "  Q=", Q // MatrixForm, "  Σ=", Σ // MatrixForm,
     "  Σinf=", Σinf // MatrixForm, " Scope1=", Scope1, " Scope2=", Scope2];];
  If[printflag == 1, Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, {ρ1, ρ2},
     Σ, {S1, S2}, β, λ1, λ2}, " {Nb,Scope1,Scope2}=", {Nb, Scope1, Scope2}]];
  If[Integflag == 0, If[printflag == 1, Print["integflag=", Integflag]];
   result = 2/(2 π)^2 Module[{},
      CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
           K, τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &,
       RiemanCoeffs[Nb / 2, 0, Scope1], RiemanCoeffs[Nb / 2, 0, Scope2]]]];
  If[Integflag == 1, If[printflag == 1, Print["integflag=", Integflag]];
   result = 2/(2 π)^2 Module[{}, CoeffBasedIntegrate[
       (NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ, M, Q, {ρ1, ρ2},
          Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, ZeroTangeanteRiemanCoeffs[
        Nb / 2, 0, Scope1], ZeroTangeanteRiemanCoeffs[Nb / 2, 0, Scope2]]]];

  If[Integflag == 2, If[printflag == 1, Print["integflag=", Integflag]];
```

```
    result = 2/(2 π)^2 Module[{},
       CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
           K, τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &,
        ReenforcedRiemanCoeffs[Floor[Nb / 2], 0, Scope1 / 3, Scope1],
        ReenforcedRiemanCoeffs[Floor[Nb / 2], 0, Scope2 / 3, Scope2]]]];
  If[Integflag == 3, If[printflag == 1, Print["integflag=", Integflag]];
    result = 2/(2 π)^2 Module[{},
       CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
           K, τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &,
        ZeroTangeanteReenforcedRiemanCoeffs[Floor[Nb / 2], 0, Scope1 / 3, Scope1],
        ZeroTangeanteReenforcedRiemanCoeffs[Floor[Nb / 2], 0, Scope2 / 3, Scope2]]]];
  If[Integflag == 10, If[printflag == 1, Print["integflag=", Integflag]];
    coeffs1 = SpecialExpCoeffs[
      {0, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, 0}]},
      {Scope1 / 5, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {Scope1 / 5, 0}]},
      {Scope1 / 2, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {Scope1 / 2, 0}]},
      0, Scope1, Nb];
    coeffs2 = SpecialExpCoeffs[
      {0, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, 0}]},
      {Scope2 / 5, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, Scope2 / 5}]},
      {Scope2 / 2, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, Scope2 / 2}]},
      0, Scope2, Nb];
    result = 2/(2 π)^2 Module[{},
       CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ,
           M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, coeffs1, coeffs2]]];
  If[Integflag == 11, If[printflag == 1, Print["integflag=", Integflag]];
    y0 = Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
         K, τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, 0}];
    coeffs1 = SpecialExpCoeffs2[
      y0,
      {Scope1 / 2, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ, M,
           Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {Scope1 / 2, 0}]},
      0, Nb];
    coeffs2 = SpecialExpCoeffs2[
      {0, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, 0}]},
      {Scope2 / 5, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
           τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, Scope2 / 5}]},
```

```
    {Scope2 / 2, Apply[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
          τ, M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, {0, Scope2 / 2}]},
    0, Scope2, Nb];
  result = 2/(2 π)^2 Module[{},
      CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ,
          M, Q, {ρ1, ρ2}, Σ, {Y1, Y2}, β, λ1, λ2, #1, #2]) &, coeffs1, coeffs2]];
 result
]
```

Example of execution

```
Timing[
 Module[{S1 = 0.05, S2 = 0.05, K = 0.0000001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041,
   ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05,
   β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 40, printflag = 0, M, Σinf, Σ},
  scope1 = 2/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2             );
  Σinf = ( θ1                    Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf     θ2                );
  Σ = ( Σ1                Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs    Σ2             );
  {NewSuperBiHestonVanilla[K, τ, M, Σinf,
    {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2, scope1, scope2, nb, printflag],
   NewSuperBiHest∂□ □onVanilla[0, τ, M, Σinf, {ρ1, ρ2}, Σ,
     {S1, S2}, β, λ1, λ2, scope1, scope2, nb, printflag],
   NewSuperBiHestonVanilla[-K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
    β, λ1, λ2, scope1, scope2, nb, printflag]
  }]]
{12.078, {0.00859547, 0.00861146, 0.00860579}}
```

```
Timing[
 Module[{S1 = 0.05, S2 = 0.045, K = 0.0000001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041,
   ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05,
   β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 40, printflag = 0, M, Σinf, Σ},
  scope1 = 2/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  {NewSuperBiHestonVanilla[K, τ, M, Σinf,
    {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2, scope1, scope2, nb, printflag],
   NewSuperBiHestonVanilla[0, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
    β, λ1, λ2, scope1, scope2, nb, printflag],
   NewSuperBiHestonVanilla[-K, τ, M, Σinf, {ρ1, ρ2}, Σ,
    {S1, S2}, β, λ1, λ2, scope1, scope2, nb, printflag]
  }]]
{12.047, {0.0108955, 0.0109097, 0.010946}}
```

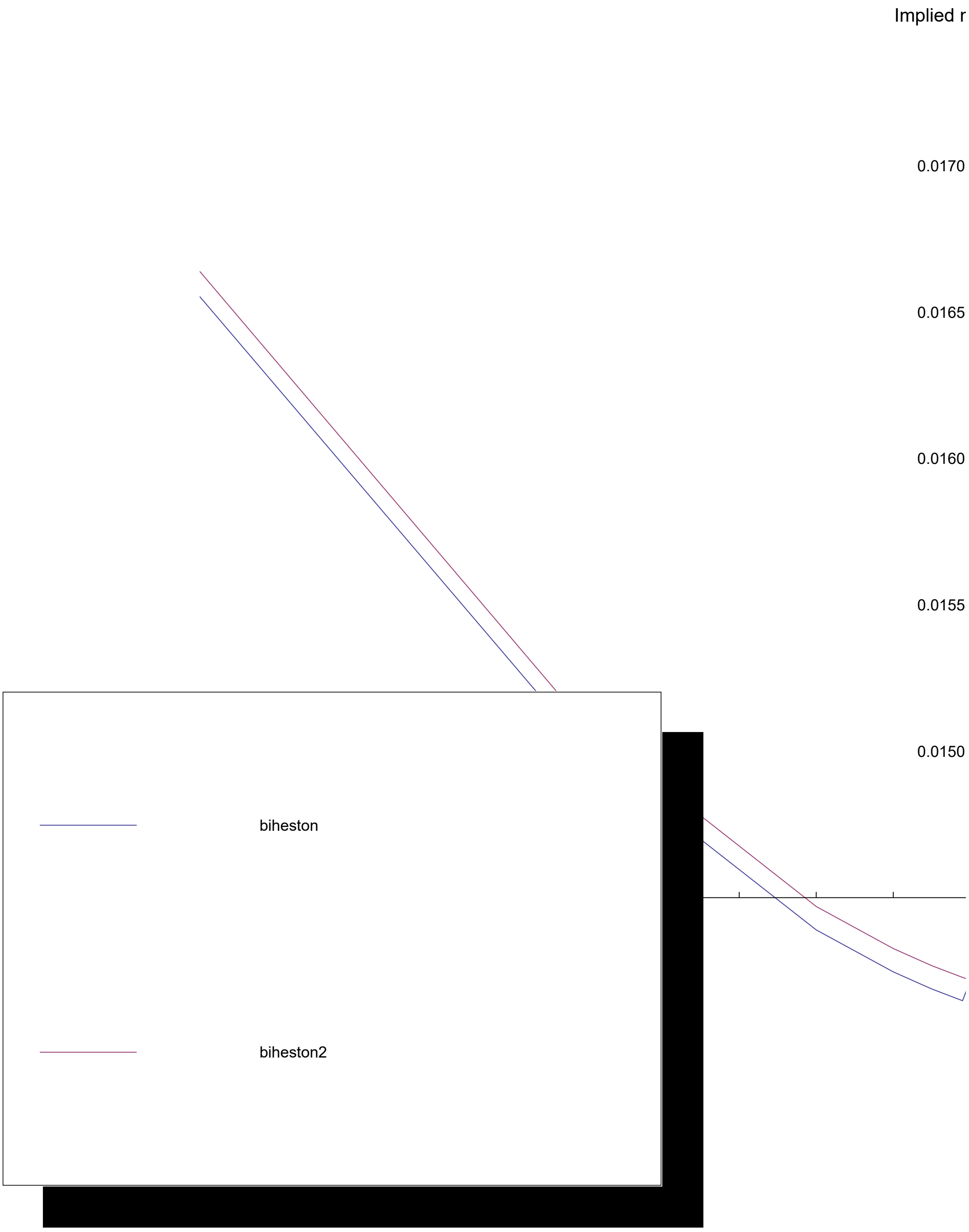
In the notebook, scope1 and scope2 are displayed as fractions, $\frac{2}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}}$ and $\frac{3}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}}$, and M, Σinf and Σ are displayed as 2×2 matrices:

$$M = \begin{pmatrix} M1 & \rho m1 \sqrt{M1\,M2} \\ \rho m2 \sqrt{M1\,M2} & M2 \end{pmatrix};\quad \Sigma inf = \begin{pmatrix} \theta1 & \sqrt{\theta1\,\theta2}\,\rho sinf \\ \sqrt{\theta1\,\theta2}\,\rho sinf & \theta2 \end{pmatrix};\quad \Sigma = \begin{pmatrix} \Sigma1 & \sqrt{\Sigma1\,\Sigma2}\,\rho s \\ \sqrt{\Sigma1\,\Sigma2}\,\rho s & \Sigma2 \end{pmatrix};$$

Handling of the Gap at the money

```
Timing[
 Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15, ρs = 0.8,
   ρsinf = 0.8, ρm1 = 0., ρm2 = 0.5, ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.04, Σ2 = 0.04, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, strikes = {-0.01, -0.005, -0.002, -0.001, -0.0005,
     -0.0001, 0, 0.0001, 0.0005, 0.001, 0.002, 0.005, 0.01}, scope1, scope2,
   nb = 40, inter, Lcoefs, vol1, vol2, v1, v2, printflag = 0, Σinf, M, Σ},
  M = ( M1              ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2] M2              );
  Σinf = ( θ1                 Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf  θ2                );
  Σ = ( Σ1              Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs  Σ2             );
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  smile = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
       NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
        β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  smile2 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
       NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
        β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  Plot[{inter[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "biheston2"}]
 ]]
```

{58.282,

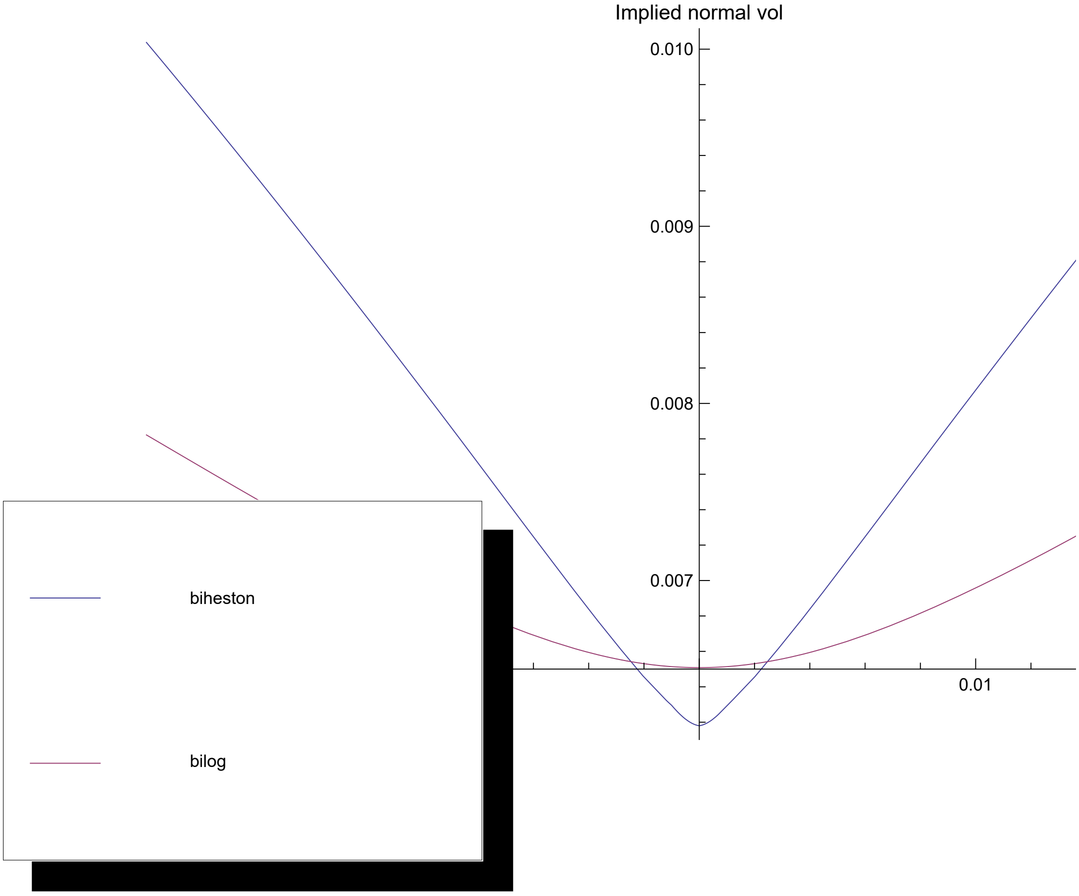


Comparaison with a lognormal model and spike at the money

Le smile presente un coin a la monaie

```
Timing[Module[{ν1 = 0.1, ν2 = 0.1, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8, ρinf12 = 0.8, β, integflag = 0,
   τ = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 60, flag = 1, det,
   Lcoefs = LegendreCoeffs[40], vol1, vol2, spdopt, strikes, βmul = 0.9},
  strikes = {-0.02, -0.015, -0.01, -0.0075, -0.005, -0.003,
    -0.001, 0, 0.001, 0.003, 0.005, 0.0075, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2];
  Print["β=", β];
  β *= βmul;
  z1max = 2; z2max = 4;
  smile1 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
       {ρ1, ρ2}, {ρ12, ρinf12}, {Σ1, Σ2}, {S1, S2}, β, strikes[[i]], τ, λ1,
       λ2, z1max, z2max, Nb, integflag, 0]]}, {i, 1, Length[strikes]}];
  inter1 = Interpolation[smile1];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, Σinf1, ρs1, χ1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, Σinf2, ρs2, χ2, ν2, Lcoefs];

  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
       S1, S2, vol1, vol2, ρ12, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter1[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

β=9.

{80.172, 

There is a discontinuity in the smile at the money that can be traced back a discontinuity of the derivative of the option price with respect to K

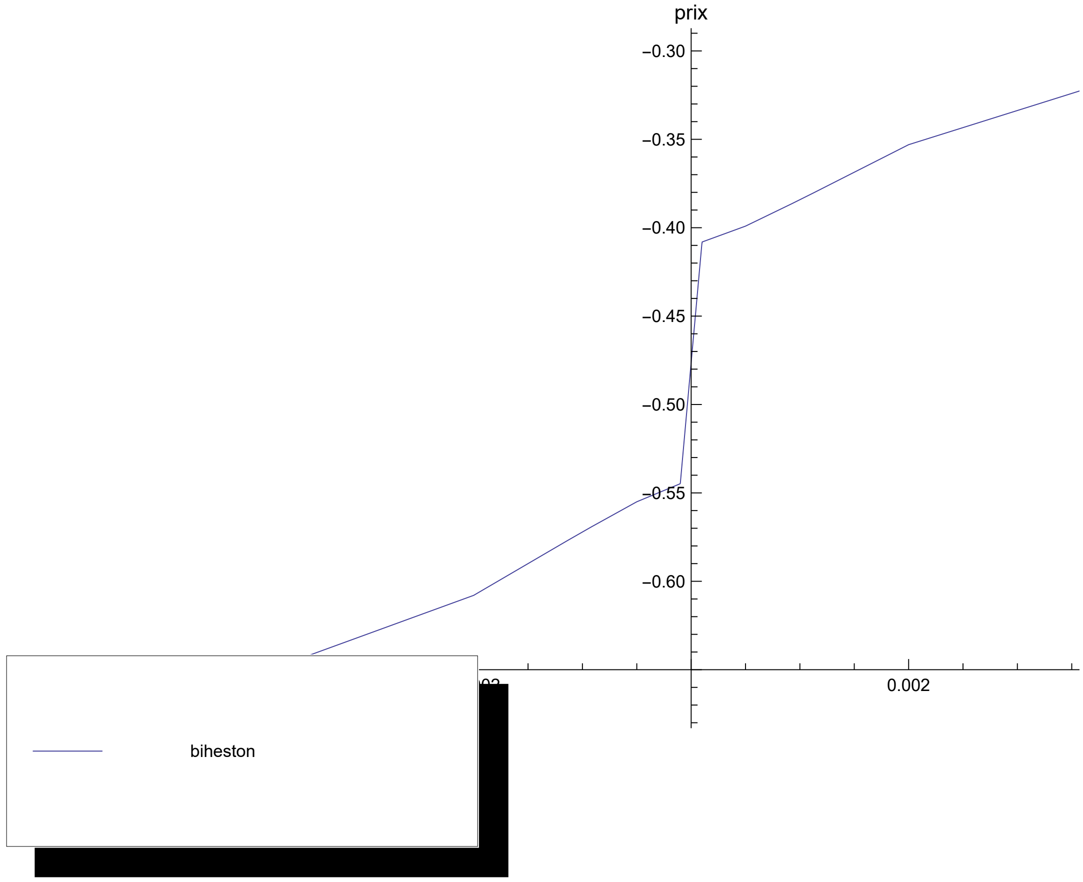
```
Timing[
 Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15, ρs = 0.8,
   ρsinf = 0.8, ρm1 = 0., ρm2 = 0.5, ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.04, Σ2 = 0.04, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, strikes = {-0.01, -0.005, -0.002, -0.001, -0.0005,
     -0.0001, 0, 0.0001, 0.0005, 0.001, 0.002, 0.005, 0.01}, scope1, scope2,
   nb = 40, inter, Lcoefs, vol1, vol2, v1, v2, printflag = 0, Σinf, M, Σ},
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  scope1 = 2/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  smile =
   Table[{strikes[[i]], NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ,
       {S1, S2}, β, λ1, λ2, scope1, scope2, nb, printflag]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  shift = 0.0001;
  smile2 = Table[
     {strikes[[i]], (inter[strikes[[i]] + shift] - inter[strikes[[i]] - shift]) / (2 shift)},
     {i, 2, Length[strikes] - 1}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  Plot[{inter2[x]},
   {x, strikes[[2]], strikes[[Length[strikes] - 1]]}, PlotLabel → "prix",
   PlotLegend → {"biheston", "biheston2"}]
 ]]
```

Matrices and fractions as typeset:

$$M = \begin{pmatrix} M1 & \rho m1 \sqrt{M1\,M2} \\ \rho m2 \sqrt{M1\,M2} & M2 \end{pmatrix};\quad \Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\,\theta 2}\ \rho sinf \\ \sqrt{\theta 1\,\theta 2}\ \rho sinf & \theta 2 \end{pmatrix};\quad \Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\,\Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\,\Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

$$scope1 = \frac{2}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};\ scope2 = \frac{3}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};$$

{76.109,

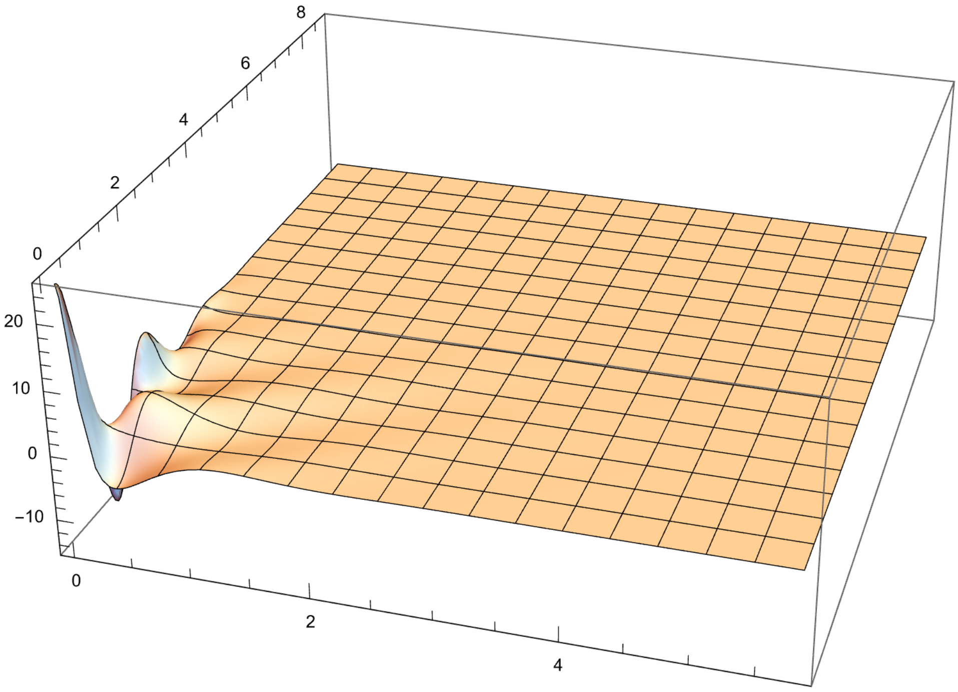


Smart Integration

$$
\begin{aligned}
&\texttt{Timing}\Big[\texttt{Module}\Big[\{\texttt{S1} = 0.04,\ \texttt{S2} = 0.04,\ \texttt{M1} = -0.075,\ \texttt{M2} = -0.075,\ \theta1 = 0.15,\ \theta2 = 0.15,\\
&\quad \rho\texttt{s} = 0.8,\ \rho\texttt{sinf} = 0.8,\ \rho\texttt{m1} = 0.,\ \rho\texttt{m2} = 0.,\ \rho1 = -0.15,\ \rho2 = -0.15,\ \Sigma1 = 0.04,\\
&\quad \Sigma2 = 0.04,\ \beta = 5,\ \tau = 5,\ \lambda1 = 1.1,\ \lambda2 = 1.2,\ \texttt{scope1},\ \texttt{scope2},\ \texttt{nb} = 60,\ \texttt{inter},\ \texttt{Lcoefs},\\
&\quad \texttt{vol1},\ \texttt{vol2},\ \nu1,\ \nu2,\ \texttt{printflag} = 0,\ \Sigma\texttt{inf},\ \texttt{M},\ \texttt{Q},\ \Sigma,\ \rho,\ \texttt{Y},\ \texttt{zmax1},\ \texttt{zmax2},\ \texttt{K} = 0.001\},\\
&\texttt{M} = \begin{pmatrix} \texttt{M1} & \rho\texttt{m1}\sqrt{\texttt{M1 M2}} \\ \rho\texttt{m2}\sqrt{\texttt{M1 M2}} & \texttt{M2} \end{pmatrix};\\
&\Sigma\texttt{inf} = \begin{pmatrix} \theta1 & \sqrt{\theta1\,\theta2}\ \rho\texttt{sinf} \\ \sqrt{\theta1\,\theta2}\ \rho\texttt{sinf} & \theta2 \end{pmatrix};\\
&\Sigma = \begin{pmatrix} \Sigma1 & \sqrt{\Sigma1\,\Sigma2}\ \rho\texttt{s} \\ \sqrt{\Sigma1\,\Sigma2}\ \rho\texttt{s} & \Sigma2 \end{pmatrix};\\
&\rho = \{\rho1,\ \rho2\};\\
&\texttt{Q} = \texttt{CholeskyDecomposition}\left[-\left(\frac{\texttt{M}.\Sigma\texttt{inf} + \Sigma\texttt{inf}.\texttt{Transpose[M]}}{2}\right)\right] \Big/ \beta;\\
&\texttt{Y} = \{\texttt{Log[S1]},\ \texttt{Log[S2]}\};\\
&\texttt{scope1} = \frac{4}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\ \tau}};\ \texttt{scope2} = \frac{6}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\ \tau}};\\
&\texttt{Plot3D[NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,}\\
&\quad \tau,\ \texttt{M},\ \texttt{Q},\ \rho,\ \Sigma,\ \texttt{Y},\ \beta,\ \lambda1,\ \lambda2,\ \omega1,\ \omega2],\ \{\omega1,\ 0,\ \texttt{scope1}\},\ \{\omega2,\ 0\ ,\ \texttt{scope2}\},\\
&\quad \texttt{PlotPoints} \to \texttt{nb},\ \texttt{PlotRange} \to \texttt{All},\ \texttt{PlotLabel} \to \texttt{"K="} <> \texttt{ToString[K]]}\Big]\Big]
\end{aligned}
$$

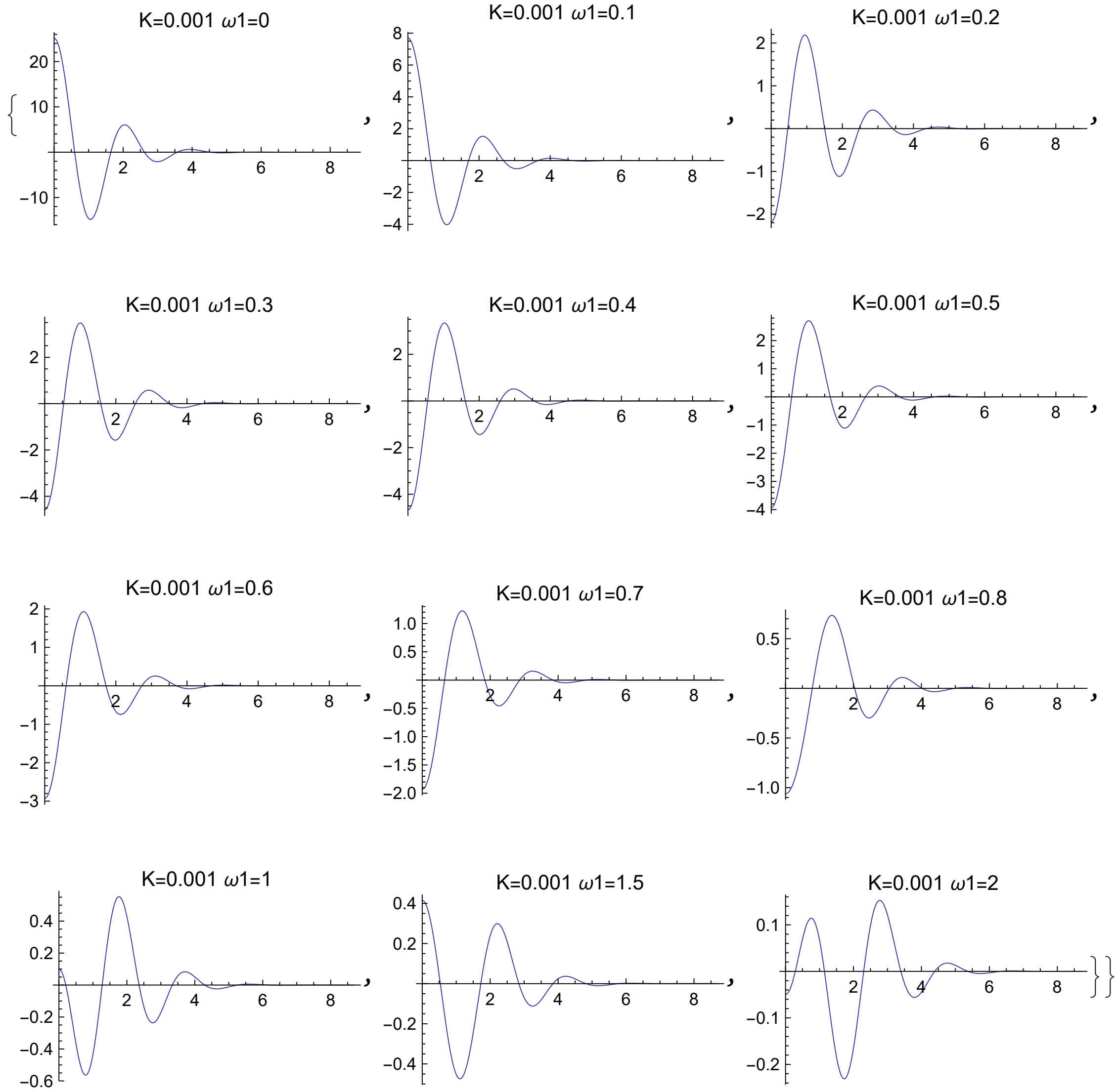
K=0.001

{265.937,



```
Timing[Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0.8, ρsinf = 0.8, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.04,
   Σ2 = 0.04, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs,
   vol1, vol2, ν1, ν2, printflag = 0, Σinf, M, Q, Σ, ρ, Y, zmax1, zmax2, K = 0.001, ω1},
  M = ( M1              ρm1 √(M1 M2)
        ρm2 √(M1 M2)    M2           );
  Σinf = ( θ1                 √(θ1 θ2) ρsinf
           √(θ1 θ2) ρsinf     θ2             );
  Σ = ( Σ1              √(Σ1 Σ2) ρs
        √(Σ1 Σ2) ρs     Σ2          );
  ρ = {ρ1, ρ2};
  Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
  Y = {Log[S1], Log[S2]};
  scope1 = 4/√((Σ1+Σ2+θ1+θ2)/4 τ); scope2 = 6/√((Σ1+Σ2+θ1+θ2)/4 τ);
  Table[Plot[NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
      τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, ω2], {ω2, 0 , scope2}, PlotPoints → nb,
     PlotRange → All, PlotLabel → "K=" <> ToString[K] <> "  ω1=" <> ToString[ω1]],
   {ω1, {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1, 1.5, 2}}]]]
```

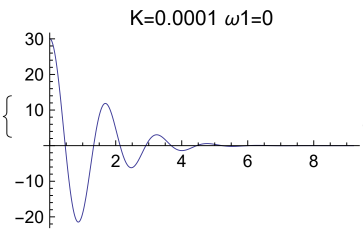
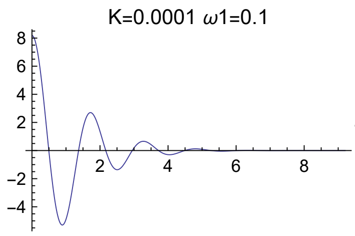
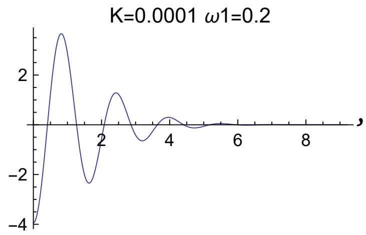
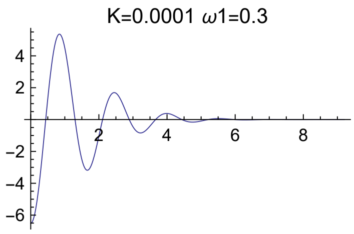
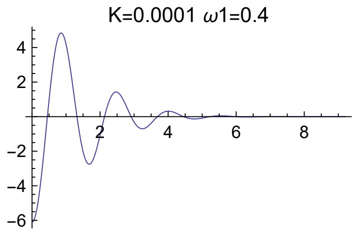
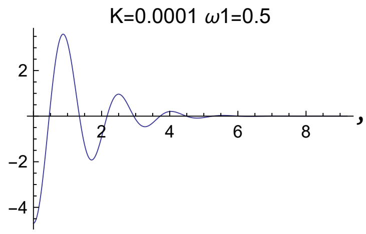
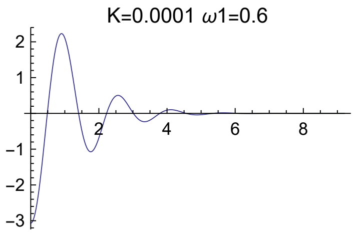
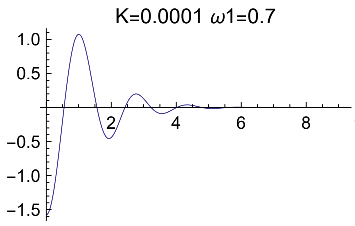
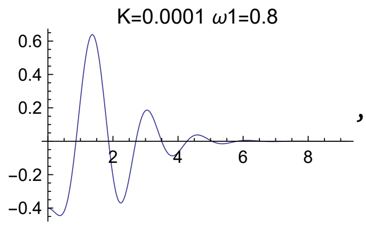
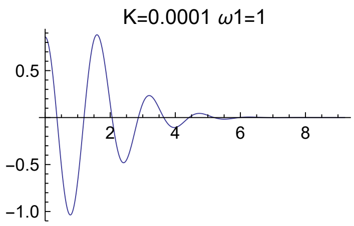
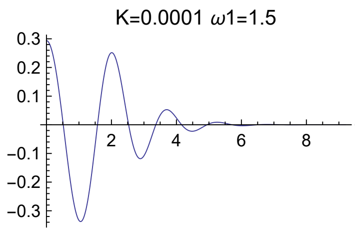
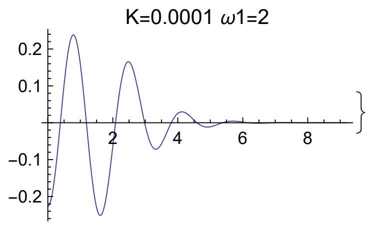
{186.765,


K=0.001 ω1=0
K=0.001 ω1=0.1
K=0.001 ω1=0.2
K=0.001 ω1=0.3
K=0.001 ω1=0.4
K=0.001 ω1=0.5
K=0.001 ω1=0.6
K=0.001 ω1=0.7
K=0.001 ω1=0.8
K=0.001 ω1=1
K=0.001 ω1=1.5
K=0.001 ω1=2

}}

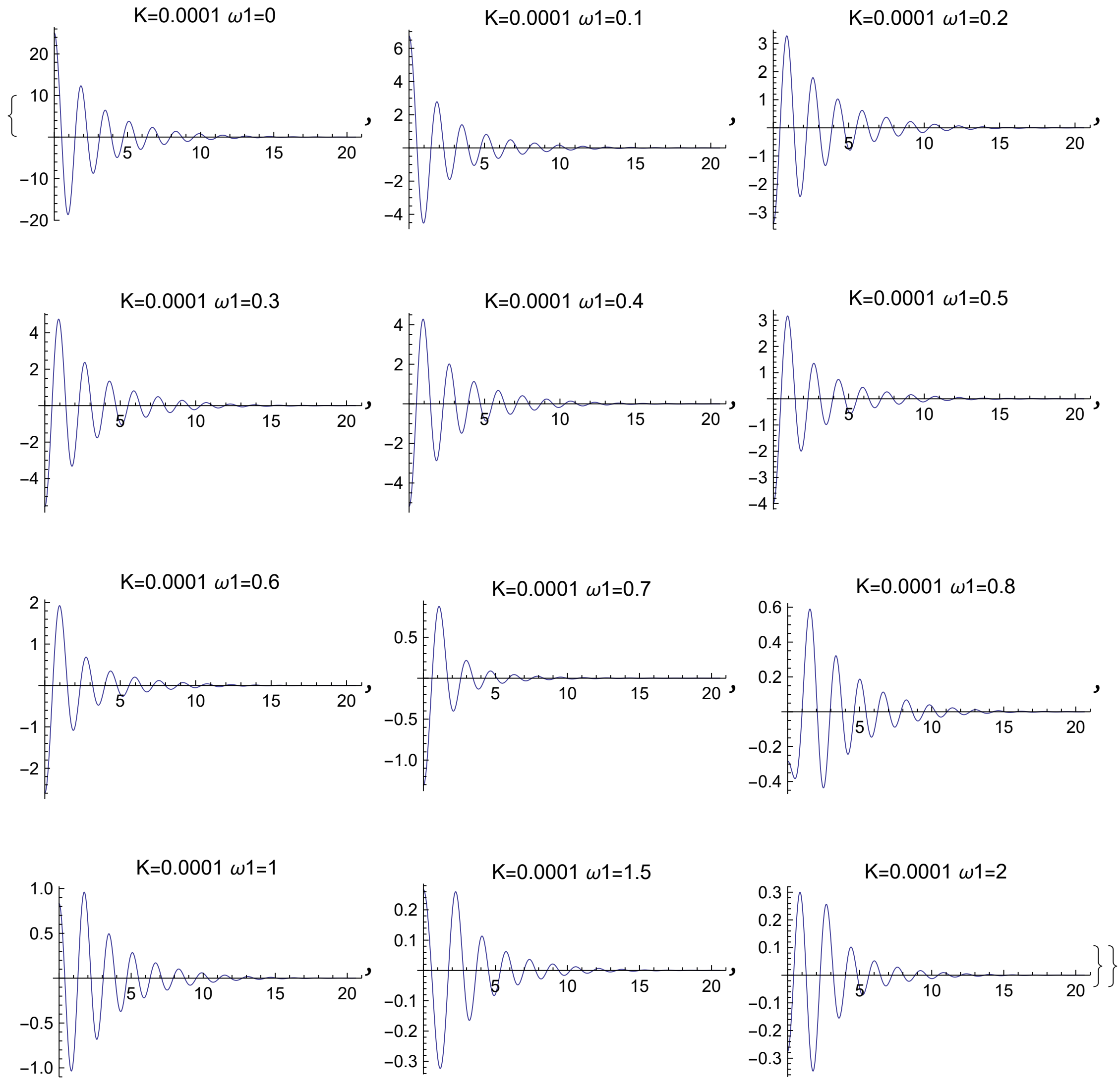
```
Timing[Module[{S1 = 0.02, S2 = 0.02, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0.3, ρsinf = 0.3, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.02, Σ2 = 0.02,
   β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs, vol1,
   vol2, ν1, ν2, printflag = 0, Σinf, M, Q, Σ, ρ, Y, zmax1, zmax2, K = 0.0001, ω1},
  M = ( M1              ρm1 √(M1 M2)
        ρm2 √(M1 M2)    M2           );
  Σinf = ( θ1                  √(θ1 θ2) ρsinf
           √(θ1 θ2) ρsinf      θ2             );
  Σ = ( Σ1              √(Σ1 Σ2) ρs
        √(Σ1 Σ2) ρs     Σ2          );
  ρ = {ρ1, ρ2};
  Q = CholeskyDecomposition[- ((M.Σinf + Σinf. Transpose[M]) / 2)] / β;
  Y = {Log[S1], Log[S2]};
  scope1 = 4 / (√((Σ1 + Σ2 + θ1 + θ2) / 4 τ));
  scope2 = 6 / (√((Σ1 + Σ2 + θ1 + θ2) / 4 τ));
  Table[Plot[NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
      τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, ω2], {ω2, 0 , scope2}, PlotPoints → nb,
     PlotRange → All, PlotLabel → "K=" <> ToString[K] <> "  ω1=" <> ToString[ω1]],
   {ω1, {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1, 1.5, 2}}]]]
```

{203.032,

{
K=0.0001 ω1=0
,

K=0.0001 ω1=0.1
,

K=0.0001 ω1=0.2
,

K=0.0001 ω1=0.3
,

K=0.0001 ω1=0.4
,

K=0.0001 ω1=0.5
,

K=0.0001 ω1=0.6
,

K=0.0001 ω1=0.7
,

K=0.0001 ω1=0.8
,

K=0.0001 ω1=1
,

K=0.0001 ω1=1.5
,

K=0.0001 ω1=2
}}

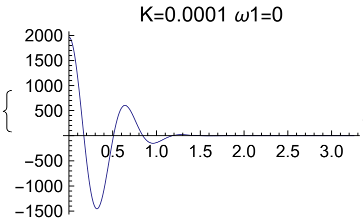
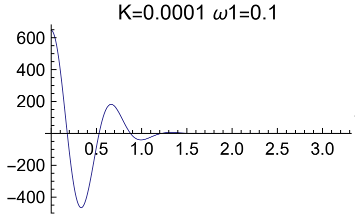
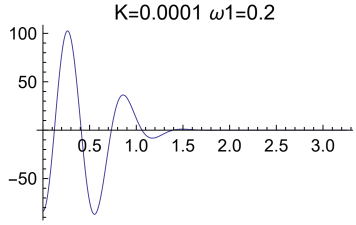
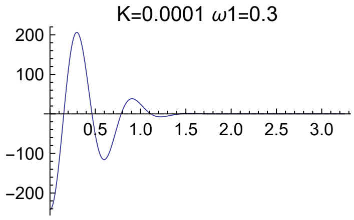
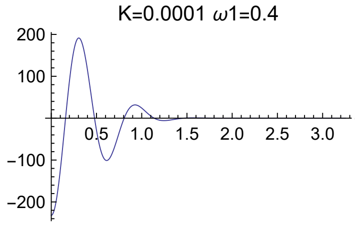
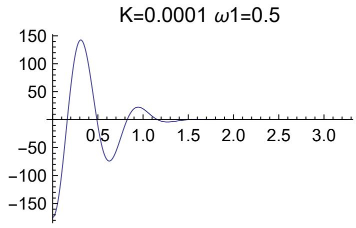
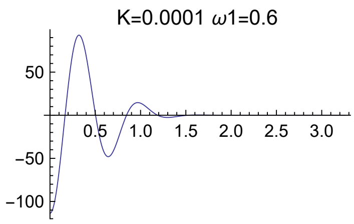
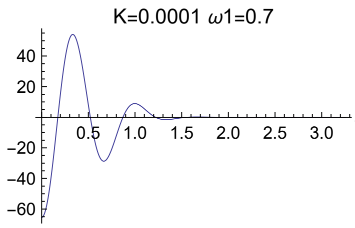
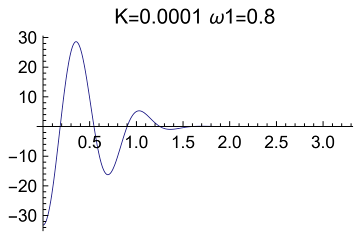
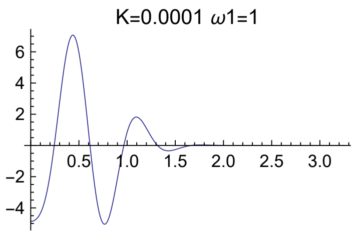
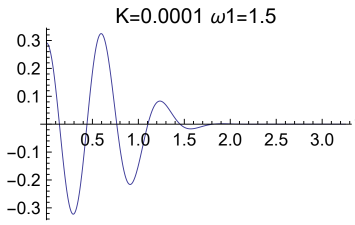
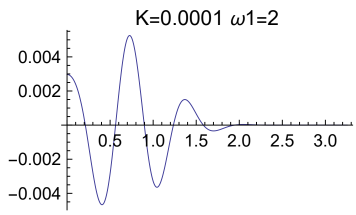
```
Timing[Module[{S1 = 0.02, S2 = 0.02, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0.3, ρsinf = 0.3, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.02, Σ2 = 0.02,
   β = 10, τ = 1, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs, vol1,
   vol2, ν1, ν2, printflag = 0, Σinf, M, Q, Σ, ρ, Y, zmax1, zmax2, K = 0.0001, ω1},
  M = ( M1              ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]  M2             );
  Σinf = ( θ1                   Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf    θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  ρ = {ρ1, ρ2};
  Q = CholeskyDecomposition[- ((M.Σinf + Σinf. Transpose[M]) / 2)] / β;
  Y = {Log[S1], Log[S2]};
  scope1 = 4 / (Sqrt[((Σ1 + Σ2 + θ1 + θ2) / 4 τ)]);
  scope2 = 6 / (Sqrt[((Σ1 + Σ2 + θ1 + θ2) / 4 τ)]);
  Table[Plot[NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
      τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, ω2], {ω2, 0 , scope2}, PlotPoints → nb,
     PlotRange → All, PlotLabel → "K=" <> ToString[K] <> "  ω1=" <> ToString[ω1]],
   {ω1, {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1, 1.5, 2}}]]]
```

{284.703,


K=0.0001 ω1=0
K=0.0001 ω1=0.1
K=0.0001 ω1=0.2
K=0.0001 ω1=0.3
K=0.0001 ω1=0.4
K=0.0001 ω1=0.5
K=0.0001 ω1=0.6
K=0.0001 ω1=0.7
K=0.0001 ω1=0.8
K=0.0001 ω1=1
K=0.0001 ω1=1.5
K=0.0001 ω1=2

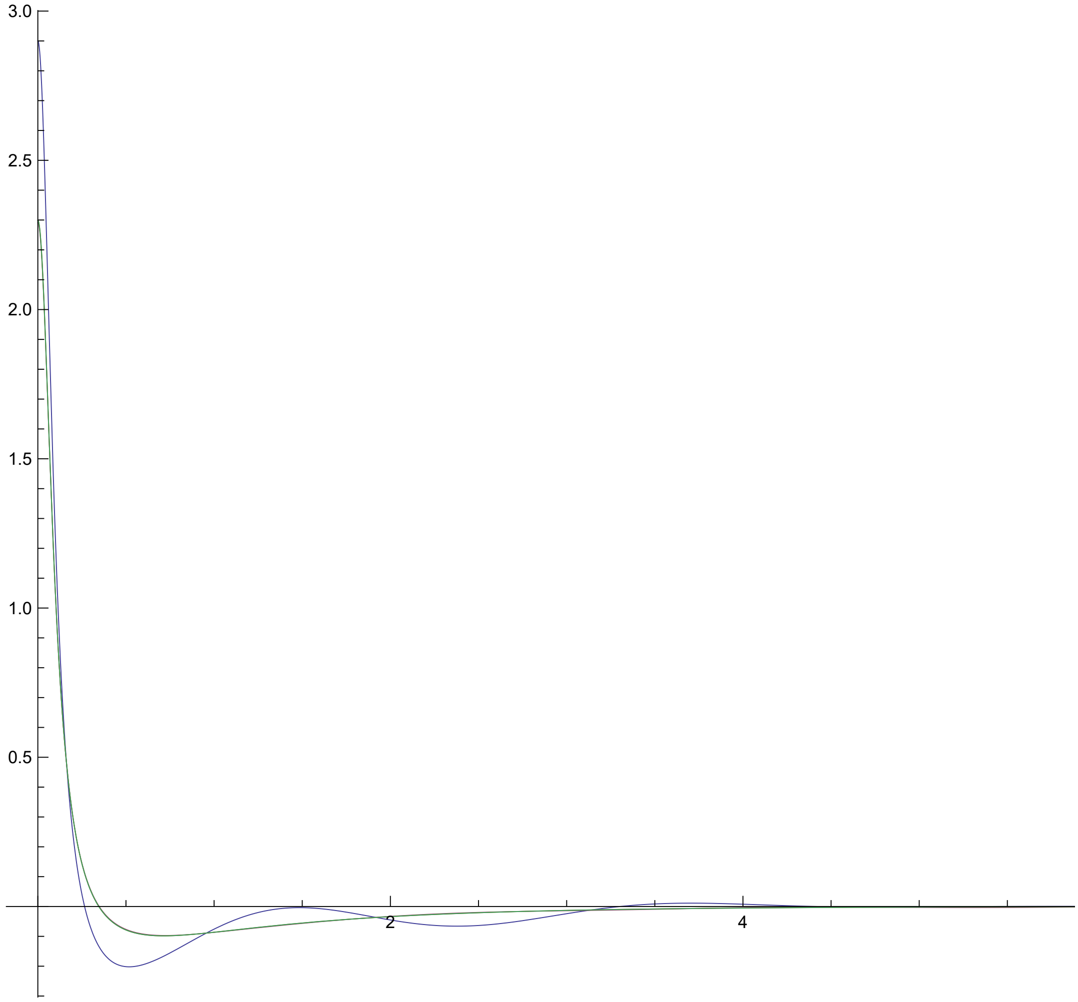
```
Timing[Module[{S1 = 0.02, S2 = 0.02, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0.3, ρsinf = 0.3, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.02, Σ2 = 0.02,
   β = 10, τ = 40, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs, vol1,
   vol2, v1, v2, printflag = 0, Σinf, M, Q, Σ, ρ, Y, zmax1, zmax2, K = 0.0001, ω1},
  M = ( M1              ρm1 √(M1 M2)
        ρm2 √(M1 M2)    M2           );
  Σinf = ( θ1                 √(θ1 θ2) ρsinf
           √(θ1 θ2) ρsinf     θ2             );
  Σ = ( Σ1              √(Σ1 Σ2) ρs
        √(Σ1 Σ2) ρs     Σ2          );
  ρ = {ρ1, ρ2};
  Q = CholeskyDecomposition[- ((M.Σinf + Σinf. Transpose[M]) / 2)] / β;
  Y = {Log[S1], Log[S2]};
  scope1 = 4 / (√((Σ1 + Σ2 + θ1 + θ2) / 4 τ));
  scope2 = 6 / (√((Σ1 + Σ2 + θ1 + θ2) / 4 τ));
  Table[Plot[NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
      τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, ω2], {ω2, 0 , scope2}, PlotPoints → nb,
     PlotRange → All, PlotLabel → "K=" <> ToString[K] <> " ω1=" <> ToString[ω1]],
   {ω1, {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 1, 1.5, 2}}]]]
```

{212.141,

{
K=0.0001 ω1=0
, 
K=0.0001 ω1=0.1
, 
K=0.0001 ω1=0.2
, 
K=0.0001 ω1=0.3
, 
K=0.0001 ω1=0.4
, 
K=0.0001 ω1=0.5
, 
K=0.0001 ω1=0.6
, 
K=0.0001 ω1=0.7
, 
K=0.0001 ω1=0.8
, 
K=0.0001 ω1=1
, 
K=0.0001 ω1=1.5
, 
K=0.0001 ω1=2
}}

```
Timing[Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0.8, ρsinf = 0.8, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.04, Σ2 = 0.04,
   β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs, vol1,
   vol2, ν1, ν2, printflag = 0, Σinf, M, Q, Σ, ρ, Y, zmax1, zmax2, K = 0.0001, ω1},
  M = ( M1              ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2] M2              );
  Σinf = ( θ1                  Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf   θ2                );
  Σ = ( Σ1                Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs    Σ2             );
  ρ = {ρ1, ρ2};
  Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
  Y = {Log[S1], Log[S2]};
  scope1 = 4/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 6/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Plot[
   {CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K, τ, M,
        Q, ρ, Σ, Y, β, λ1, λ2, ω1, #]) &, RiemanCoeffs[nb, 0, scope2 / 3]],
    CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[K,
        τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, #]) &, RiemanCoeffs[nb, 0, scope2 / 1.5]],
    CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
        K, τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, #]) &, RiemanCoeffs[nb, 0, scope2]],
    CoeffBasedIntegrate[(NewSymetrizedSuperBiHestonVanillaReducedIntegrand2[
        K, τ, M, Q, ρ, Σ, Y, β, λ1, λ2, ω1, #]) &,
     RiemanCoeffs[nb, 0, scope2 * 1.3]]}, {ω1, 0, scope2}, PlotRange → All]]]
```

{1092.16,




```
AdaptativeIntegrate[Func_, LegendreCoef0_, LegendreCoefn_, period0_, periodn_, ϵ_] :=
 Module[{sum = 0.0, increment = ϵ + 1, i = 0},
  sum = CoeffBasedIntegrate[Func,
    LegendreCoeffsFromLegendre[LegendreCoef0, 0, period0]];
  While[Abs[increment] > ϵ, increment = CoeffBasedIntegrate[
     Func, LegendreCoeffsFromLegendre[LegendreCoefn,
      period0 + i * periodn, period0 + (i + 1) * periodn]];
   i += 1;
   sum += increment];
  {i, sum}]
```

```
AdaptativeIntegrate[Func_, LegendreCoef0_, period0_, periodn_, ϵ_] :=
 Module[{sum = 0.0, increment = ϵ + 1, i = 0},
  sum = CoeffBasedIntegrate[Func,
    LegendreCoeffsFromLegendre[LegendreCoef0, 0, period0]];
  While[Abs[increment] > ϵ, increment = Func[period0 + (i + 1 / 2) * periodn] periodn;
   i += 1;
   sum += increment];
  {i, sum}]
```

```
coefs = LegendreCoeffs[15]; coefs2 = LegendreCoeffs[5];
```

```
a = AdaptativeIntegrate[((1 + Sin[#^2]) Exp[-#]) &, coefs, coefs2, 2, 0.5, 0.000001];
{a, a[[2]] - NIntegrate[((1 + Sin[x^2]) Exp[-x]), {x, 0, ∞}]}
```

$\{\{23, 1.27051\}, -1.32787 \times 10^{-6}\}$

```
a = AdaptativeIntegrate[((1 + Sin[#^2]) Exp[-#]) &, coefs, 4, 0.01, 0.0000001];
{a, a[[2]] - NIntegrate[((1 + Sin[x^2]) Exp[-x]), {x, 0, ∞}]}
```

```
{{16, 1.25513}, -0.0153883}
```

```
AdaptativeIntegrate[Func_, LegendreCoef0_, LegendreCoefn_, period0_, periodn_, ϵ_] :=
 Module[{sum = 0.0, increment = ϵ + 1, i = 0},
  sum = CoeffBasedIntegrate[Func,
    LegendreCoeffsFromLegendre[LegendreCoef0, 0, period0]];
  While[Abs[increment] > ϵ, increment = CoeffBasedIntegrate[
     Func, LegendreCoeffsFromLegendre[LegendreCoefn,
      period0 + i * periodn, period0 + (i + 1) * periodn]];
   i += 1;
   sum += increment];
  {i, sum}]
```

```
AdaptativeIntegrate[Func_, LegendreCoef0_, period0_, periodn_, ϵ_] :=
 Module[{sum = 0.0, increment = ϵ + 1, i = 0},
  sum = CoeffBasedIntegrate[Func,
    LegendreCoeffsFromLegendre[LegendreCoef0, 0, period0]];
  While[Abs[increment] > ϵ, increment = Func[period0 + (i + 1 / 2) * periodn] periodn;
   i += 1;
   sum += increment];
  {i, sum}]
```

```
NewSuperBiHestonVanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_, λ1_,
  λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonVanillaAux[K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2,
   {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
   {LegendreCoef2, period2, period2n, ϵ2}, printflag] /; K ≥ 0
```

```
NewSuperBiHestonVanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_, λ1_,
  λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonVanillaAux[K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2,
    {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag] - K /; K < 0
```

```
NewSuperBiHestonVanillaAux[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_, λ1_,
  λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 2/(2 π)^2 Module[{a, res, res2, Y = {Log[S[[1]]], Log[S[[2]]]}, Q},
   Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
   If[printflag == 1,
    Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2}]];
   If[printflag == 2, Print[
     "{NbLegendreCoef1,NbLegendreCoef1n,period1,period1n,ϵ1}=",
     {Length[LegendreCoef1], Length[LegendreCoef1n], period1, period1n, ϵ1}]];
   If[printflag == 2, Print["{NbLegendreCoef2,period2,period2n,ϵ2}=",
     {Length[LegendreCoef2], period2, period2n, ϵ2}]];
   If[printflag == 1, Print["{LegendreCoef1,LegendreCoef1n,period1,period1n,ϵ1}=",
     {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
     " {LegendreCoef2,period2,period2n,ϵ2}=",
     {LegendreCoef2, period2, period2n, ϵ2}]];
   res = AdaptativeIntegrate[Function[ω2, a = AdaptativeIntegrate[Function[ω1,
          NewSymetrizedSuperBiHestonVanillaReducedIntegrand[K, τ, M, Q, ρ, Σ, Y, β,
           λ1, λ2, ω1, ω2]], LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1];
      If[printflag == 2, Print["Integ_2=", a]];
      a[[2]]], LegendreCoef2, period2, period2n, ϵ2];
   If[printflag == 2, Print["Integ_1=", res]];
   res[[2]]]
```

```
Timing[Module[{S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.01, M2 = -0.02,
   θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5,
   ρ2 = 0.8,  Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1,
   scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n, period1, period1n, ϵ1,
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2              );
  Σinf = ( θ1                    Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf     θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  {NewSuperBiHestonVanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
    {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag],
   NewSuperBiHestonVanilla[0, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
    {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag],
   NewSuperBiHestonVanilla[-K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
    {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag]}]]
```

```
{6.437, {0.00877397, 0.00877929, 0.0087846}}
```

Correlation smile effect

The purpose of the whole framework is be able to model correlation smile effect that is visualized here through the rotation effect of the smile due to croos term inside the mean reverting matrix M

```
Timing[Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15,
```

```
θ2 = 0.15, ρs = 0.8, ρsinf = 0.8, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15,
 Σ1 = 0.04, Σ2 = 0.04, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2,
 nb = 100, inter, Lcoefs, vol1, vol2, ν1, ν2, printflag = 0, Σinf, M, Σ},
strikes = {-0.03, -0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002,
  -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.03};
```

$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\ \theta 2}\ \rho sinf \\ \sqrt{\theta 1\ \theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$

$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\ \Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\ \Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$

$scope1 = \frac{4}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};\ scope2 = \frac{6}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};$

$\rho m1 = 0.;\ \rho m2 = 0.;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

```
smile00 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
    NewSuperBiHestonVanilla2[strikes〚i〛, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
     β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter00 = Interpolation[smile00, InterpolationOrder → 1];
```

$\rho m1 = 0.5;\ \rho m2 = 0.;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

```
smile10 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
    NewSuperBiHestonVanilla2[strikes〚i〛, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
     β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter10 = Interpolation[smile10, InterpolationOrder → 1];
```

$\rho m1 = 0.;\ \rho m2 = 0.5;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

```
smile01 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
    NewSuperBiHestonVanilla2[strikes〚i〛, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
     β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter01 = Interpolation[smile01, InterpolationOrder → 1];
```

$\rho m1 = -0.5;\ \rho m2 = 0.;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

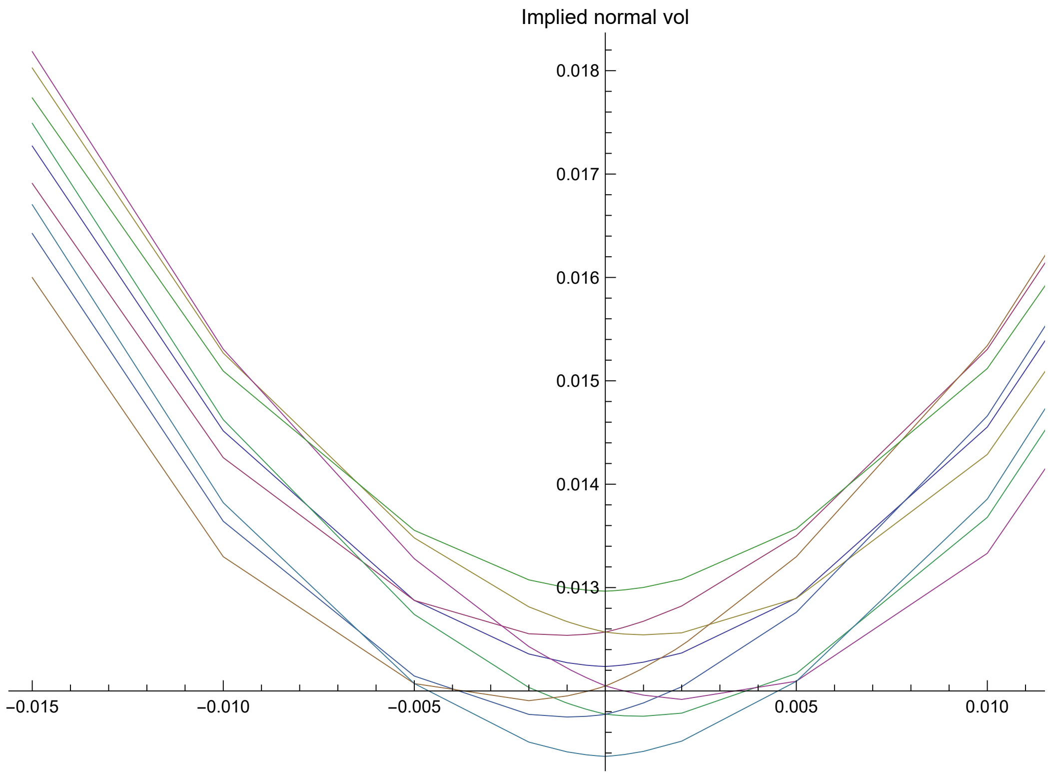
```
smile50 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
    NewSuperBiHestonVanilla2[strikes〚i〛, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
     β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter50 = Interpolation[smile50, InterpolationOrder → 1];
```

$\rho m1 = 0.;\ \rho m2 = -0.5;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$

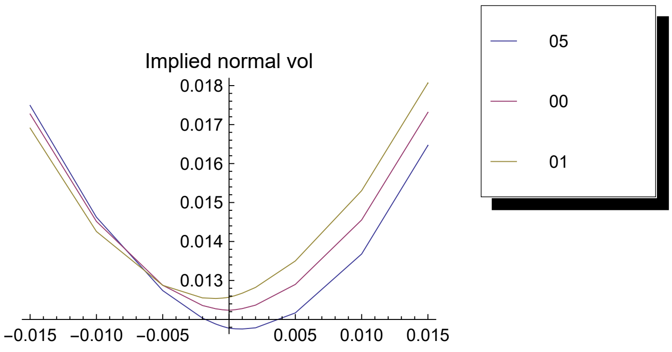
```
smile05 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
    NewSuperBiHestonVanilla2[strikes〚i〛, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
     β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter05 = Interpolation[smile05, InterpolationOrder → 1];
```

```
ρm1 = 0.5; ρm2 = 0.5; M = ( M1            ρm1 √(M1 M2) );
                          ( ρm2 √(M1 M2)  M2           )
smile11 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
      β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter11 = Interpolation[smile11, InterpolationOrder → 1];
ρm1 = -0.5; ρm2 = -0.5; M = ( M1            ρm1 √(M1 M2) );
                            ( ρm2 √(M1 M2)  M2           )
smile55 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
      β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter55 = Interpolation[smile55, InterpolationOrder → 1];
ρm1 = -0.5; ρm2 = 0.5; M = ( M1            ρm1 √(M1 M2) );
                           ( ρm2 √(M1 M2)  M2           )
smile51 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
      β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter51 = Interpolation[smile51, InterpolationOrder → 1];
ρm1 = 0.5; ρm2 = -0.5; M = ( M1            ρm1 √(M1 M2) );
                           ( ρm2 √(M1 M2)  M2           )
smile15 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
      β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
inter15 = Interpolation[smile15, InterpolationOrder → 1];
Lcoefs = LegendreCoeffs[40];
ν1 = 0.2; vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, M1, ν1, Lcoefs];
ν2 = 0.2; vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, M2, ν2, Lcoefs];
smile2 =
  Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
      S1, S2, vol1, vol2, ρs, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
inter2 = Interpolation[smile2, InterpolationOrder → 1];
Plot[{inter00[x], inter01[x], inter10[x], inter05[x],
   inter50[x], inter15[x], inter51[x], inter11[x], inter55[x]},
  {x, strikes[[1]] / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
  PlotLegend → {"00", "01", "10", "05", "50", "15", "51", "11", "55"},
  LegendPosition → {1, 0}]
]]
```

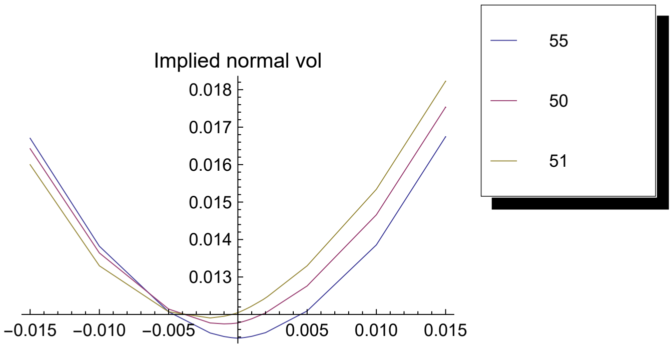
{2948.37,



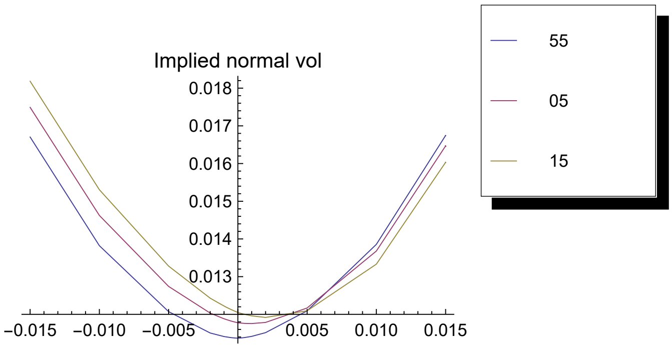

```
Plot[{inter05[x], inter00[x], inter01[x]},
 {x, strikes⟦1⟧ / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
 PlotLegend → {"05", "00", "01"}, LegendPosition → {1, 0}]
```



```
Plot[{inter55[x], inter50[x], inter51[x]},
 {x, strikes[[1]] / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
 PlotLegend → {"55", "50", "51"}, LegendPosition → {1, 0}]
```



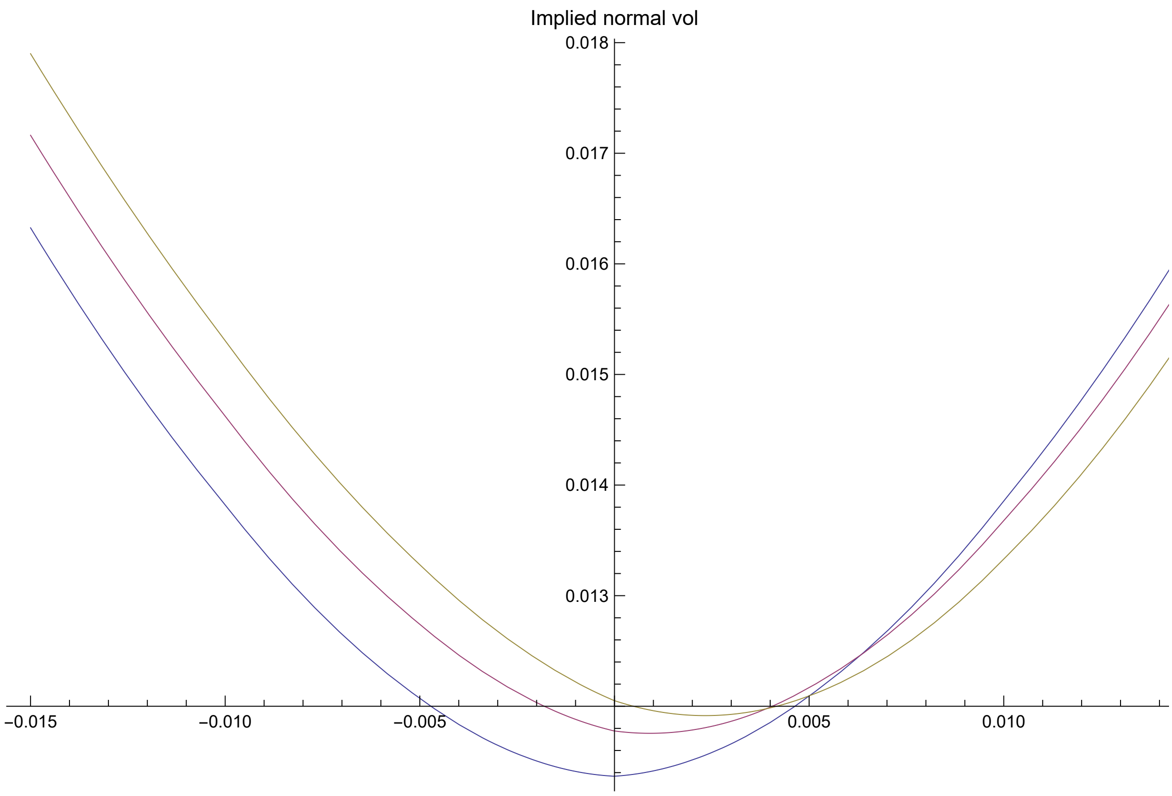

```
Plot[{inter55[x], inter05[x], inter15[x]},
 {x, strikes[[1]] / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
 PlotLegend → {"55", "05", "15"}, LegendPosition → {1, 0}]
```



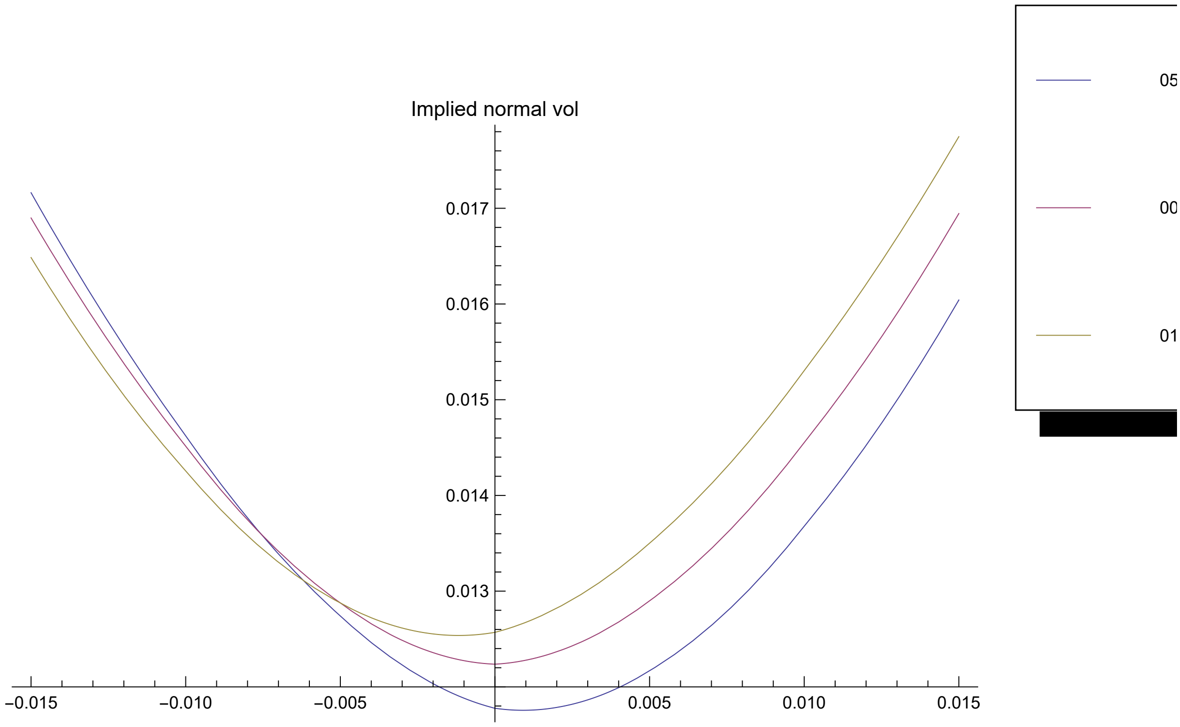

```
AjustOrder[n_] := Module[{},
   inter15 = Interpolation[smile15, InterpolationOrder → n];
   inter51 = Interpolation[smile51, InterpolationOrder → n];
   inter55 = Interpolation[smile55, InterpolationOrder → n];
   inter11 = Interpolation[smile11, InterpolationOrder → n];
   inter05 = Interpolation[smile05, InterpolationOrder → n];
   inter50 = Interpolation[smile50, InterpolationOrder → n];
   inter10 = Interpolation[smile10, InterpolationOrder → n];
   inter00 = Interpolation[smile00, InterpolationOrder → n];
   inter01 = Interpolation[smile01, InterpolationOrder → n];
  ]
```

L' ordre de l' interpolation ne change rien car le smile presente reelement un coin

```
AjustOrder[2];
Plot[{inter55[x], inter05[x], inter15[x]},
 {x, strikes⟦1⟧ / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
 PlotLegend → {"55", "05", "15"}, LegendPosition → {1, 0}]
```



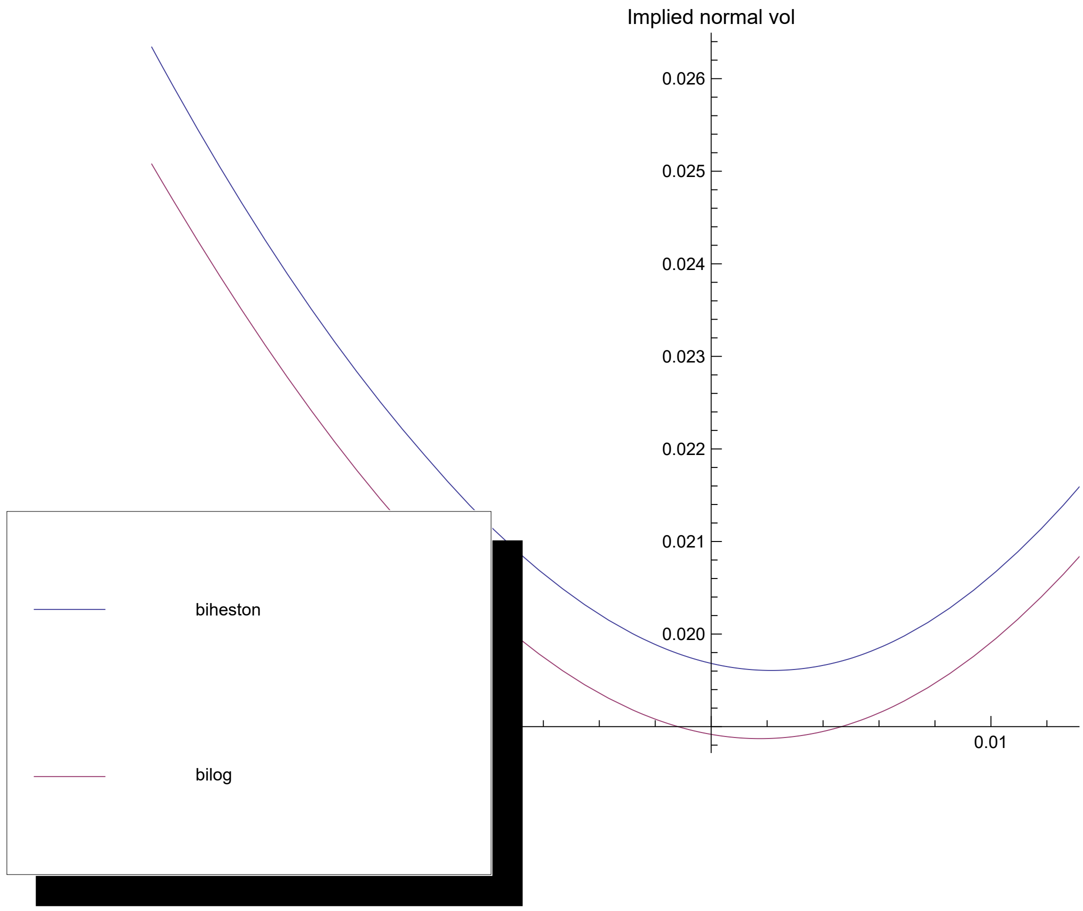
```
Plot[{inter05[x], inter00[x], inter01[x]},
 {x, strikes⟦1⟧ / 2, Last[strikes] / 2}, PlotLabel → "Implied normal vol",
 PlotLegend → {"05", "00", "01"}, LegendPosition → {1, 0}]
```



Influence of the stochastic correlation

```
Timing[Module[
  {S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041, ρs = 0.6,
   ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0,
    0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02};
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
       {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
       {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 2];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
       S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter000[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

{19.905,


Implied normal vol
0.026
0.025
0.024
0.023
0.022
0.021
0.020
0.01
biheston
bilog

```
Timing[
 Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15, ρs = 0.8,
   ρsinf = 0.8, ρm1 = 0., ρm2 = 0., ρ1 = -0.15, ρ2 = -0.15, Σ1 = 0.04, Σ2 = 0.04, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, nb = 60, inter, Lcoefs = LegendreCoeffs[40],
   vol1, vol2, ν1 = 0.01, ν2 = 0.01, printflag = 0, Σinf, M, Σ},
  strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002,
    -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02};
```

$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\, M2} \\ \rho m2\ \sqrt{M1\, M2} & M2 \end{pmatrix};$$

$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\, \theta 2}\ \rho sinf \\ \sqrt{\theta 1\, \theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$$

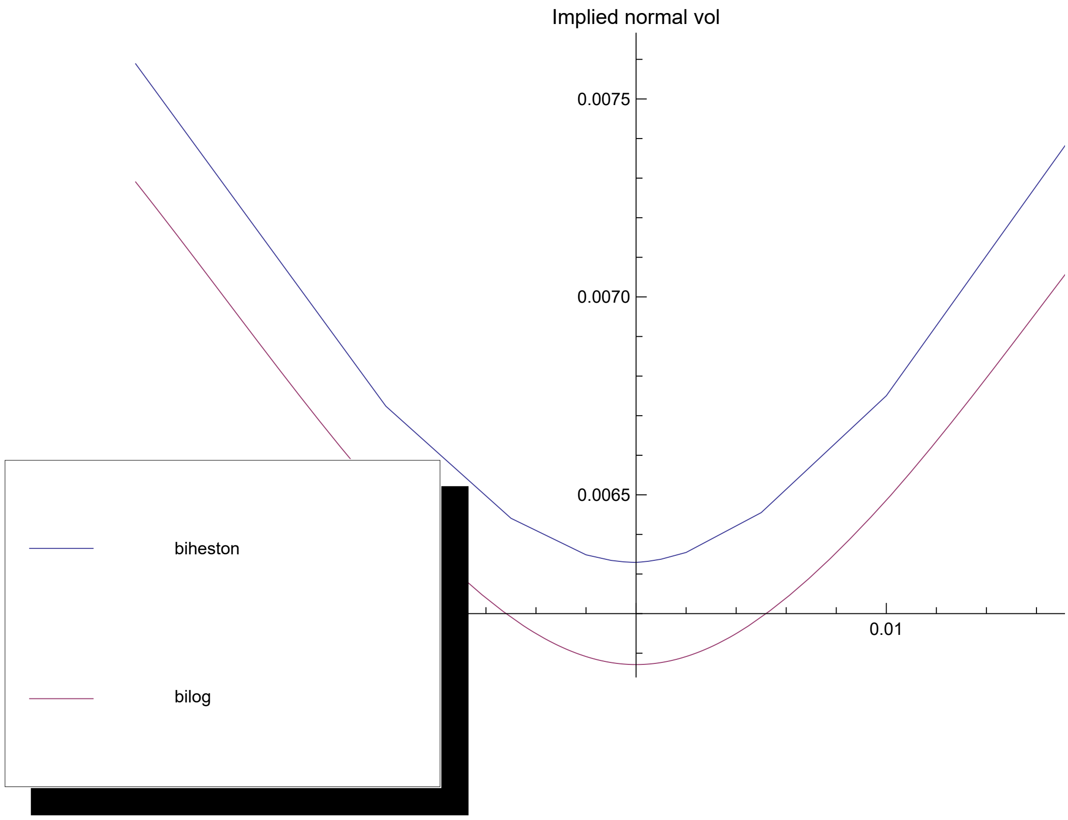
$$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\, \Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\, \Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

$$scope1 = \frac{5}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};\ scope2 = \frac{8}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};$$

$$\rho m1 = 0.;\ \rho m2 = 0.;\ M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\, M2} \\ \rho m2\ \sqrt{M1\, M2} & M2 \end{pmatrix};$$

```
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
       β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 1];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
       S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter000[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

{214.938,



smile000

```
{{-0.02, 0.00719348}, {-0.01, 0.00655477}, {-0.005, 0.00630812},
 {-0.002, 0.00621388}, {-0.001, 0.00619576}, {-0.0005, 0.0061896},
 {-0.0002, 0.00618687}, {-0.0001, 0.00618613}, {0, 0.00618547}, {0.0001, 0.00618639},
 {0.0002, 0.00618739}, {0.0005, 0.00619087}, {0.001, 0.00619828},
 {0.002, 0.00621879}, {0.005, 0.00631958}, {0.01, 0.00657606}, {0.02, 0.00723526}}
```

```
Timing[Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15,
    θ2 = 0.15, ρs = 0.8, ρsinf = 0.8, ρm1 = 0., ρm2 = 0.5, ρ1 = -0.15, ρ2 = -0.15,
    Σ1 = 0.04, Σ2 = 0.04, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2,
    nb = 40, inter, Lcoefs, vol1, vol2, ν1, ν2, printflag = 0, Σinf, M, Σ},
  strikes = {-0.03, -0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002,
    -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.03};
```

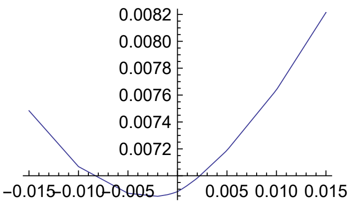
$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$$

$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\ \theta 2}\ \rho sinf \\ \sqrt{\theta 1\ \theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$$

$$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\ \Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\ \Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

$$scope1 = \frac{4}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};\ scope2 = \frac{6}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};$$

```
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
       β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 1];
  Plot[inter000[x], {x, strikes[[1]] / 2, Last[strikes] / 2}]
 ]]
```

{118.609, 

}

```
Timing[Module[{S1 = 0.04, S2 = 0.04, M1 = -0.075, M2 = -0.075, θ1 = 0.15,
   θ2 = 0.15, ρs = 0.8, ρsinf = 0.8, ρm1 = 0., ρm2 = 0.5, ρ1 = -0.15, ρ2 = -0.15,
   Σ1 = 0.04, Σ2 = 0.04, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2,
   nb = 40, inter, Lcoefs, vol1, vol2, ν1, ν2, printflag = 0, Σinf, M, Σ},
  strikes = {-0.03, -0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002,
    -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.01, 0.02, 0.03};
```

$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\ M2} \\ \rho m2\ \sqrt{M1\ M2} & M2 \end{pmatrix};$$

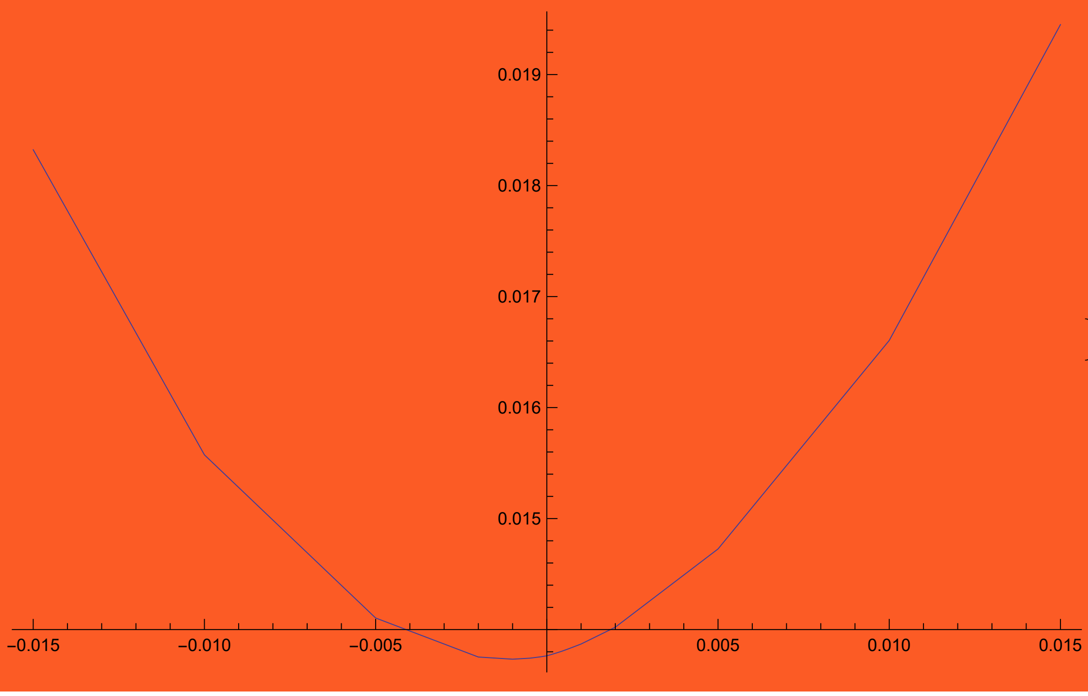
$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\ \theta 2}\ \rho sinf \\ \sqrt{\theta 1\ \theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$$

$$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\ \Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\ \Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

$$scope1 = \frac{4}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};\ scope2 = \frac{6}{\sqrt{\frac{\Sigma 1+\Sigma 2+\theta 1+\theta 2}{4}\ \tau}};$$

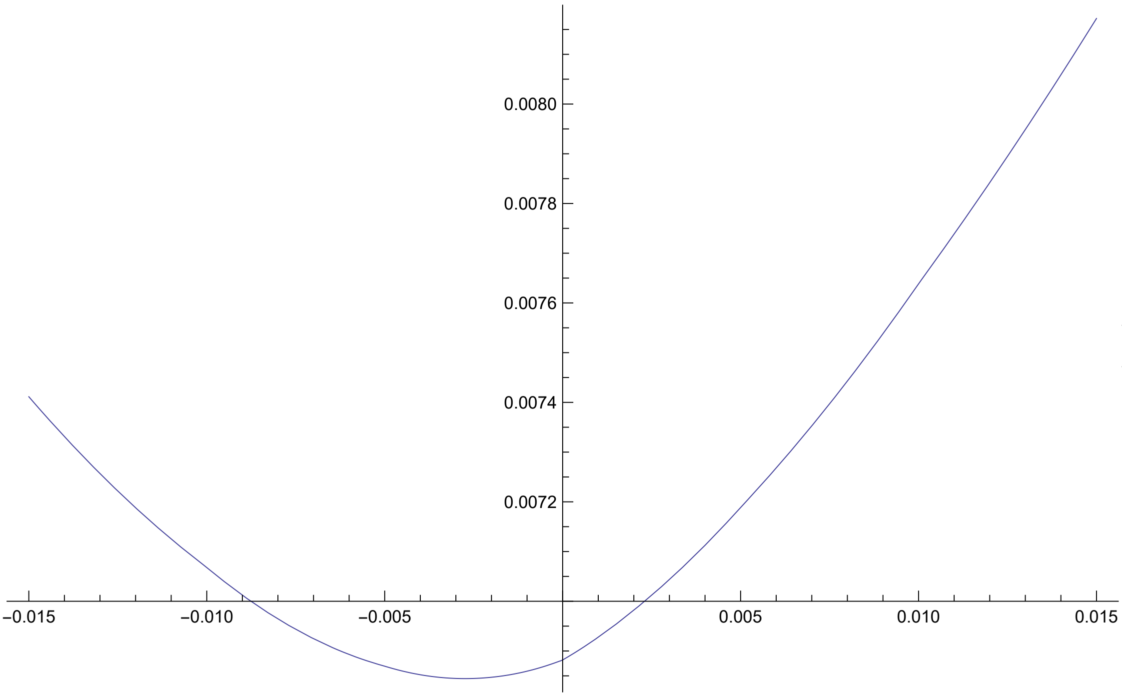
```
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla2[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2},
       β, λ1, λ2, scope1, scope2, nb, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 1];
  Plot[inter000[x], {x, strikes[[1]] / 2, Last[strikes] / 2}]
 ]]
```

{57.596,



}

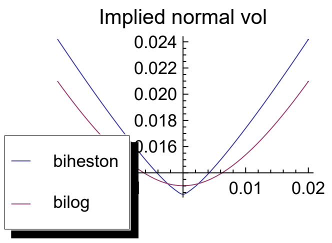
{109.11,



}

```
Timing[Module[{ν1 = 0.1, ν2 = 0.1, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8, ρinf12 = 0.8, β, integflag = 0,
   τ = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 60, flag = 1, det,
   Lcoefs = LegendreCoeffs[40], vol1, vol2, spdopt, strikes, βmul = 0.9},
  strikes = {-0.02, -0.015, -0.01, -0.0075, -0.005, -0.003,
    -0.001, 0, 0.001, 0.003, 0.005, 0.0075, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2];
  Print["β=", β];
  β *= βmul;
  z1max = 2; z2max = 4;
  smile1 = Table[{strikes⟦i⟧, NormalImplicitVol[S1 - S2, strikes⟦i⟧, τ,
      NewSuperBiHestonVanilla[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
       {ρ1, ρ2}, {ρ12, ρinf12}, {Σ1, Σ2}, {S1, S2}, β, strikes⟦i⟧, τ, λ1,
       λ2, z1max, z2max, Nb, integflag, 0]]}, {i, 1, Length[strikes]}];
  inter1 = Interpolation[smile1];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, Σinf1, ρs1, χ1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, Σinf2, ρs2, χ2, ν2, Lcoefs];

  smile2 =
   Table[{strikes⟦i⟧, NormalImplicitVol[S1 - S2, strikes⟦i⟧, τ, LogNormalSpreadOption[
       S1, S2, vol1, vol2, ρ12, strikes⟦i⟧, τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter1[x], inter2[x]},
   {x, strikes⟦1⟧, Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

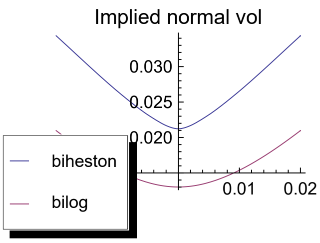
β=9.

{44.07, 

}

```
Timing[Module[{ν1 = 0.1, ν2 = 0.1, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8, ρinf12 = 0.8, β, integflag = 0,
   τ = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 80, flag = 1, det,
   Lcoefs = LegendreCoeffs[40], vol1, vol2, spdopt, strikes, βmul = 0.9},
  strikes = {-0.02, -0.01, -0.0075, -0.005, -0.002, -0.001, -0.0005, -0.0002,
    -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2];
  Print["β=", β];
  β *= βmul;
  z1max = 8; z2max = 12;
  smile1 = Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ,
       NewSuperBiHestonVanilla[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
        {ρ1, ρ2}, {ρ12, ρinf12}, {Σ1, Σ2}, {S1, S2}, β, strikes〚i〛, τ, λ1,
        λ2, z1max, z2max, Nb, integflag, 0]]}, {i, 1, Length[strikes]}];
  inter1 = Interpolation[smile1];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, Σinf1, ρs1, χ1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, Σinf2, ρs2, χ2, ν2, Lcoefs];

  smile2 =
   Table[{strikes〚i〛, NormalImplicitVol[S1 - S2, strikes〚i〛, τ, LogNormalSpreadOption[
        S1, S2, vol1, vol2, ρ12, strikes〚i〛, τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter1[x], inter2[x]},
   {x, strikes〚1〛, Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

β=9.

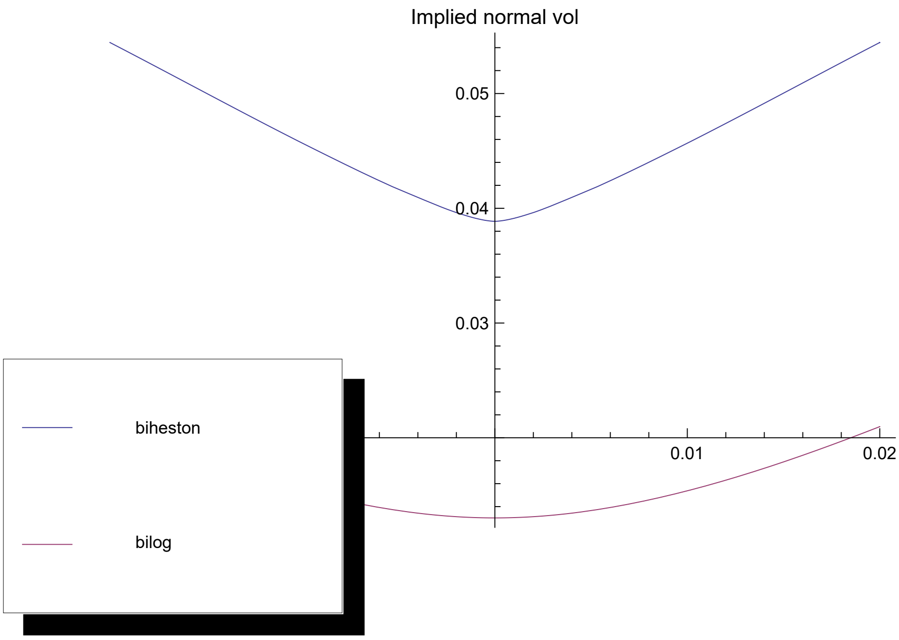
{96.19, 

}

```
Timing[Module[{ν1 = 0.1, ν2 = 0.1, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8, ρinf12 = 0.8, β, integflag = 0,
   τ = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 100, flag = 1, det,
   Lcoefs = LegendreCoeffs[40], vol1, vol2, spdopt, strikes, βmul = 0.9},
  strikes = {-0.02, -0.01, -0.0075, -0.005, -0.002, -0.001, -0.0005, -0.0002,
    -0.0001, 0, 0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.02};
  β = 1.2 βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2];
  Print["β=", β];
  β *= βmul;
  z1max = 15; z2max = 20;
  smile1 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
      NewSuperBiHestonVanilla[{ν1, ρs1, χ1, Σinf1}, {ν2, ρs2, χ2, Σinf2},
       {ρ1, ρ2}, {ρ12, ρinf12}, {Σ1, Σ2}, {S1, S2}, β, strikes[[i]], τ, λ1,
       λ2, z1max, z2max, Nb, integflag, 0]]}, {i, 1, Length[strikes]}];
  inter1 = Interpolation[smile1];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, Σinf1, ρs1, χ1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, Σinf2, ρs2, χ2, ν2, Lcoefs];

  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
       S1, S2, vol1, vol2, ρ12, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter1[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

β=10.8

{149.262,



}

Option Vanille Partielle $(S_1 - K)^+$

x2>0

integration1 = Simplify$\left[\int_{\text{Log}[K]}^{\infty} e^{i\,k1\,x1}\left(e^{x1} - K\right) dx1\right]$

$\text{If}\left[\text{Im}[k1] > 1,\ -\frac{K^{1+i\,k1}}{k1\,(-i + k1)},\right.$
$\left.\text{Integrate}\left[e^{x1+i\,k1\,x1} - e^{i\,k1\,x1} K,\ \{x1, \text{Log}[K], \infty\},\ \text{Assumptions} \to \text{Im}[k1] \le 1\right]\right]$

integration11 = Simplify[integration1, Im[k1] > 1]

$-\frac{K^{1+i\,k1}}{k1\,(-i + k1)}$

integration12 = Simplify$\left[\text{Integrate}\left[e^{i\,k2\,x2}\ \text{integration11},\ \{x2, 0, \infty\}\right],\ K > 0\right]$

$-\frac{K^{1+i\,k1}\ \text{If}\left[\text{Im}[k2] > 0,\ \frac{i}{k2},\ \text{Integrate}\left[e^{i\,k2\,x2},\ \{x2, 0, \infty\},\ \text{Assumptions} \to \text{Im}[k2] \le 0\right]\right]}{k1\,(-i + k1)}$

Simplify[integration12, Im[k2] > 0]

$\frac{i\,K^{1+i\,k1}}{i\,k1\,k2 - k1^2\,k2}$

x2<0

integration1 = Simplify$\left[\int_{\text{Log}[K]}^{\infty} e^{i\,k1\,x1}\left(e^{x1} - K\right) dx1\right]$

$\text{If}\left[\text{Im}[k1] > 1,\ -\frac{K^{1+i\,k1}}{k1\,(-i + k1)},\right.$
$\left.\text{Integrate}\left[e^{x1+i\,k1\,x1} - e^{i\,k1\,x1} K,\ \{x1, \text{Log}[K], \infty\},\ \text{Assumptions} \to \text{Im}[k1] \le 1\right]\right]$

integration11 = Simplify[integration1, Im[k1] > 1]

$-\frac{K^{1+i\,k1}}{k1\,(-i + k1)}$

integration12 = Simplify$\left[\text{Integrate}\left[e^{i\,k2\,x2}\ \text{integration11},\ \{x2, -\infty, 0\}\right],\ K > 0\right]$

$-\frac{K^{1+i\,k1}\ \text{If}\left[\text{Im}[k2] < 0,\ -\frac{i}{k2},\ \text{Integrate}\left[e^{-i\,k2\,x2},\ \{x2, 0, \infty\},\ \text{Assumptions} \to \text{Im}[k2] \ge 0\right]\right]}{k1\,(-i + k1)}$

Simplify[integration12, Im[k2] < 0]

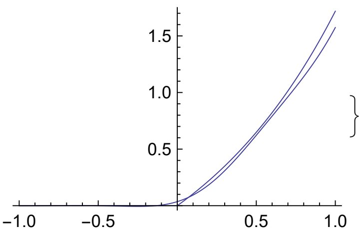
$-\frac{K^{1+i\,k1}}{k1\,k2 + i\,k1^2\,k2}$

```
FirstUnderlyingPayOff[x1_, x2_, K_] := Max[e^x1 - K, 0]
```

$$\text{FirstUnderlyingVanillaFourierPayOffDroite[k1\_, k2\_, K\_]} := \frac{i\,K^{1+i\,k1}}{i\,k1\,k2 - k1^2\,k2}$$

$$\text{FirstUnderlyingVanillaFourierPayOffGauche[k1\_, k2\_, K\_]} := -\frac{K^{1+i\,k1}}{k1\,k2 + i\,k1^2\,k2}$$

```
Timing[Module[{x2 = 0.1, K = 1, λ1 = 2, λ2 = 1.2, coeff = RiemanCoeffs[40, -8, 8]},
  g1 = ListPlot[Table[{i / 100, Re[Module[{x1 = i / 100}, 1/(2 π)^2 CoeffBasedIntegrate[
        (ⅇ^(-ⅈ x1 (#1+ⅈ λ1)-ⅈ x2 (#2-ⅈ λ2)) FirstUnderlyingVanillaFourierPayOffGauche[
            #1 + ⅈ λ1, (#2 - ⅈ λ2), K] + ⅇ^(-ⅈ x1 (#1+ⅈ λ1)-ⅈ x2 (#2+ⅈ λ2))
           FirstUnderlyingVanillaFourierPayOffDroite[#1 + ⅈ λ1, (#2 + ⅈ λ2), K]) &,
        coeff, coeff]]]}, {i, -100, 100}], Joined → True];
  g2 = ListPlot[Table[{i / 100, Module[{x1 = i / 100},
       Re[FirstUnderlyingPayOff[x1, x2, K]]]}, {i, -100, 100}], Joined → True];
  Show[g1, g2, PlotRange → All]]]
```

{37.532,  }

```
SymetrizedSuperBiHestonUnderlying1VanillaIntegrand[
  K_, τ_, M_, Q_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 Module[{x1 = Y[[1]], x2 = Y[[2]], k1 = (ω1 + ⅈ λ1), Sk1 = (-ω1 + ⅈ λ1),
   k2 = (ω2 + ⅈ λ2), Sk2 = (-ω2 + ⅈ λ2), Sk1A = (-ω1 + ⅈ λ1), k1A = (ω1 - ⅈ λ1),
   k2A = (ω2 - ⅈ λ2), α, αA, Symα, SymαA, α2, αA2, Symα2, SymαA2, propagatorDroit,
   SympropagatorDroit, propagatorGauche, SympropagatorGauche, propagatorDroit2,
   SympropagatorDroit2, propagatorGauche2, SympropagatorGauche2},
  Re[α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
   αA = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2A);
   Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
   SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
   propagatorDroit =
    SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
   SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
   propagatorGauche = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2A}, β, τ];
   SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
   α propagatorDroit FirstUnderlyingVanillaFourierPayOffDroite[k1, k2, K] +
    Symα SympropagatorDroit FirstUnderlyingVanillaFourierPayOffDroite[Sk1, k2, K] +
    αA propagatorGauche FirstUnderlyingVanillaFourierPayOffGauche[k1, k2A, K] +
    SymαA SympropagatorGauche
     FirstUnderlyingVanillaFourierPayOffGauche[Sk1A, k2A, K]]]
```

```
NewSuperBiHestonUnderlying1Vanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonUnderlying1VanillaAux[K, τ, M, Σinf, ρ, Σ, S, β,
   λ1, λ2, {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
   {LegendreCoef2, period2, period2n, ϵ2}, printflag] /; K ≥ 0
```

```
NewSuperBiHestonUnderlying1Vanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonUnderlying1VanillaAux[K, τ, M, Σinf, ρ, Σ, S, β,
    λ1, λ2, {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag] - K /; K < 0
```

```
NewSuperBiHestonUnderlying1VanillaAux[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 2/(2 π)^2 Module[{a, res, res2, Y = {Log[S[[1]]], Log[S[[2]]]}, Q},
   Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)]/β;
   If[printflag == 1,
    Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2}]];
   If[printflag == 2, Print[
     "{NbLegendreCoef1,NbLegendreCoef1n,period1,period1n,ϵ1}=",
     {Length[LegendreCoef1], Length[LegendreCoef1n], period1, period1n, ϵ1}]];
   If[printflag == 2, Print["{NbLegendreCoef2,period2,period2n,ϵ2}=",
     {Length[LegendreCoef2], period2, period2n, ϵ2}]];
   If[printflag == 1, Print["{LegendreCoef1,LegendreCoef1n,period1,period1n,ϵ1}=",
     {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
     " {LegendreCoef2,period2,period2n,ϵ2}=",
     {LegendreCoef2, period2, period2n, ϵ2}]];
   res = AdaptativeIntegrate[Function[ω2, a = AdaptativeIntegrate[Function[ω1,
          SymetrizedSuperBiHestonUnderlying1VanillaIntegrand[K, τ, M, Q, ρ, Σ, Y, β,
           λ1, λ2, ω1, ω2]], LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1];
      If[printflag == 2, Print["Integ_2=", a]];
      a[[2]]], LegendreCoef2, period2, period2n, ϵ2];
   If[printflag == 2, Print["Integ_1=", res]];
   res[[2]]]
```

```
Module[{S1 = 0.05, S2 = 0.05, K = 0.05, M1 = -0.01, M2 = -0.02,
  θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5,
  ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1,
  scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n, period1, period1n, ϵ1,
  Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
 scope1 = 2/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
 scope2 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
 Nb1 = 12;
 LegendreCoef1 = LegendreCoeffs[Nb1];
 Nb1n = 8;
 LegendreCoef1n = LegendreCoeffs[Nb1n];
 period1 = scope1;
 period1n = scope1;
 ϵ1 = 0.00001;
 Nb2 = 10;
 LegendreCoef2 = LegendreCoeffs[Nb2];
 period2 = scope2;
 period2n = scope2 / 10;
 ϵ2 = 0.00001;
 M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
 Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
 Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
 Timing[
  NewSuperBiHestonUnderlying1Vanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
   {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
   {LegendreCoef2, period2, period2n, ϵ2}, printflag]]]

{1.156, 0.00918266}
```

```
Off[InterpolatingFunction::"dmval"];
Timing[Module[
  {S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041, ρs = 0.6,
   ρsinf = 0.8, ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
```

```
scope2 = 4 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
Nb1 = 12;
LegendreCoef1 = LegendreCoeffs[Nb1];
Nb1n = 8;
LegendreCoef1n = LegendreCoeffs[Nb1n];
period1 = scope1;
period1n = scope1;
ϵ1 = 0.00001;
Nb1F = 35;
LegendreCoef1F = LegendreCoeffs[Nb1F];
Nb1nF = 12;
LegendreCoef1nF = LegendreCoeffs[Nb1nF];
Nb2 = 20;
LegendreCoef2 = LegendreCoeffs[Nb2];
period2 = scope2;
period2n = scope2 / 10;
ϵ2 = 0.00001;
Nb2F = 20; LegendreCoef2F = LegendreCoeffs[Nb2F];
```

$$M = \begin{pmatrix} M1 & \rho m1 \sqrt{M1\,M2} \\ \rho m2 \sqrt{M1\,M2} & M2 \end{pmatrix};$$

$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\,\theta 2}\ \rho sinf \\ \sqrt{\theta 1\,\theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$$

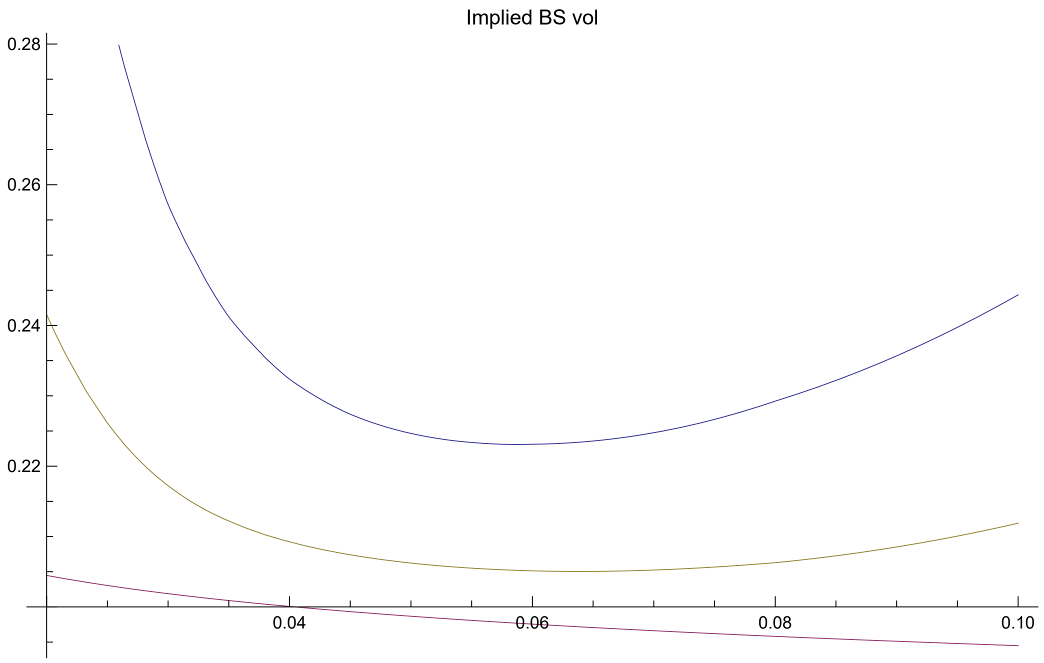
$$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\,\Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\,\Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

```
strikes = {0.02, 0.025, 0.03, 0.035, 0.04, 0.045,
  0.048, 0.049, 0.0499, 0.05, 0.0501, 0.0505, 0.055, 0.06, 0.08, 0.1};
smile000 = Table[{strikes[[i]], ImpVolBS[S1, strikes[[i]], τ,
    NewSuperBiHestonUnderlying1Vanilla[strikes[[i]],
     τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
     {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
     {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter000 = Interpolation[smile000, InterpolationOrder → 2];
smile2 =
 Table[{strikes[[i]], ImpVolHeston2[S1, strikes[[i]], τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs]},
  {i, 1, Length[strikes]}];
inter2 = Interpolation[smile2];
smile3 = Table[
  {strikes[[i]] + 0.0001, ImpVolBS[S1, strikes[[i]] + 0.0001, τ, NewSuperBiHestonVanilla[
     strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, 0.0001}, β, λ1, λ2,
     {LegendreCoef1F, LegendreCoef1nF, period1, period1n / 2, ϵ1}, {LegendreCoef2F,
      period2, period2n / 4, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];

inter3 = Interpolation[smile3];
Plot[{inter000[x], inter2[x], inter3[x]},
 {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied BS vol",
 PlotLegend → {"BiHestonUnderlying1", "Heston", "Bihestonlimit"},
```

```
   LegendPosition → {1, 0}]
]]
```

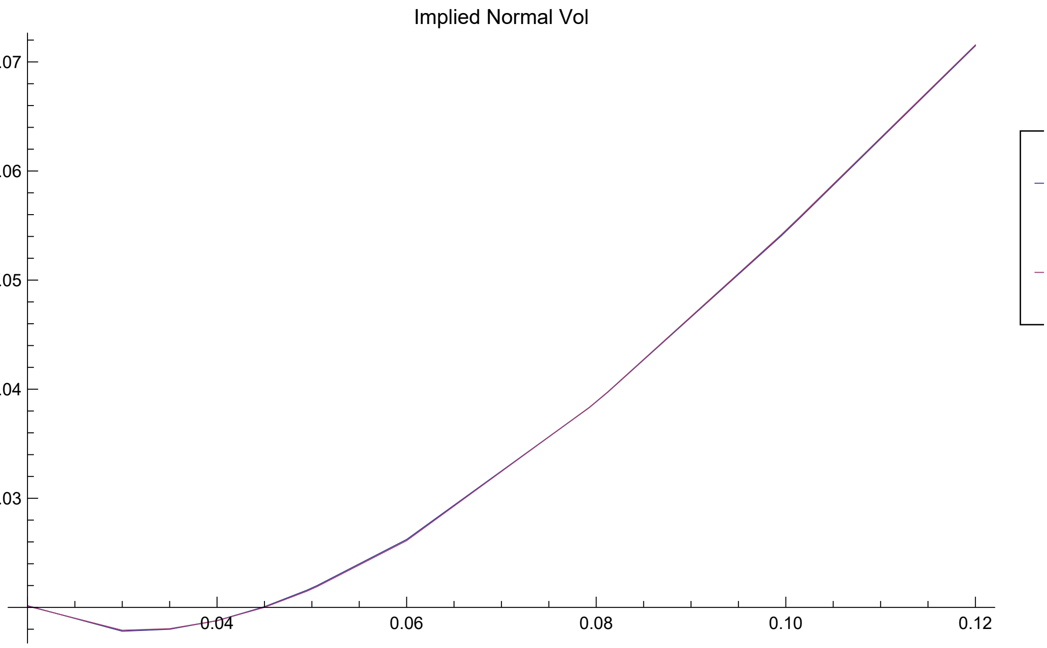
InterpolatingFunction::dmval : Input value {0.0200016} lies outside the range of data in the interpolating function. Extrapolation will be used. ≫

{66.955,

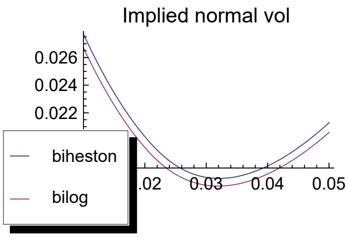


```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
   Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001}, Q,
   strikes = {0.02, 0.03, 0.035, 0.04, 0.045, 0.05, 0.06, 0.08, 0.1, 0.12},
   scope1, scope2, nb = 70, inter, inter2, integflag = 0,
   coeffs = LegendreCoeffs[40], M, Σinf, Σ},
  scope1 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  M = ( M1                ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]   M2             );
  Σinf = ( θ1                    Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf     θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  smile = Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]],
      τ, NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ,
       S, β, λ1, λ2, scope1, scope2, nb, 0]]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  Q = Sqrt[-M1 θ1] / β;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]], τ, HestonCall2[S[[1]],
       strikes[[i]] + S[[2]], τ, Σ1, θ1, ρ1, -M1, Q, coeffs]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  Plot[{inter[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied Normal Vol",
   PlotLegend → {"BiHeston", "Heston"}, LegendPosition → {1, 0}, LegendSize → 0.5]
 ]]
```

{53.586,



```
Timing[Module[
  {S1 = 0.05, S2 = 0.01, K = 0.00001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041, ρs = 0.6,
   ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2              );
  Σinf = ( θ1                     Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf      θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0, 0.0001,
     0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02} + 0.03;
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
       NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
        {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
        {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 2];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
        S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter000[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

{22.588, 
Implied normal vol
0.026
0.024
0.022
biheston
bilog
.02
0.03
0.04
0.05
}

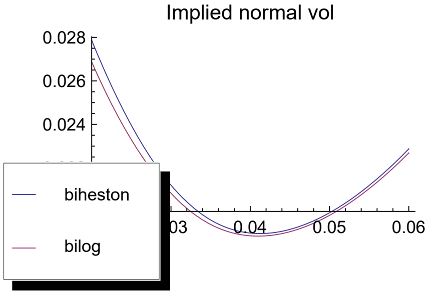
```
Timing[Module[{S1 = 0.05, S2 = 0.0001, K = 0.00001, M1 = -0.01,
    M2 = -0.02, θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3,
    ρm2 = -0.3, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1,
    λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
    period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
    Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 35;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 15;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.000001;
  Nb2 = 16;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 20;
  ϵ2 = 0.000001;
```

$$M = \begin{pmatrix} M1 & \rho m1 \sqrt{M1\,M2} \\ \rho m2 \sqrt{M1\,M2} & M2 \end{pmatrix};$$

$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\,\theta 2}\ \rho sinf \\ \sqrt{\theta 1\,\theta 2}\ \rho sinf & \theta 2 \end{pmatrix};$$

$$\Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\,\Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\,\Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

```
  strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0, 0.0001,
     0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02} + 0.04;
  smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
       NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
        {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
        {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 2];
  vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
  vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
        S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter000[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"biheston", "bilog"}]
 ]]
```

{100.745, 

}

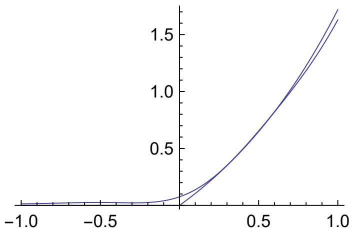
Option Vanille Partielle $(S_2 - K)^+$

```
SecondUnderlyingVanillaFourierPayOffDroite[k1_, k2_, K_] := (ⅈ K^(1+ⅈ k2))/(ⅈ k1 k2 - k2^2 k1)
```

```
SecondUnderlyingVanillaFourierPayOffGauche[k1_, k2_, K_] := -(K^(1+ⅈ k2))/(k1 k2 + ⅈ k2^2 k1)
```

```
SecondUnderlyingPayOff[x1_, x2_, K_] := Max[ⅇ^x2 - K, 0]
```

```
Timing[Module[{x1 = 0.1, K = 1, λ1 = 2, λ2 = 1.2, coeff = RiemanCoeffs[40, -8, 8]},
  g1 = ListPlot[Table[{i / 100, Re[Module[{x2 = i / 100}, 1/(2 π)^2 CoeffBasedIntegrate[
          (ⅇ^(-ⅈ x1 (#1-ⅈ λ1)-ⅈ x2 (#2+ⅈ λ2)) SecondUnderlyingVanillaFourierPayOffGauche[
               #1 - ⅈ λ1, (#2 + ⅈ λ2), K] + ⅇ^(-ⅈ x1 (#1+ⅈ λ1)-ⅈ x2 (#2+ⅈ λ2))
              SecondUnderlyingVanillaFourierPayOffDroite[#1 + ⅈ λ1, (#2 + ⅈ λ2), K]) &,
          coeff, coeff]]]}, {i, -100, 100}], Joined → True];
  g2 = ListPlot[Table[{i / 100, Module[{x2 = i / 100},
        Re[SecondUnderlyingPayOff[x1, x2, K]]]}, {i, -100, 100}], Joined → True];
  Show[g1, g2, PlotRange → All]]]
```

{13.229, 

}

```
SymetrizedSuperBiHestonUnderlying2VanillaIntegrand[
  K_, τ_, M_, Q_, ρ_, Σ_, Y_, β_, λ1_, λ2_, ω1_, ω2_] :=
 Module[{x1 = Y[[1]], x2 = Y[[2]], k1 = (ω1 + ⅈ λ1), Sk1 = (-ω1 + ⅈ λ1), k2 = (ω2 + ⅈ λ2),
   Sk1A = (-ω1 - ⅈ λ1), k1A = (ω1 - ⅈ λ1), k2A = (ω2 + ⅈ λ2), α, αA, Symα,
   SymαA, α2, αA2, Symα2, SymαA2, propagatorDroit, SympropagatorDroit,
   propagatorGauche, SympropagatorGauche, propagatorDroit2,
   SympropagatorDroit2, propagatorGauche2, SympropagatorGauche2},
  Re[α = ⅇ^(-ⅈ x1 k1-ⅈ x2 k2);
   αA = ⅇ^(-ⅈ x1 k1A-ⅈ x2 k2A);
   Symα = ⅇ^(-ⅈ x1 Sk1-ⅈ x2 k2);
   SymαA = ⅇ^(-ⅈ x1 Sk1A-ⅈ x2 k2A);
   propagatorDroit =
    SuperBiHestonLaplaceTransformReduced[M, Q, ρ, Σ, {-ⅈ k1, -ⅈ k2}, β, τ];
   SympropagatorDroit = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ Sk1, -ⅈ k2}, β, τ];
   propagatorGauche = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ k1A, -ⅈ k2A}, β, τ];
   SympropagatorGauche = SuperBiHestonLaplaceTransformReduced[
     M, Q, ρ, Σ, {-ⅈ Sk1A, -ⅈ k2A}, β, τ];
   α propagatorDroit SecondUnderlyingVanillaFourierPayOffDroite[k1, k2, K] +
    Symα SympropagatorDroit SecondUnderlyingVanillaFourierPayOffDroite[Sk1, k2, K] +
    αA propagatorGauche SecondUnderlyingVanillaFourierPayOffGauche[k1A, k2A, K] +
    SymαA SympropagatorGauche
     SecondUnderlyingVanillaFourierPayOffGauche[Sk1A, k2A, K]]]
```

```
NewSuperBiHestonUnderlying2Vanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonUnderlying2VanillaAux[K, τ, M, Σinf, ρ, Σ, S, β,
   λ1, λ2, {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
   {LegendreCoef2, period2, period2n, ϵ2}, printflag] /; K ≥ 0
```

```
NewSuperBiHestonUnderlying2Vanilla[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 NewSuperBiHestonUnderlying2VanillaAux[K, τ, M, Σinf, ρ, Σ, S, β,
    λ1, λ2, {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag] - K /; K < 0
```

```
NewSuperBiHestonUnderlying2VanillaAux[K_, τ_, M_, Σinf_, ρ_, Σ_, S_, β_,
  λ1_, λ2_, {LegendreCoef1_, LegendreCoef1n_, period1_, period1n_, ϵ1_},
  {LegendreCoef2_, period2_, period2n_, ϵ2_}, printflag_] :=
 2/(2 π)^2 Module[{a, res, res2, Y = {Log[S[[1]]], Log[S[[2]]]}, Q},
   Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
   If[printflag == 1,
    Print["{K,τ,M,Σinf,ρ,Σ,S,β,λ1,λ2}=", {K, τ, M, Σinf, ρ, Σ, S, β, λ1, λ2}]];
   If[printflag == 2, Print[
     "{NbLegendreCoef1,NbLegendreCoef1n,period1,period1n,ϵ1}=",
     {Length[LegendreCoef1], Length[LegendreCoef1n], period1, period1n, ϵ1}]];
   If[printflag == 2, Print["{NbLegendreCoef2,period2,period2n,ϵ2}=",
     {Length[LegendreCoef2], period2, period2n, ϵ2}]];
   If[printflag == 1, Print["{LegendreCoef1,LegendreCoef1n,period1,period1n,ϵ1}=",
     {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
     " {LegendreCoef2,period2,period2n,ϵ2}=",
     {LegendreCoef2, period2, period2n, ϵ2}]];
   res = AdaptativeIntegrate[Function[ω2, a = AdaptativeIntegrate[Function[ω1,
          SymetrizedSuperBiHestonUnderlying2VanillaIntegrand[K, τ, M, Q, ρ, Σ, Y, β,
           λ1, λ2, ω1, ω2]], LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1];
      If[printflag == 2, Print["Integ_2=", a]];
      a[[2]]], LegendreCoef2, period2, period2n, ϵ2];
   If[printflag == 2, Print["Integ_1=", res]];
   res[[2]]]
```

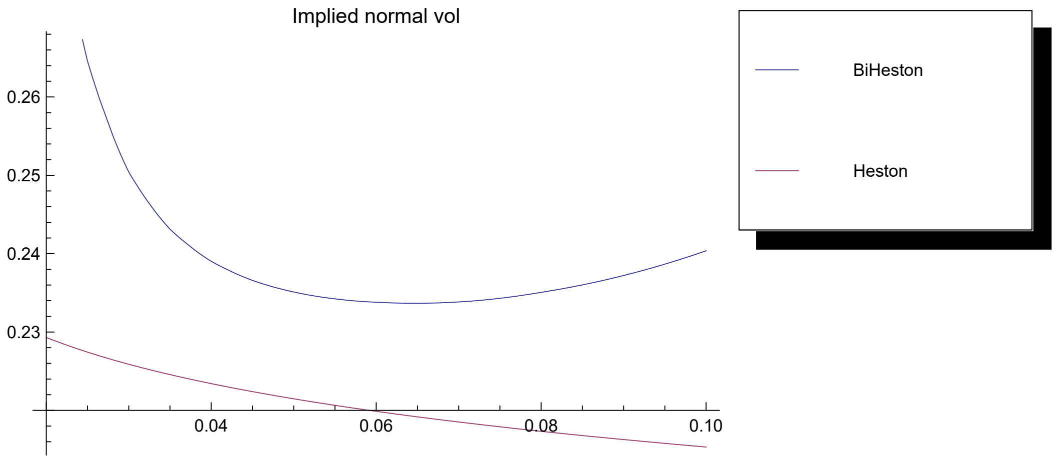
```
Module[{S1 = 0.05, S2 = 0.05, K = 0.05, M1 = -0.01, M2 = -0.02,
  θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5,
  ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1,
  scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n, period1, period1n, ϵ1,
  Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
 scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
 scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
 Nb1 = 12;
 LegendreCoef1 = LegendreCoeffs[Nb1];
 Nb1n = 8;
 LegendreCoef1n = LegendreCoeffs[Nb1n];
 period1 = scope1;
 period1n = scope1;
 ϵ1 = 0.00001;
 Nb2 = 10;
 LegendreCoef2 = LegendreCoeffs[Nb2];
 period2 = scope2;
 period2n = scope2 / 10;
 ϵ2 = 0.00001;
 M = ( M1                 ρm1 Sqrt[M1 M2]
       ρm2 Sqrt[M1 M2]    M2             );
 Σinf = ( θ1                     Sqrt[θ1 θ2] ρsinf
          Sqrt[θ1 θ2] ρsinf      θ2                );
 Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
       Sqrt[Σ1 Σ2] ρs     Σ2             );
 Timing[
  NewSuperBiHestonUnderlying2Vanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
   {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
   {LegendreCoef2, period2, period2n, ϵ2}, printflag]]]

{0.312, 0.0104251}
```

$$\text{scope1} = \frac{2}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};\qquad \text{scope2} = \frac{3}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};$$

$$M = \begin{pmatrix} M1 & \rho m1\sqrt{M1\,M2} \\ \rho m2\sqrt{M1\,M2} & M2 \end{pmatrix};\quad \Sigma inf = \begin{pmatrix} \theta1 & \sqrt{\theta1\,\theta2}\,\rho sinf \\ \sqrt{\theta1\,\theta2}\,\rho sinf & \theta2 \end{pmatrix};\quad \Sigma = \begin{pmatrix} \Sigma1 & \sqrt{\Sigma1\,\Sigma2}\,\rho s \\ \sqrt{\Sigma1\,\Sigma2}\,\rho s & \Sigma2 \end{pmatrix};$$

```
Timing[Module[
  {S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.01, M2 = -0.02, θ1 = 0.03, θ2 = 0.041, ρs = 0.6,
   ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = -0.5, ρ2 = -0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
  M = ( M1              ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]  M2             );
  Σinf = ( θ1                  Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf   θ2                );
  Σ = ( Σ1               Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs   Σ2             );
  strikes = {0.02, 0.025, 0.03, 0.035, 0.04, 0.045,
    0.048, 0.049, 0.0499, 0.05, 0.0501, 0.0505, 0.055, 0.06, 0.08, 0.1};
  smile000 = Table[{strikes[[i]], ImpVolBS[S1, strikes[[i]], τ,
      NewSuperBiHestonUnderlying2Vanilla[strikes[[i]],
       τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
       {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
       {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
  inter000 = Interpolation[smile000, InterpolationOrder → 2];
  smile2 =
   Table[{strikes[[i]], ImpVolHeston2[S2, strikes[[i]], τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]},
    {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2];
  Plot[{inter000[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"BiHeston", "Heston"}, LegendPosition → {1, 0}]
 ]]
```

{8.627,



}

Link with Heston formula

```
LNHestonRiccatiVanillaCallIntegrand[F_, K_, z_, V_, τ_, θ_, λ_, ν_, ρ_] :=
 Module[{ψp, ψm, ξ, X},
  Re[ⅇ^(-Log[F] I z)
    HestonLaplaceTransform4[-λ, θ, ρ, V, -ⅈ z, ν, τ] × FourierPayOffLNSepp[z, K]]]
```

```
LNHestonRiccatiVanillaCall[F_, K_, V0_, τ_, λ_, θ_, ν_, ρ_, limsup_] :=
 F + 1 / Pi NIntegrate[
    LNHestonRiccatiVanillaCallIntegrand[F, K, k1 + I / 2, V0, τ, θ, λ, ν, ρ],
    {k1, 0, limsup}, MaxRecursion → 20]
```

```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
   Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001},
   strikes = {0.01, 0.02, 0.025, 0.03, 0.035, 0.038, 0.04, 0.042, 0.045, 0.05, 0.06,
     0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15}, scope1, scope2, nb = 120,
   integflag = 0, coeffs = LegendreCoeffs[40], K, limsup = 3000, Σinf, Σ, M, Q},
  scope1 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; K = 0.041;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
  {NewSuperBiHestonVanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, S, β, λ1, λ2, scope1, scope2, nb,
    0], LNHestonRiccatiVanillaCall[S[[1]], K + S[[2]], Σ1, τ, -M1, Σ1, θ1, ρ1, limsup]}]]
```

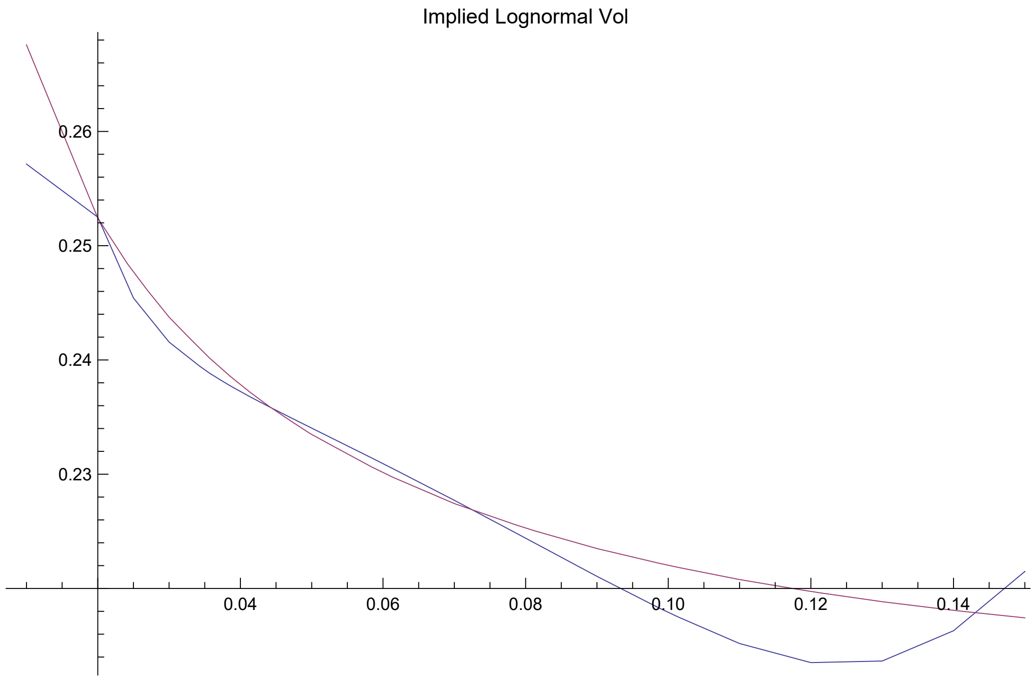
{55.016, {0.00793082, 0.00569173}}

```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
   Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001},
   strikes = {0.01, 0.02, 0.025, 0.03, 0.035, 0.038, 0.04, 0.042, 0.045, 0.05, 0.06,
     0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15}, scope1, scope2, nb = 40,
   integflag = 0, coeffs = LegendreCoeffs[40], K, limsup = 3000, Σinf, Σ, M, Q},
  scope1 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; K = 0.041;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  Q = CholeskyDecomposition[-((M.Σinf + Σinf.Transpose[M])/2)] / β;
  {NewSuperBiHestonVanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, S, β, λ1, λ2, scope1, scope2, nb,
    0], LNHestonRiccatiVanillaCall[S[[1]], K + S[[2]], Σ1, τ, -M1, Σ1, θ1, ρ1, limsup]}]]
```

{5.39, {0.00806349, 0.0743083}}

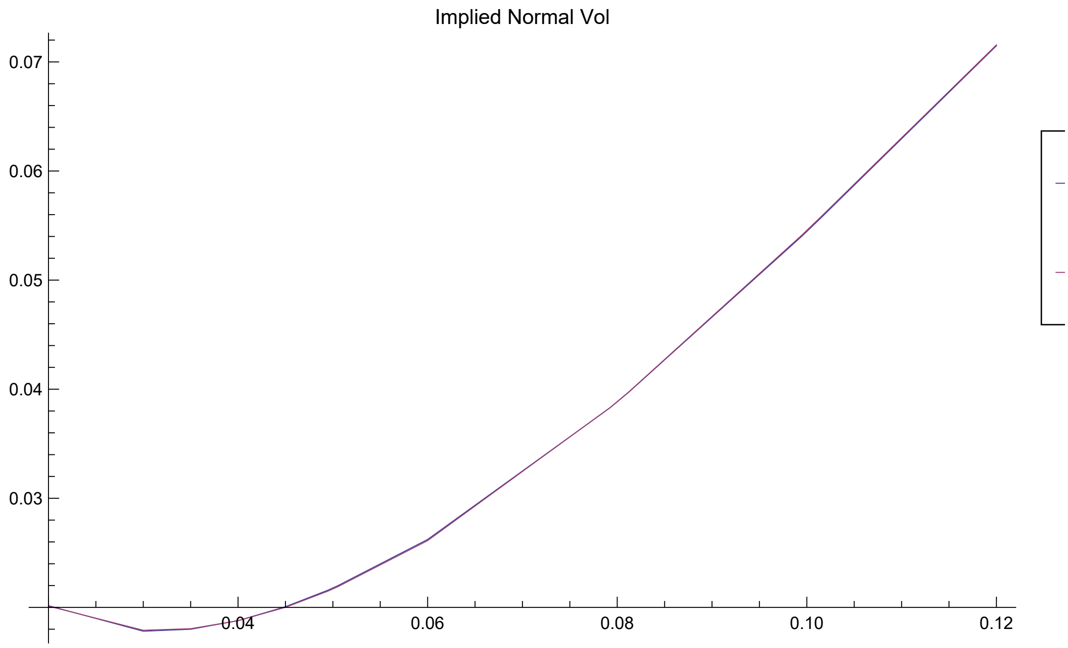
```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
   ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
   Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001}, Q,
   strikes = {0.01, 0.02, 0.025, 0.03, 0.035, 0.038, 0.04, 0.042, 0.045,
     0.05, 0.06, 0.07, 0.08, 0.09, 0.1, 0.11, 0.12, 0.13, 0.14, 0.15}, scope1,
   scope2, nb = 250, integflag = 0, coeffs = LegendreCoeffs[40], M, Σinf, Σ},
  scope1 = 3 / Sqrt[(Σ1 + Σ2 + θ1 + θ2)/4 τ]; scope2 = 5 / Sqrt[(Σ1 + Σ2 + θ1 + θ2)/4 τ];
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2              );
  Σinf = ( θ1                    Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf     θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  smile = Table[{strikes[[i]], ImpVolBS[S[[1]], strikes[[i]] + S[[2]],
       τ, NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ,
        S, β, λ1, λ2, scope1, scope2, nb, 0]]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  inter1 = Interpolation[smile, InterpolationOrder → 2];
  Q = Sqrt[-M1 θ1] / β;
  smile2 = Table[{strikes[[i]], ImpVolBS[S[[1]] - S[[2]], strikes[[i]], τ, HestonCall2[S[[1]],
        strikes[[i]] + S[[2]], τ, Σ1, θ1, ρ1, -M1, Q, coeffs]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  inter21 = Interpolation[smile2, InterpolationOrder → 2];
  {Plot[{inter[x], inter2[x]}, {x, strikes[[1]], Last[strikes]},
    PlotLabel → "Implied Lognormal Vol", PlotLegend → {"BiHeston", "Heston"},
    LegendPosition → {1, 0}, LegendSize → 0.5],
   Plot[{inter1[x], inter21[x]}, {x, strikes[[1]], Last[strikes]},
    PlotLabel → "Implied Lognormal Vol", PlotLegend → {"BiHeston", "Heston"},
    LegendPosition → {1, 0}, LegendSize → 0.5]}
 ]]
```

$\Big\{3476.64, \Big\{$



```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
    ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
    Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001}, Q,
    strikes = {0.02, 0.03, 0.035, 0.04, 0.045, 0.05, 0.06, 0.08, 0.1, 0.12},
    scope1, scope2, nb = 70, inter, inter2, integflag = 0,
    coeffs = LegendreCoeffs[40], M, Σinf, Σ},
  scope1 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  M = ( M1                  ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]     M2              );
  Σinf = ( θ1                    Sqrt[θ1 θ2] ρsinf
           Sqrt[θ1 θ2] ρsinf     θ2                );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρs
        Sqrt[Σ1 Σ2] ρs     Σ2             );
  smile = Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]],
       τ, NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ,
        S, β, λ1, λ2, scope1, scope2, nb, 0]]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  Q = Sqrt[-M1 θ1] / β;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]], τ, HestonCall2[S[[1]],
        strikes[[i]] + S[[2]], τ, Σ1, θ1, ρ1, -M1, Q, coeffs]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  Plot[{inter[x], inter2[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied Normal Vol",
   PlotLegend → {"BiHeston", "Heston"}, LegendPosition → {1, 0}, LegendSize → 0.5]
 ]]
```

{53.586,



Comparaison with Montecarlo Calculations

Test of the Heston Formula

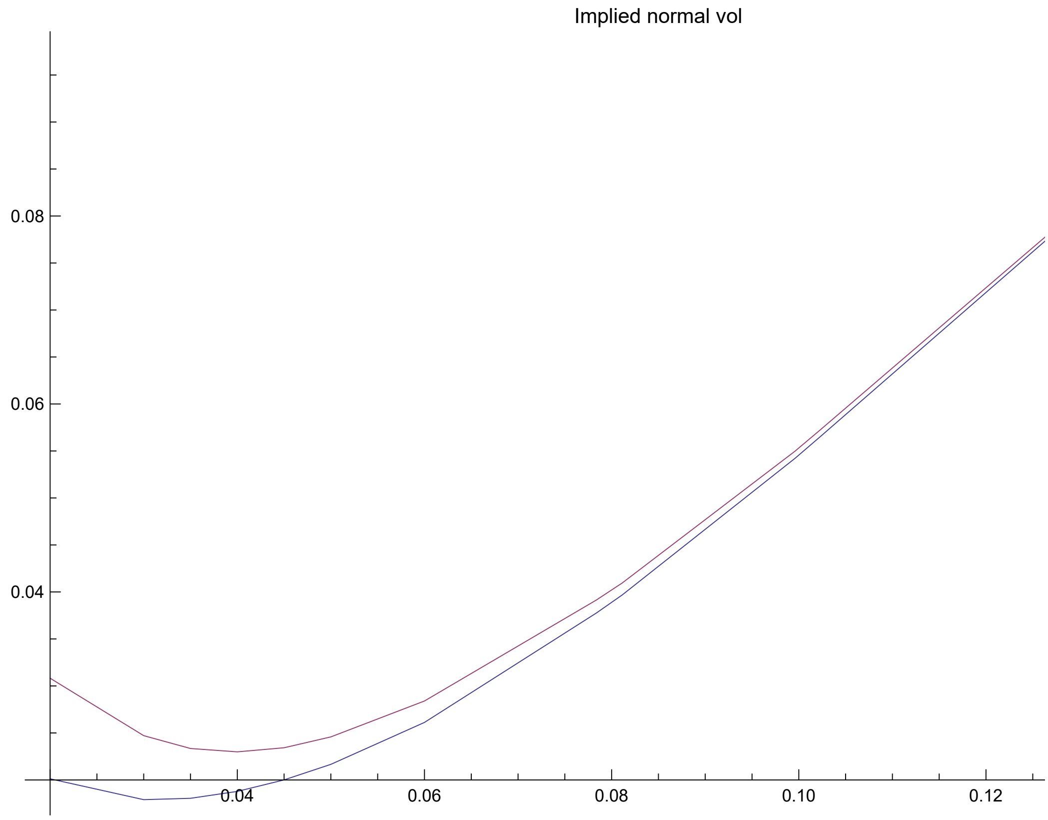
we use a function that computes the all smile in one shot :

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150,
    flag = 1, vol1, vol2, spdopt, StrikeList, M, Q, Σ, M1, M2, ρm1,
    ρm2, TimeStepsNb = 100, nbSample = 100, dt, printflag = 0},
  dt = T/TimeStepsNb;
  StrikeList = {0.02, 0.025, 0.03, 0.0325, 0.035,
    0.037, 0.039, 0.041, 0.043, 0.045, 0.0475, 0.05, 0.055, 0.06};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  HestonMonteCarloSmile[StrikeList, M1,
   Σinf1, ρ1, Σ1, S1, β, TimeStepsNb, dt, nbSample, printflag]]]
```

$$dt = \frac{T}{TimeStepsNb};\quad scope1 = \frac{z1max}{\sqrt{\frac{\Sigma1+\Sigma2+\Sigma inf1+\Sigma inf2}{4}\,\tau}};\quad scope2 = \frac{z2max}{\sqrt{\frac{\Sigma1+\Sigma2+\Sigma inf1+\Sigma inf2}{4}\,\tau}};\quad M = \begin{pmatrix} M1 & \rho m1\,\sqrt{M1\,M2} \\ \rho m2\,\sqrt{M1\,M2} & M2 \end{pmatrix}$$

```
β=2.25
{7.062, {0.0245178, 0.0200306, 0.0160621, 0.0142936, 0.0127151, 0.0115819, 0.0105933,
  0.00970611, 0.00893601, 0.00820456, 0.00739942, 0.00666619, 0.0054356, 0.00441009}}
```

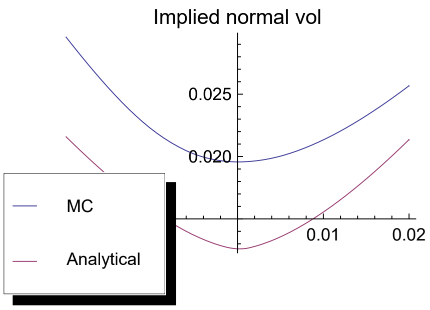
```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
    ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
    Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001}, Q,
    strikes = {0.02, 0.03, 0.035, 0.04, 0.045, 0.05, 0.06, 0.08, 0.1, 0.15},
    scope1, scope2, nb = 70, inter, inter2, integflag = 0,
    coeffs = LegendreCoeffs[40], dt, TimeStepsNb = 200, nbSample = 5000},
  scope1 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; dt = τ / TimeStepsNb;
  Q = Sqrt[-M1 θ1] / β;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]], τ, HestonCall2[S[[1]],
        strikes[[i]] + S[[2]], τ, Σ1, θ1, ρ1, -M1, Q, coeffs]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  pricelist = HestonMonteCarloSmile[strikes,
     M1, θ1, ρ1, Σ1, S[[1]], β, TimeStepsNb, dt, nbSample, 0];
  smile3 = Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]],
       strikes[[i]], τ, pricelist[[i]]]}, {i, 1, Length[strikes]}];
  inter3 = Interpolation[smile3, InterpolationOrder → 1];

  Plot[{inter2[x], inter3[x]},
   {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
   PlotLegend → {"Heston", "MC"}, LegendPosition → {1, 0}, LegendSize → 0.5]
 ]]
```

{219.915, 

Test of the Bi Heston Formula

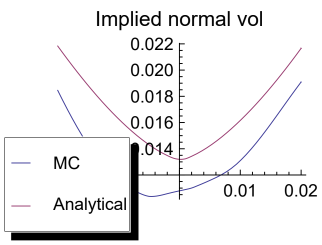
```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 120, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 10000, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList = {-0.02, -0.015, -0.01, -0.007, -0.005, -0.003,
    -0.002, -0.001, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 3; z2max = 5;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2              );
  ΣinfM = ( Σinf1                          Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12       Σinf2                    );
  Σ = ( Σ1                   Sqrt[Σ1 Σ2] ρ12
        Sqrt[Σ1 Σ2] ρ12      Σ2              );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  pricelist = BiHestonMonteCarloSmile[StrikeList, M, Q, ΣinfM, {ρ1, ρ2},
    {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  smile = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2,
       StrikeList[[i]], T, pricelist[[i]]]}, {i, 1, Length[StrikeList]}];
  inter = Interpolation[smile];
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β,
        λ1, λ2, scope1, scope2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  Plot[{inter[x], inter2[x]}, {x, StrikeList[[1]], Last[StrikeList]},
   PlotLabel → "Implied normal vol", PlotLegend → {"MC", "Analytical"}]
 ]]
```

β=2.25

{577.625,  }

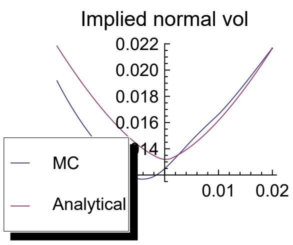
```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
    vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
    TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T / TimeStepsNb;
  StrikeList = {-0.02, -0.015, -0.01, -0.007, -0.005, -0.003,
    -0.002, -0.001, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1              ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]  M2             );
  ΣinfM = ( Σinf1                         Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12      Σinf2                    );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρ12
        Sqrt[Σ1 Σ2] ρ12    Σ2              );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M]) / 2)] / β;
  pricelist = BiHestonMonteCarloSmile[StrikeList, M, Q, ΣinfM, {ρ1, ρ2},
    {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  smile = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2,
      StrikeList[[i]], T, pricelist[[i]]]}, {i, 1, Length[StrikeList]}];
  inter = Interpolation[smile];
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
      NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β,
       λ1, λ2, scope1, scope2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  Plot[{inter[x], inter2[x]}, {x, StrikeList[[1]], Last[StrikeList]},
   PlotLabel → "Implied normal vol", PlotLegend → {"MC", "Analytical"}]
 ]]
```

β=2.25

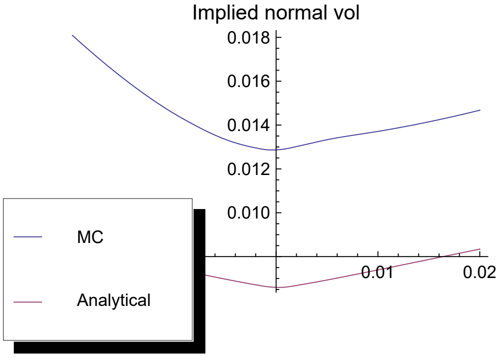
{29.11,  }

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
    vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
    TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T / TimeStepsNb;
  StrikeList = {-0.02, -0.015, -0.01, -0.007, -0.005, -0.003,
    -0.002, -0.001, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1             ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]  M2            );
  ΣinfM = ( Σinf1                       Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12    Σinf2                    );
  Σ = ( Σ1                 Sqrt[Σ1 Σ2] ρ12
        Sqrt[Σ1 Σ2] ρ12    Σ2              );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  pricelist = BiHestonMonteCarloSmile[StrikeList, M, Q, ΣinfM, {ρ1, ρ2},
    {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  smile = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2,
      StrikeList[[i]], T, pricelist[[i]]]}, {i, 1, Length[StrikeList]}];
  inter = Interpolation[smile];
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
      NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β,
       λ1, λ2, scope1, scope2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  Plot[{inter[x], inter2[x]}, {x, StrikeList[[1]], Last[StrikeList]},
   PlotLabel → "Implied normal vol", PlotLegend → {"MC", "Analytical"}]
 ]]
```

β=2.25

{29.141,  }

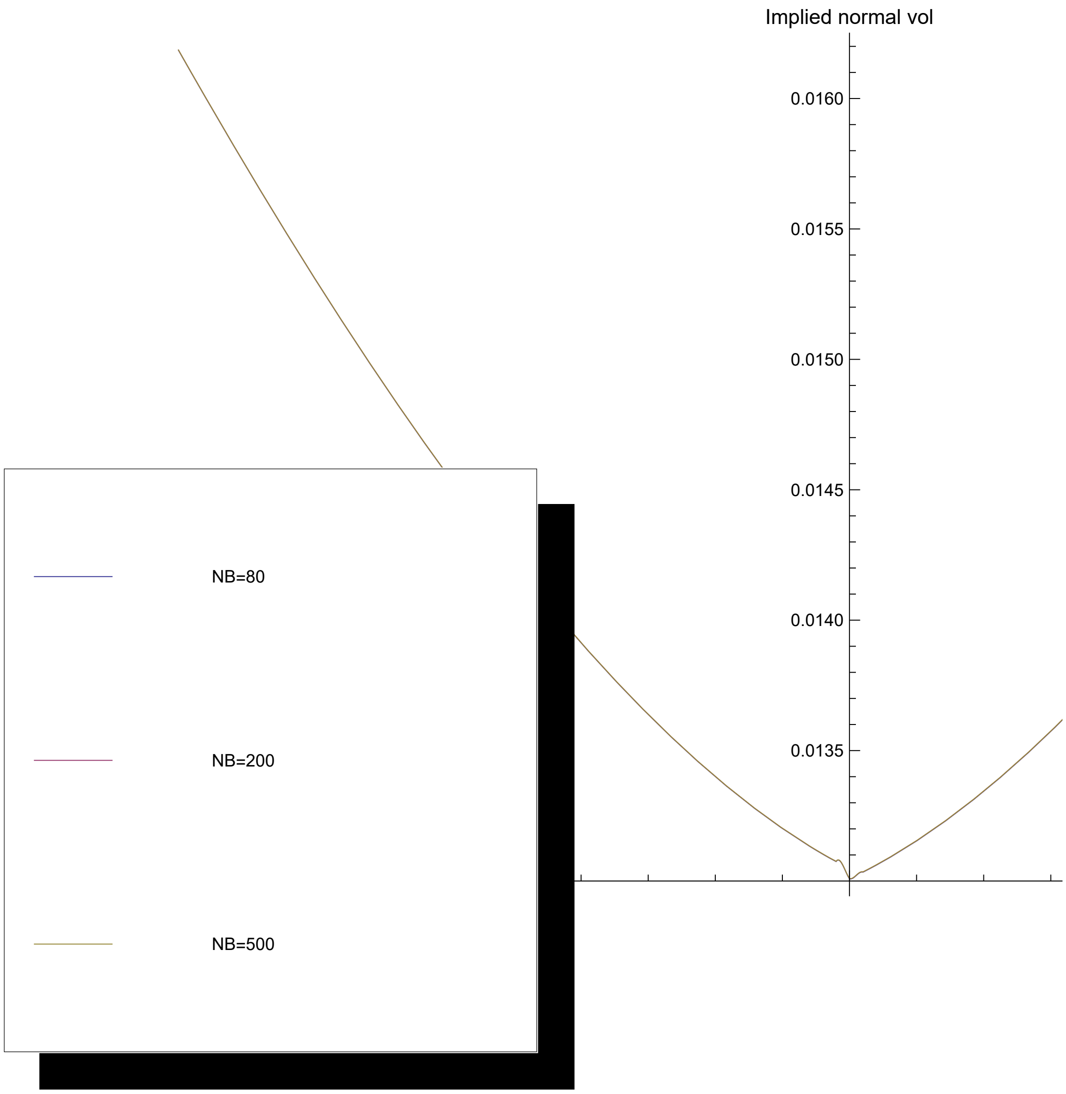
```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList = {-0.02, -0.015, -0.01, -0.007, -0.005, -0.003,
    -0.002, -0.001, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρ12}, {Sqrt[Σ1 Σ2] ρ12, Σ2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  pricelist = BiHestonMonteCarloSmile[StrikeList, M, Q, ΣinfM, {ρ1, ρ2},
    {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  smile = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2,
      StrikeList[[i]], T, pricelist[[i]]]}, {i, 1, Length[StrikeList]}];
  inter = Interpolation[smile];
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
      NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β,
       λ1, λ2, scope1, scope2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  Plot[{inter[x], inter2[x]}, {x, StrikeList[[1]], Last[StrikeList]},
   PlotLabel → "Implied normal vol", PlotLegend → {"MC", "Analytical"}]
 ]]
β=2.25
```

{81.375,  }
Implied normal vol
0.018
0.016
0.014
0.012
0.010
0.01
0.02
MC
Analytical

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb1 = 80; Nb2 = 200; Nb3 = 500;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρ12}, {Sqrt[Σ1 Σ2] ρ12, Σ2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM. Transpose[M])/2)] / β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1, scope2, Nb1, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1, scope2, Nb2, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1, scope2, Nb3, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"NB=" <> ToString[Nb1], "NB=" <> ToString[Nb2], "NB=" <> ToString[Nb3]}]
 ]]
```

β=2.25

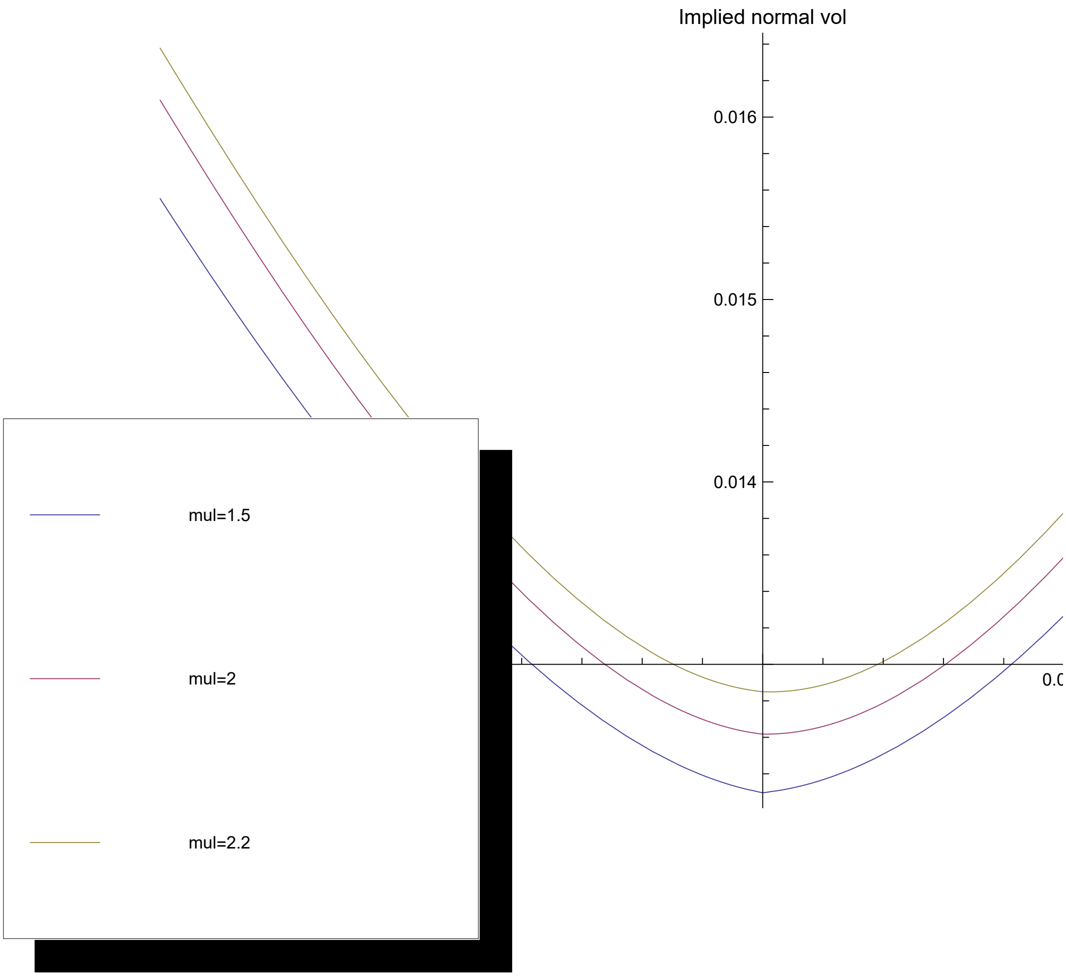
{7969.55,


Implied normal vol
0.0160
0.0155
0.0150
0.0145
0.0140
0.0135
NB=80
NB=200
NB=500

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb = 60; mul1 = 1.5; mul2 = 2; mul3 = 2.2;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρ12}, {Sqrt[Σ1 Σ2] ρ12, Σ2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)]/β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul1, scope2 mul1, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul2, scope2 mul2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul3, scope2 mul3, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"mul=" <> ToString[mul1], "mul=" <> ToString[mul2], "mul=" <> ToString[mul3]}]
 ]]
```

β=2.25

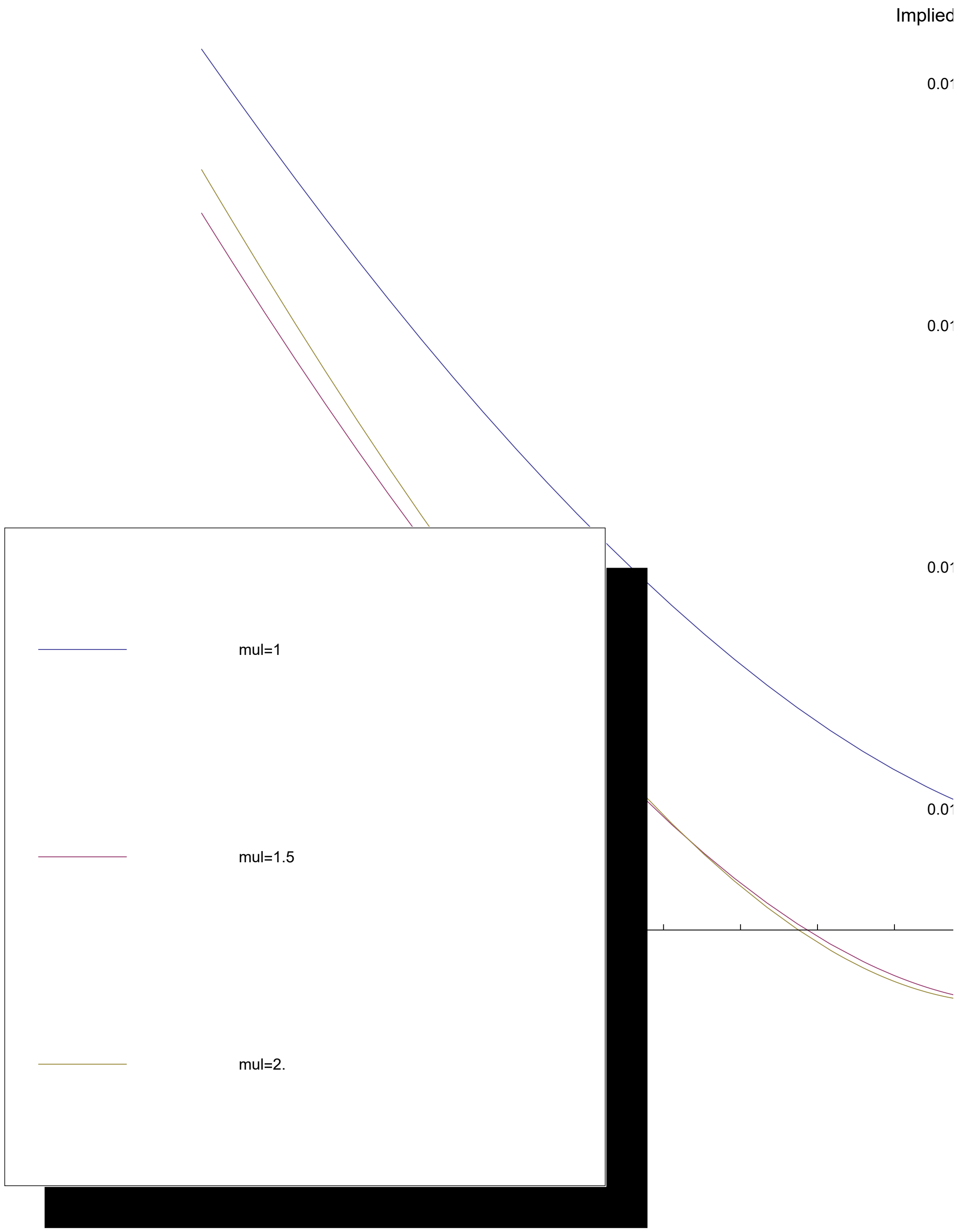
{391.843,



```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb, flag = 1,
    vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
    TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb = 80; mul1 = 1; mul2 = 1.5; mul3 = 2.;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]},
       {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12},
           {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρ12},
       {Sqrt[Σ1 Σ2] ρ12, Σ2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul1, scope2 mul1, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul2, scope2 mul2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul3, scope2 mul3, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"mul=" <> ToString[mul1], "mul=" <> ToString[mul2], "mul=" <> ToString[mul3]}]
 ]]
```

β=2.25

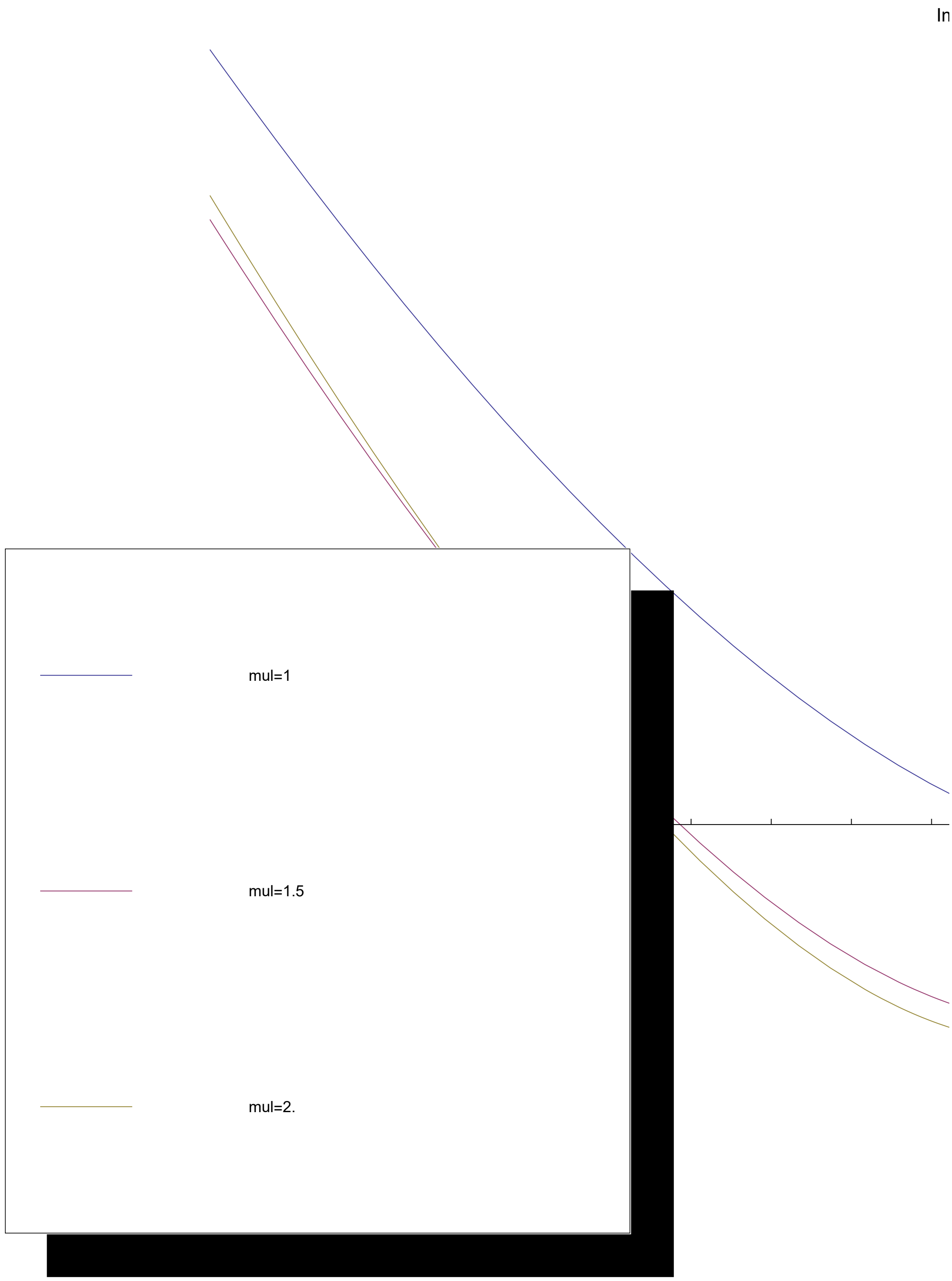
{686.825,



```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T / TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb = 120; mul1 = 1; mul2 = 1.5; mul3 = 2.;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρ12}, {Sqrt[Σ1 Σ2] ρ12, Σ2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul1, scope2 mul1, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul2, scope2 mul2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla2[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul3, scope2 mul3, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"mul=" <> ToString[mul1], "mul=" <> ToString[mul2], "mul=" <> ToString[mul3]}]
 ]]
```

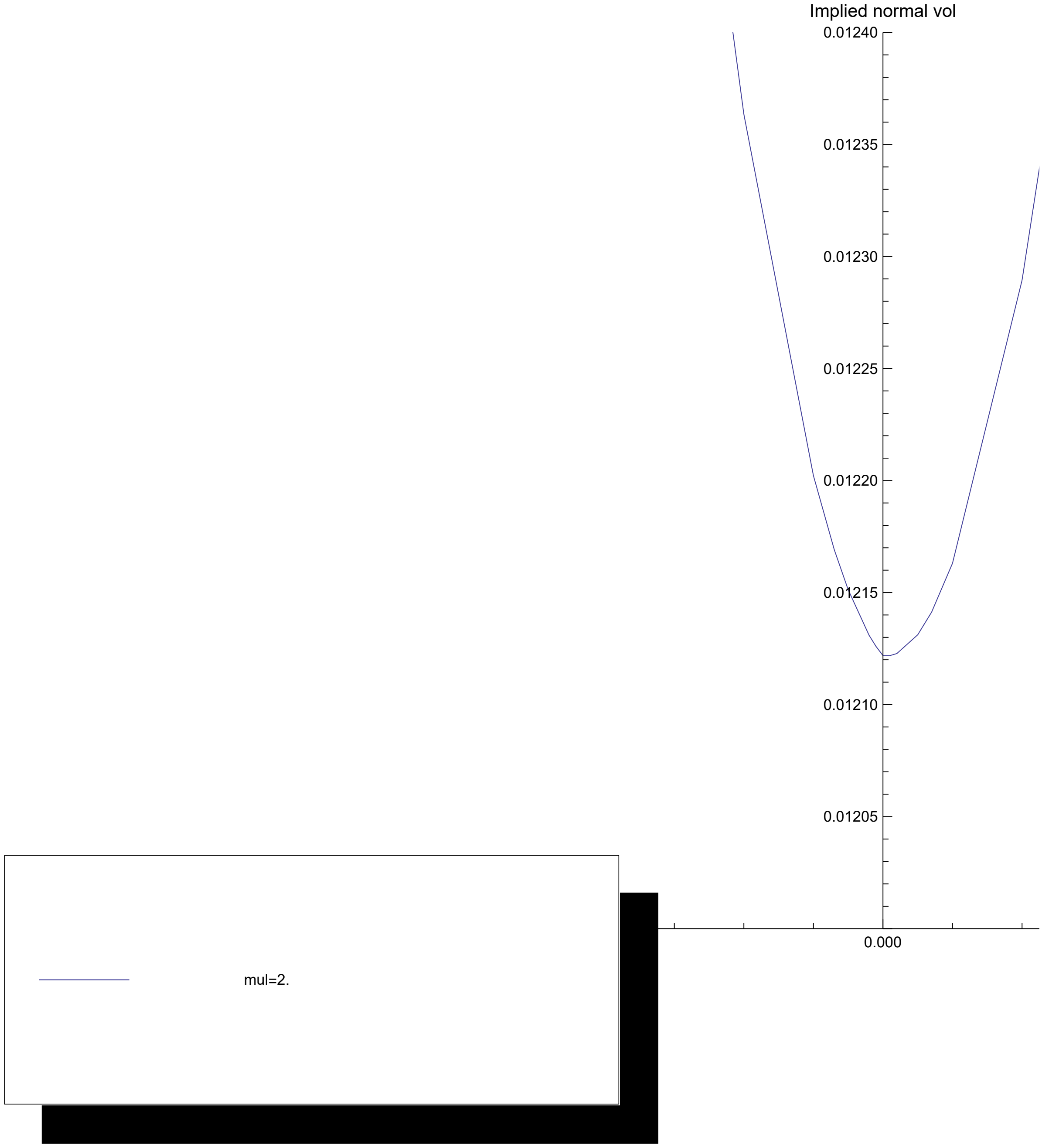
β=2.25

{1527.86,




```
inter4 = Interpolation[smile4, InterpolationOrder → 1];
```

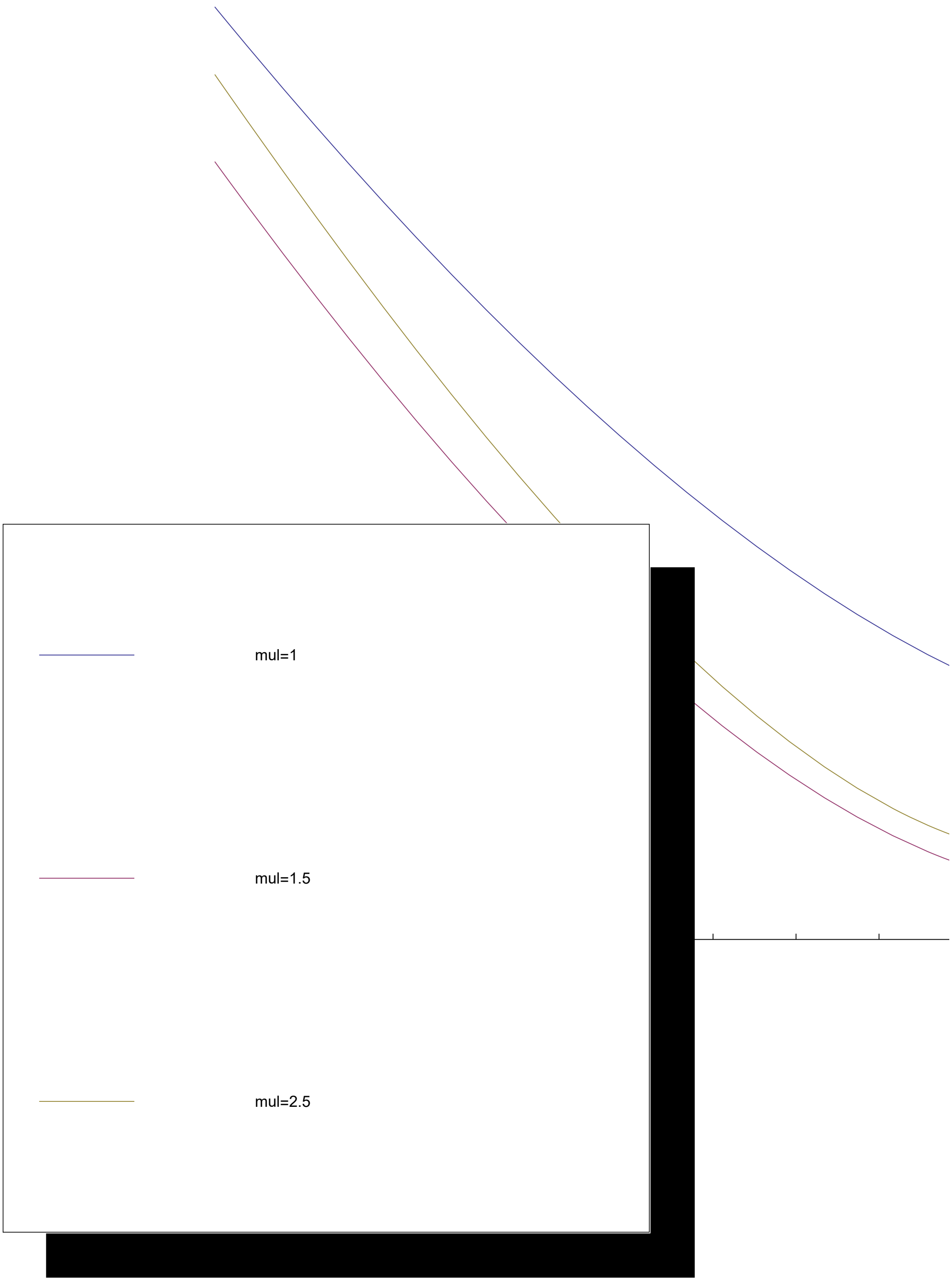
```
Plot[{inter4[x]}, {x, StrikeList⟦1⟧, Last[StrikeList]},
 PlotLabel → "Implied normal vol",
 PlotLegend → {"mul=" <> ToString[mul3]}, PlotRange → {0.012, 0.0124}]
```



```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb = 120; mul1 = 1; mul2 = 1.5; mul3 = 2.5;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1               ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]  M2              );
  ΣinfM = ( Σinf1                         Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12      Σinf2                    );
  Σ = ( Σ1                  Sqrt[Σ1 Σ2] ρ12
        Sqrt[Σ1 Σ2] ρ12     Σ2              );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul1, scope2 mul1, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul2, scope2 mul2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul3, scope2 mul3, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"mul=" <> ToString[mul1], "mul=" <> ToString[mul2], "mul=" <> ToString[mul3]}]
 ]]
```

β=2.25

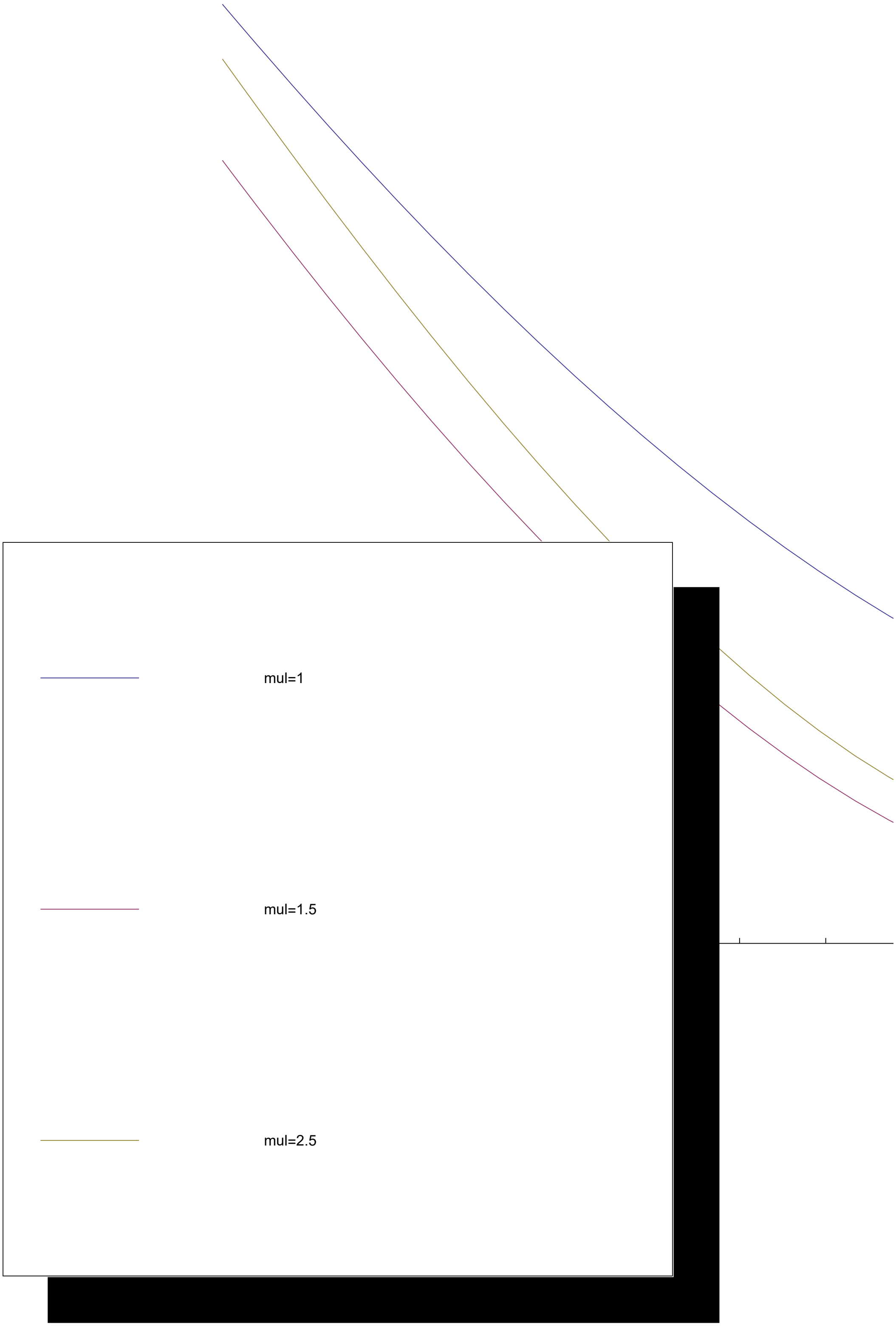
{1271.81,



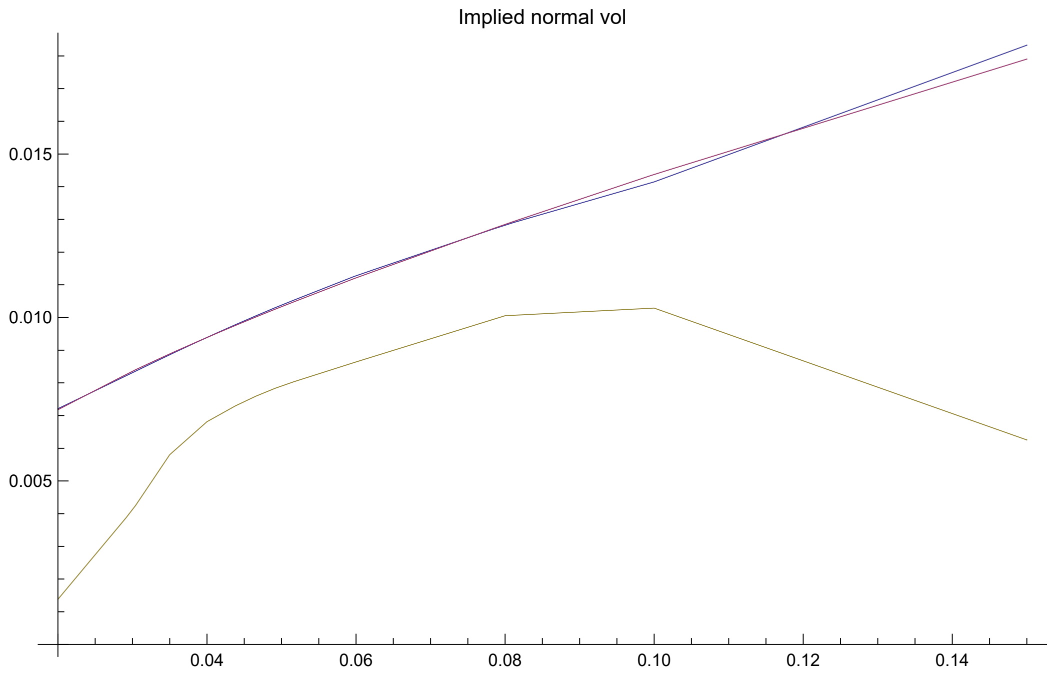
```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 40, flag = 1,
   vol1, vol2, spdopt, inter, inter2, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 40, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList =
   {-0.01, -0.007, -0.005, -0.003, -0.002, -0.001, -0.0007, -0.0005, -0.0002, -0.0001,
    0, 0.0001, 0.0002, 0.0005, 0.0007, 0.001, 0.002, 0.003, 0.005, 0.007, 0.01};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 2.5; z2max = 4.5; Nb = 500; mul1 = 1; mul2 = 1.5; mul3 = 2.5;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1                 ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]    M2              );
  ΣinfM = ( Σinf1                         Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12      Σinf2                    );
  Σ = ( Σ1                    Sqrt[Σ1 Σ2] ρ12
        Sqrt[Σ1 Σ2] ρ12       Σ2              );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  smile2 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul1, scope2 mul1, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter2 = Interpolation[smile2];
  smile3 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul2, scope2 mul2, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter3 = Interpolation[smile3];
  smile4 = Table[{StrikeList[[i]], NormalImplicitVol[S1 - S2, StrikeList[[i]], T,
       NewSuperBiHestonVanilla[StrikeList[[i]], T, M, ΣinfM, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1,
        λ2, scope1 mul3, scope2 mul3, Nb, printflag]]}, {i, 1, Length[StrikeList]}];
  inter4 = Interpolation[smile4];
  Plot[{inter2[x], inter3[x], inter4[x]},
   {x, StrikeList[[1]], Last[StrikeList]}, PlotLabel → "Implied normal vol", PlotLegend →
    {"mul=" <> ToString[mul1], "mul=" <> ToString[mul2], "mul=" <> ToString[mul3]}]
 ]]
```

β=2.25

{20 549.9,



```
Timing[Module[{M1 = -0.075, M2 = -0.075, θ1 = 0.15, θ2 = 0.15,
    ρs = 0., ρsinf = 0., ρm1 = 0., ρm2 = 0., ρ1 = -0.5, ρ2 = -0.5, Σ1 = 0.04,
    Σ2 = 0.04, β = 2.25, τ = 5, λ1 = 1.1, λ2 = 1.1, ν1, S = {0.04, 0.0001}, Q,
    strikes = {0.02, 0.03, 0.035, 0.04, 0.045, 0.05, 0.06, 0.08, 0.1, 0.15},
    scope1, scope2, nb = 70, inter, inter2, integflag = 0,
    coeffs = LegendreCoeffs[40], dt, TimeStepsNb = 100, nbSample = 400, M, Σinf, Σ},
  scope1 = 3/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; scope2 = 5/Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ]; dt = τ/TimeStepsNb;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  smile = Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]],
       strikes[[i]], τ, SuperBiHestonCall[strikes[[i]], τ, M, Σinf, {ρ1, ρ2},
        Σ, S, β, λ1, λ2, scope1, scope2, nb, 0]]}, {i, 1, Length[strikes]}];
  inter = Interpolation[smile, InterpolationOrder → 1];
  Q = Sqrt[-M1 θ1] / β;
  smile2 =
   Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]], strikes[[i]], τ, HestonCall2[S[[1]],
        strikes[[i]] + S[[2]], τ, Σ1, θ1, ρ1, -M1, Q, coeffs]]}, {i, 1, Length[strikes]}];
  inter2 = Interpolation[smile2, InterpolationOrder → 1];
  pricelist = BiHestonMonteCarloSmile[strikes,
    M, Σinf, {ρ1, ρ2}, Σ, S, β, TimeStepsNb, dt, nbSample, 0];
  smile3 = Table[{strikes[[i]], NormalImplicitVol[S[[1]] - S[[2]],
       strikes[[i]], τ, pricelist[[i]]]}, {i, 1, Length[strikes]}];
  inter3 = Interpolation[smile3, InterpolationOrder → 1];

  Plot[{inter[x], inter2[x], inter3[x]}, {x, strikes[[1]], Last[strikes]},
   PlotLabel → "Implied normal vol", PlotLegend → {"BiHeston", "Heston", "MC"},
   LegendPosition → {1, 0}, LegendSize → 0.5]
 ]]
```

{250.078, 

Appendices

Appendix 1

Computation of the Fourier transform of the payoff

Case K > 0

```
integration1 = Simplify[∫_{Log[e^{x2}+K]}^{∞} e^{i k1 x1} (α e^{x1} - β e^{x2} - K) dx1]
```

$$\text{If}\left[\text{Im}[k1] > 1,\ \frac{\left(e^{x2}+K\right)^{i\,k1}\left(K\,(i + k1\,(-1+\alpha)) + e^{x2}\,(k1\,(\alpha-\beta) + i\,\beta)\right)}{(-1 - i\,k1)\,k1},\ \text{Integrate}\left[-e^{i\,k1\,x1}\,K + e^{x1+i\,k1\,x1}\,\alpha - e^{i\,k1\,x1+x2}\,\beta,\ \{x1, \text{Log}[e^{x2}+K], \infty\},\ \text{Assumptions} \to \text{Im}[k1] \le 1\right]\right]$$

```
integration11 = Simplify[integration1, Im[k1] > 1]
```

$$\frac{\left(e^{x2}+K\right)^{i\,k1}\left(K\,(i + k1\,(-1+\alpha)) + e^{x2}\,(k1\,(\alpha-\beta) + i\,\beta)\right)}{(-1 - i\,k1)\,k1}$$

$$\texttt{Simplify}\left[\texttt{Integrate}\left[e^{i\,k2\,x2}\ \texttt{integration11},\ \texttt{x2}\right],\ K>0\right]$$

$$\frac{1}{(-1-i\,k1)\,k1\,k2\,(-i+k2)}\,e^{i\,k2\,x2}\,K^{i\,k1}\left(e^{x2}\,k2\,(-i\,k1\,(\alpha-\beta)+\beta)\ \texttt{Hypergeometric2F1}\left[-i\,k1,\ 1+i\,k2,\ 2+i\,k2,\ -\frac{e^{x2}}{K}\right]+K\,(-i+k2)\,(1-i\,k1\,(-1+\alpha))\ \texttt{Hypergeometric2F1}\left[-i\,k1,\ i\,k2,\ 1+i\,k2,\ -\frac{e^{x2}}{K}\right]\right)$$

ca converge pour Im[k1] > 1, Im[k2] > 0 pour x2 ∈ [G, +∞[

Case K = 0

$$\texttt{integration2}=\texttt{Simplify}\left[\int_{x2}^{\infty} e^{i\,k1\,x1}\left(\alpha\,e^{x1}-\beta\,e^{x2}\right)dx1\right]$$

$$\texttt{If}\left[\texttt{Im}[k1]>1,\ \frac{e^{x2+i\,k1\,x2}\,(k1\,(\alpha-\beta)+i\,\beta)}{(-1-i\,k1)\,k1},\ \texttt{Integrate}\left[e^{x1+i\,k1\,x1}\,\alpha-e^{i\,k1\,x1+x2}\,\beta,\ \{x1,\ x2,\ \infty\},\ \texttt{Assumptions}\to\texttt{Im}[k1]\le 1\right]\right]$$

$$\texttt{integration22}=\texttt{Simplify}[\texttt{integration2},\ \texttt{Im}[k1]>1]$$

$$\frac{e^{x2+i\,k1\,x2}\,(k1\,(\alpha-\beta)+i\,\beta)}{(-1-i\,k1)\,k1}$$

$$\texttt{Simplify}\left[\texttt{Integrate}\left[e^{i\,k2\,x2}\ \texttt{integration22},\ \texttt{x2}\right],\ K>0\right]$$

$$\frac{e^{i\,(-i+k1+k2)\,x2}\,(k1\,(\alpha-\beta)+i\,\beta)}{k1\,(-i+k1)\,(-i+k1+k2)}$$

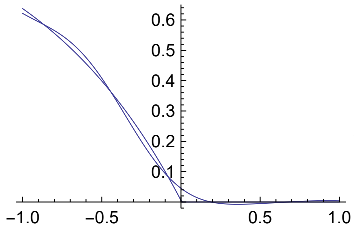
We can check that it is indeed the limit of $K\to 0^{+}$

$$\texttt{Simplify}\left[\texttt{Integrate}\left[e^{i k2\,x2}\ (\texttt{integration11 /. } K\to 0),\ \texttt{x2}\right],\ x2>0\right]$$

$$\frac{e^{x2\,(1+i\,k1+i k2)}\,(k1\,(\alpha-\beta)+i\,\beta)}{k1\,(-i+k1)\,(k1-i\,(1+i k2))}$$

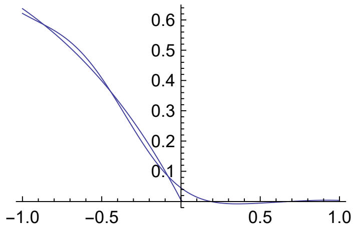
Check of the Fourier Transform numerically K>0

```
Timing[Module[{x1 = 0.1, K = 0.1, λ = 2, λ2 = 1.2, G = -10},
  g1 = ListPlot[Table[{i / 100, Re[Module[{x2 = i / 100}, 1/(2 π)^2 CoeffBasedIntegrate[
        (e^(-i x1 (#1+i λ)-i x2 (#2-i λ2)) CompleteFourierPayOffGauche[#1 + i λ, (#2 - i λ2), K] +
           e^(-i x1 (#1+i λ)-i x2 (#2+i λ2)) CompleteFourierPayOffDroite[#1 +
              i λ, (#2 + i λ2), K]) &, RiemanCoeffs[40, -8, 8],
        RiemanCoeffs[40, -8, 8]]]]}, {i, -100, 100}], Joined → True];
  g2 = ListPlot[Table[{i / 100, Module[{x2 = i / 100}, Re[PayOff[x1, x2, K]]]},
     {i, -100, 100}], Joined → True];
  Show[g1, g2, PlotRange → All]]]
```

{611.187,  }

Check of the Fourier Transform numerically K==0

```
Timing[Module[{x1 = 0.1, λ = 2, λ2 = 1.2, G = -10},
  g1 = ListPlot[Table[{i / 100, Re[Module[{x2 = i / 100}, 1/(2 π)^2 CoeffBasedIntegrate[
        (e^(-i x1 (#1+i λ)-i x2 (#2-i λ2)) CompleteFourierPayOffGauche[#1 + i λ, (#2 - i λ2)] +
           e^(-i x1 (#1+i λ)-i x2 (#2+i λ2)) CompleteFourierPayOffDroite[#1 +
              i λ, (#2 + i λ2)]) &, RiemanCoeffs[40, -8, 8],
        RiemanCoeffs[40, -8, 8]]]]}, {i, -100, 100}], Joined → True];
  g2 = ListPlot[Table[{i / 100, Module[{x2 = i / 100}, Re[PayOff[x1, x2]]]},
     {i, -100, 100}], Joined → True];
  Show[g1, g2, PlotRange → All]]]
```

{611.187,  }

Appendix 2

Computation of the β optimal

Mais il faut que les valeur propres de M soient negatives, c' est equivalent a une trace negative et un determinant positif, c' est a dire

$$\mathbf{Det}\left[\begin{pmatrix} \frac{-\lambda_1}{2} & \frac{\lambda_2\,\Sigma_{\infty 2}-\beta\left(\frac{\nu_2^2}{4}\right)}{2\,\Sigma_{12}} \\ \frac{\lambda_1\,\Sigma_{\infty 1}-\beta\left(\frac{\nu_1^2}{4}\right)}{2\,\Sigma_{12}} & \frac{-\lambda_2}{2}\end{pmatrix}\right] > 0$$

Soit encore

$$\mathbf{Simplify}\left[\mathbf{Det}\left[\begin{pmatrix} \frac{-\lambda_1}{2} & \frac{\lambda_2\,\Sigma_{\infty 2}-\beta\left(\frac{\nu_2^2}{4}\right)}{2\,\Sigma_{12}} \\ \frac{\lambda_1\,\Sigma_{\infty 1}-\beta\left(\frac{\nu_1^2}{4}\right)}{2\,\Sigma_{12}} & \frac{-\lambda_2}{2}\end{pmatrix}\right]\right]$$

$$\frac{\beta\,\nu_1^2\left(-\beta\,\nu_2^2+4\,\lambda_2\,\Sigma_{\infty 2}\right)+4\,\lambda_1\left(\beta\,\nu_2^2\,\Sigma_{\infty 1}+4\,\lambda_2\left(\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)\right)}{64\,\Sigma_{12}^2}$$

la question est quant pour $\beta >$ 1 on peut avoir $\beta\,\nu_1^2\left(-\beta\,\nu_2^2+4\,\lambda_2\,\Sigma_{\infty 2}\right)+4\,\lambda_1\left(\beta\,\nu_2^2\,\Sigma_{\infty 1}+4\,\lambda_2\left(\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)\right) > 0$

$$\mathbf{Collect}\left[\beta\,\nu_1^2\left(-\beta\,\nu_2^2+4\,\lambda_2\,\Sigma_{\infty 2}\right)+4\,\lambda_1\left(\beta\,\nu_2^2\,\Sigma_{\infty 1}+4\,\lambda_2\left(\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)\right),\ \beta\right]$$

Sum::sumwarn : Warning: Σ_{12}^2 contains a capital sigma; use ESCsumESC to enter a summation sign. ≫

$$-\beta^2\,\nu_1^2\,\nu_2^2+\beta\left(4\,\lambda_1\,\nu_2^2\,\Sigma_{\infty 1}+4\,\lambda_2\,\nu_1^2\,\Sigma_{\infty 2}\right)+16\,\lambda_1\,\lambda_2\left(\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)$$

$$\mathbf{F}[\beta\_] := -\beta^2\,\nu_1^2\,\nu_2^2+\beta\left(4\,\lambda_1\,\nu_2^2\,\Sigma_{\infty 1}+4\,\lambda_2\,\nu_1^2\,\Sigma_{\infty 2}\right)+16\,\lambda_1\,\lambda_2\left(\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)$$

Sum::sumwarn : Warning: Σ_{12}^2 contains a capital sigma; use ESCsumESC to enter a summation sign. ≫

$$\mathbf{Solve}[\mathbf{D}[\mathbf{F}[\beta],\ \beta] == 0,\ \beta]$$

$$\left\{\left\{\beta \to \frac{2\left(\lambda_1\,\nu_2^2\,\Sigma_{\infty 1}+\lambda_2\,\nu_1^2\,\Sigma_{\infty 2}\right)}{\nu_1^2\,\nu_2^2}\right\}\right\}$$

let define $\beta_1 = \frac{2\,\lambda_1\,\Sigma_{\infty 1}}{\nu_1^2}$ and $\beta_2 = \frac{2\,\lambda_2\,\Sigma_{\infty 2}}{\nu_2^2}$ on naturellement $\beta_1 > 1$ et $\beta_2 > 1$

et

$\beta_{\mathrm{optimal}} = \beta_1 + \beta_2$

$$\mathbf{Simplify}[\mathbf{F}[\beta]\ /.\ \mathbf{Solve}[\mathbf{D}[\mathbf{F}[\beta],\ \beta] == 0,\ \beta]]$$

$$\left\{4\left(\frac{\lambda_1^2\,\nu_2^2\,\Sigma_{\infty 1}^2}{\nu_1^2}+\frac{\lambda_2^2\,\nu_1^2\,\Sigma_{\infty 2}^2}{\nu_2^2}+2\,\lambda_1\,\lambda_2\left(2\,\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)\right)\right\}$$

$$\mathbf{Simplify}\left[\frac{\lambda_1^2\,\nu_2^2\,\Sigma_{\infty 1}^2}{\nu_1^2}+\frac{\lambda_2^2\,\nu_1^2\,\Sigma_{\infty 2}^2}{\nu_2^2}\ /.\ \left\{\Sigma_{\infty 1} \to \frac{\beta_1\,\nu_1^2}{2\,\lambda_1},\ \Sigma_{\infty 2} \to \frac{\beta_2\,\nu_2^2}{2\,\lambda_2}\right\}\right]$$

Sum::sumwarn : Warning: $\Sigma_{\infty 1}^2$ contains a capital sigma; use ESCsumESC to enter a summation sign. ≫

Sum::sumwarn : Warning: $\Sigma_{\infty 2}^2$ contains a capital sigma; use ESCsumESC to enter a summation sign. ≫

$$\frac{1}{4}\left(\beta_1^2+\beta_2^2\right)\nu_1^2\,\nu_2^2$$

$$\mathrm{F}\left[\beta_{\mathrm{optimal}}\right] = 4\left(\frac{\lambda_1^2\,\nu_2^2\,\Sigma_{\infty 1}^2}{\nu_1^2}+\frac{\lambda_2^2\,\nu_1^2\,\Sigma_{\infty 2}^2}{\nu_2^2}+2\,\lambda_1\,\lambda_2\left(2\,\Sigma_{12}^2-\Sigma_{\infty 1}\,\Sigma_{\infty 2}\right)\right) =$$

$$\frac{1}{4}\left(\beta_1^2+\beta_2^2-2\,\beta_1\,\beta_2\right)\,\nu_1^2\,\nu_2^2+16\,\lambda_1\,\lambda_2\,\Sigma_{12}^2=$$

$$\frac{1}{4}\,(\beta_1+\beta_2)^2+16\,\lambda_1\,\lambda_2\,\Sigma_{12}^2=\frac{1}{4}\,\left(\beta_{\text{optimal}}\right)^2+16\,\lambda_1\,\lambda_2\,\Sigma_{12}^2$$

Donc $\beta_{\text{optimal}} > 1$ et $F\left[\beta_{\text{optimal}}\right] > 0$

pour que les valeur propre de M soient reelle, il faut aussi que le discriminant > 0

discriminant = trace^2 − 4 determinant

$$\text{Simplify}\left[\text{Tr}\left[\begin{pmatrix}\frac{-\lambda_1}{2} & \frac{\lambda_2\,\Sigma_{\infty 2}-\beta\left(\frac{\nu_2^2}{4}\right)}{2\,\Sigma_{12}}\\ \frac{\lambda_1\,\Sigma_{\infty 1}-\beta\left(\frac{\nu_1^2}{4}\right)}{2\,\Sigma_{12}} & \frac{-\lambda_2}{2}\end{pmatrix}\right]^2-4\,\text{Det}\left[\begin{pmatrix}\frac{-\lambda_1}{2} & \frac{\lambda_2\,\Sigma_{\infty 2}-\beta\left(\frac{\nu_2^2}{4}\right)}{2\,\Sigma_{12}}\\ \frac{\lambda_1\,\Sigma_{\infty 1}-\beta\left(\frac{\nu_1^2}{4}\right)}{2\,\Sigma_{12}} & \frac{-\lambda_2}{2}\end{pmatrix}\right]\ //.\ \left\{\Sigma_{\infty 1}\to\frac{\beta_1\,\nu_1^2}{2\,\lambda_1},\ \Sigma_{\infty 2}\to\frac{\beta_2\,\nu_2^2}{2\,\lambda_2}\right\}\right]$$

$$\frac{(\beta-2\,\beta_1)\,(\beta-2\,\beta_2)\,\nu_1^2\,\nu_2^2+4\,(\lambda_1-\lambda_2)^2\,\Sigma_{12}^2}{16\,\Sigma_{12}^2}$$

donc si β est plus grand que $2\,\beta_1$ et $2\,\beta_2$ ce sera positif

$$\beta\text{Optimal}[\nu1\_,\chi1\_,\Sigma\text{inf1}\_,\nu2\_,\chi2\_,\Sigma\text{inf2}\_]:=\frac{2\,\left(\chi1\,\nu2^2\,\Sigma\text{inf1}+\chi2\,\nu1^2\,\Sigma\text{inf2}\right)}{\nu1^2\,\nu2^2}$$

$$\beta\text{Optimal2}[\nu1\_,\chi1\_,\Sigma\text{inf1}\_,\nu2\_,\chi2\_,\Sigma\text{inf2}\_]:=\frac{4\,\left(\text{Max}\left[\chi1\,\nu2^2\,\Sigma\text{inf1},\,\chi2\,\nu1^2\,\Sigma\text{inf2}\right]\right)}{\nu1^2\,\nu2^2}$$

$$\text{MDeterminant}[\nu1\_,\chi1\_,\Sigma\text{inf1}\_,\nu2\_,\chi2\_,\Sigma\text{inf2}\_,\Sigma12\_,\beta\_]:=$$
$$-\beta^2\,\nu1^2\,\nu2^2+\beta\,\left(4\,\chi1\,\nu2^2\,\Sigma\text{inf1}+4\,\chi2\,\nu1^2\,\Sigma\text{inf2}\right)+16\,\chi1\,\chi2\,\left(\Sigma12^2-\Sigma\text{inf1}\,\Sigma\text{inf2}\right)$$

geometrie du risque : rapprochement avec Heston

```
Clear[Q11, Q12, Q21, Q22]
```

$$\text{Simplify}\left[\text{Solve}\left[\left\{Q11^2+Q21^2==\nu1^2/4,\ Q11\,\rho1+Q21\,\rho2==\frac{\nu1}{2}\,\rho s1,\ Q12^2+Q22^2==\nu2^2/4,\ Q12\,\rho1+Q22\,\rho2==\frac{\nu2}{2}\,\rho s2\right\},\ \{Q11,Q21,Q12,Q22\}\right],\ (\nu1>0)\,\&\&\,(\nu2>0)\right]$$

$$\left\{\left\{Q11\to\frac{\nu1\,\left(\rho1^2\,\rho s1-\rho2\,\sqrt{\rho1^2\,\left(\rho1^2+\rho2^2-\rho s1^2\right)}\right)}{2\,\rho1\,\left(\rho1^2+\rho2^2\right)},\right.\right.$$

$$Q21\to\frac{\nu1\,\left(\rho2\,\rho s1+\sqrt{\rho1^2\,\left(\rho1^2+\rho2^2-\rho s1^2\right)}\right)}{2\,\left(\rho1^2+\rho2^2\right)},$$

$$Q12 \to \frac{\nu 2\left(\rho 1^2\,\rho s2 - \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q22 \to \frac{\nu 2\left(\rho 2\,\rho s2 + \sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\left(\rho 1^2+\rho 2^2\right)}\Big\},$$

$$\Big\{Q11 \to \frac{\nu 1\left(\rho 1^2\,\rho s1 - \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q21 \to \frac{\nu 1\left(\rho 2\,\rho s1 + \sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}\right)}{2\left(\rho 1^2+\rho 2^2\right)},$$

$$Q12 \to \frac{\nu 2\left(\rho 1^2\,\rho s2 + \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q22 \to \frac{\nu 2\,\rho 2\,\rho s2 - \nu 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}}{2\left(\rho 1^2+\rho 2^2\right)}\Big\},$$

$$\Big\{Q11 \to \frac{\nu 1\left(\rho 1^2\,\rho s1 + \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q21 \to \frac{\nu 1\,\rho 2\,\rho s1 - \nu 1\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}}{2\left(\rho 1^2+\rho 2^2\right)},$$

$$Q12 \to \frac{\nu 2\left(\rho 1^2\,\rho s2 - \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q22 \to \frac{\nu 2\left(\rho 2\,\rho s2 + \sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\left(\rho 1^2+\rho 2^2\right)}\Big\},$$

$$\Big\{Q11 \to \frac{\nu 1\left(\rho 1^2\,\rho s1 + \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q21 \to \frac{\nu 1\,\rho 2\,\rho s1 - \nu 1\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s1^2\right)}}{2\left(\rho 1^2+\rho 2^2\right)},$$

$$Q12 \to \frac{\nu 2\left(\rho 1^2\,\rho s2 + \rho 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}\right)}{2\,\rho 1\left(\rho 1^2+\rho 2^2\right)},$$

$$Q22 \to \frac{\nu 2\,\rho 2\,\rho s2 - \nu 2\sqrt{\rho 1^2\left(\rho 1^2+\rho 2^2-\rho s2^2\right)}}{2\left(\rho 1^2+\rho 2^2\right)}\Big\}\Big\}$$

Les quatre solution sont equivalente car redonne les meme observables

```
DetermineQ[ρ1_, ρ2_, ν1_, ν2_, ρs1_, ρs2_, printflag_] :=
 Module[{Q11, Q21, Q12, Q22},
```

$$\text{If}\Big[(\rho 1 \neq 0) \mid\mid (\rho 2 \neq 0),$$

$$\text{Q11} = \frac{\nu 1 \left(\rho 1\, \rho s1 - \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q21} = \frac{\nu 1 \left(\rho 2\, \rho s1 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q12} = \frac{\nu 2 \left(\rho 1\, \rho s2 - \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q22} = \frac{\nu 2 \left(\rho 2\, \rho s2 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{If}\Big[((\text{Q11} == 0)\ \&\&\ (\text{Q12} == 0)) \mid\mid ((\text{Q21} == 0)\ \&\&\ (\text{Q22} == 0)),$$

$$\text{Q11} = \frac{\nu 1 \left(\rho 1^2\, \rho s1 - \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q21} = \frac{\nu 1 \left(\rho 2\, \rho s1 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q12} = \frac{\nu 2 \left(\rho 1^2\, \rho s2 + \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q22} = \frac{\nu 2\, \rho 2\, \rho s2 - \nu 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)}\Big];$$

$$\text{If}\Big[((\text{Q11} == 0)\ \&\&\ (\text{Q12} == 0)) \mid\mid ((\text{Q21} == 0)\ \&\&\ (\text{Q22} == 0)),$$

$$\text{Q11} = \frac{\nu 1 \left(\rho 1^2\, \rho s1 + \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q21} = \frac{\nu 1\, \rho 2\, \rho s1 - \nu 1 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q12} = \frac{\nu 2 \left(\rho 1^2\, \rho s2 - \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$\text{Q22} = \frac{\nu 2 \left(\rho 2\, \rho s2 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}\Big];$$

$$\text{If}\Big[((\text{Q11} == 0)\ \&\&\ (\text{Q12} == 0)) \mid\mid ((\text{Q21} == 0)\ \&\&\ (\text{Q22} == 0)),$$

$$\text{Q11} = \frac{\nu 1 \left(\rho 1^2\, \rho s1 + \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$Q21 = \frac{\nu 1\, \rho 2\, \rho s1 - \nu 1 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)};$$

$$Q12 = \frac{\nu 2 \left(\rho 1^2\, \rho s2 + \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)};$$

$$Q22 = \frac{\nu 2\, \rho 2\, \rho s2 - \nu 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)}\Big];$$

```
, Print["error: ρ1==0 and ρ2==0"];

]
If[printflag > 0, Print["{ρ1,ρ2,ν1,ν2,ρs1,ρs2}=", {ρ1, ρ2, ν1, ν2, ρs1, ρs2}]];
If[printflag > 0, Print["Autres sol=",
```

$$\left\{\left\{\frac{\nu 1 \left(\rho 1^2\, \rho s1 - \rho 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2\, \rho 1 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 1 \left(\rho 2\, \rho s1 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)},\right.\right.$$

$$\left.\frac{\nu 2 \left(\rho 1\, \rho s2 + \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 2\, \rho 2\, \rho s2 - \nu 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)}\right\},$$

$$\left\{\frac{\nu 1 \left(\rho 1\, \rho s1 + \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 1\, \rho 2\, \rho s1 - \nu 1 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)},\right.$$

$$\left.\frac{\nu 2 \left(\rho 1\, \rho s2 - \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 2 \left(\rho 2\, \rho s2 + \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}\right\},$$

$$\left\{\frac{\nu 1 \left(\rho 1\, \rho s1 + \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 1\, \rho 2\, \rho s1 - \nu 1 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s1^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)},\right.$$

$$\left.\left.\left.\frac{\nu 2 \left(\rho 1\, \rho s2 + \rho 2 \sqrt{\left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}\right)}{2 \left(\rho 1^2 + \rho 2^2\right)}, \frac{\nu 2\, \rho 2\, \rho s2 - \nu 2 \sqrt{\rho 1^2 \left(\rho 1^2 + \rho 2^2 - \rho s2^2\right)}}{2 \left(\rho 1^2 + \rho 2^2\right)}\right\}\right\}\right]\Big];$$

```
{{Q11, Q12}, {Q21, Q22}}]
```

```
Module[{ν1 = 0.2, ν2 = 0.2, ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.5, ρs2 = -0.5},
 Q = DetermineQ[ρ1, ρ2, ν1, ν2, ρs1, ρs2, 1]]
```

```
{ρ1,ρ2,ν1,ν2,ρs1,ρs2}={0.5, 0.5, 0.2, 0.2, -0.5, -0.5}

Autres sol={{-0.1, 0., 0., -0.1}, {0., -0.1, -0.1, 0.}, {0., -0.1, 0., -0.1}}

{{-0.1, 0.}, {0., -0.1}}
```

Determination du secteur du drift : rapprochement avec les hestons monodimentionnels

```
Σt = {{Σ11, Σ12}, {Σ12, Σ22}}
```

```
{{Σ11, Σ12}, {Σ12, Σ22}}
```

```
M = {{M11, M12}, {M21, M22}}
{{M11, M12}, {M21, M22}}

Simplify[Σt.M + Transpose[M].Σt]
{{2 (M11 Σ11 + M21 Σ12), M12 Σ11 + M11 Σ12 + M22 Σ12 + M21 Σ22},
  {M12 Σ11 + M11 Σ12 + M22 Σ12 + M21 Σ22, 2 (M12 Σ12 + M22 Σ22)}}

Q = {{Q11, Q12}, {Q21, Q22}};

Transpose[Q].Q
{{Q11^2 + Q21^2, Q11 Q12 + Q21 Q22}, {Q11 Q12 + Q21 Q22, Q12^2 + Q22^2}}
```

```
DetermineM[Q_, β_, Σinf1_, Σinf2_, χ1_, χ2_, Σ012_, flag_] :=
 Module[{M11 = -χ1 / 2, M22 = -χ2 / 2, M21, M12, M, ν1, ν2},
  ν1 = Sqrt[4 (Q[[1, 1]]^2 + Q[[2, 1]]^2)]; ν2 = Sqrt[4 (Q[[1, 2]]^2 + Q[[2, 2]]^2)];

  M21 = (β (Q[[1, 1]]^2 + Q[[2, 1]]^2) - χ1 Σinf1) / (2 Σ012);
  M12 = (β (Q[[1, 2]]^2 + Q[[2, 2]]^2) - χ2 Σinf2) / (2 Σ012);
  M = {{M11, M12}, {M21, M22}};
  If[flag == 1,
   Print["Q_,β,Σinf1,Σinf2,χ1,χ2,Σ012=", {Q, β, Σinf1, Σinf2, χ1, χ2, Σ012}];
   Print["ν1^2/2-χ1 Σinf1 doit etre negatif 1 : ", ν1^2 / 2 - χ1 Σinf1];
   Print["ν2^2/2-χ2 Σinf2 doit etre negatif 2 : ", ν2^2 / 2 - χ2 Σinf2];
   Print["eigenvalues[M]=", Eigenvalues[M]];];
  M
 ]
```

il faut verifier que (Tr[M] < 0 et Det[M] > 0)

```
Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15, Σinf1 = 0.15, Σinf2 = 0.15,
   Σ012 = 0.12, ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, β = 3, Q, M},
 Q = DetermineQ[ρ1, ρ2, ν1, ν2, ρs1, ρs2, 0];
 M = DetermineM[Q, β, Σinf1, Σinf2, χ1, χ2, Σ012, 1];
]
ν1^2/2-χ1 Σinf1 doit etre negatif 1 : -0.0025
ν2^2/2-χ2 Σinf2 doit etre negatif 2 : -0.0025
eigenvalues[M]={-0.10625, -0.04375}
```

Appendix 3

Matrix Exponential

If faut montrer que F = a11 = $\left(\dfrac{2\xi}{e^{\xi\tau/2}\,((M+\rho Q\gamma)+\xi)-e^{-\xi\tau/2}\,(M+\rho Q\gamma-\xi)}\right)$

mais pour obtenir le bon A on doit changer en l ' opposé : (peut etre t <→ τ)

$$H = \begin{pmatrix} \frac{(M+Q\rho\gamma)}{2} & \frac{QQ}{2} \\ \frac{-\gamma\gamma}{2} & \frac{-(M+Q\rho\gamma)}{2} \end{pmatrix};\ \xi 1 = \text{Simplify}[-4\,\text{Det}[H]]$$

$$M^2 + 2MQ\gamma\rho + Q^2\gamma^2\left(-1+\rho^2\right)$$

```
F = Simplify[
    Simplify[expM1.{1, 0} /. {a → H[[1, 1]], b → H[[1, 2]], c → H[[2, 1]], d → H[[2, 2]]}] ][[
   1]] //. {√(M^2 + 2 M Q γ ρ + Q^2 γ^2 (-1 + ρ^2)) → ξ}
```

$$\frac{e^{-\frac{\xi\tau}{2}}\left(\left(-1+e^{\xi\tau}\right)M+\left(1+e^{\xi\tau}\right)\xi+\left(-1+e^{\xi\tau}\right)Q\gamma\rho\right)}{2\sqrt{M^2+2MQ\gamma\rho+Q^2\gamma^2\left(-1+\rho^2\right)}}$$

$$\frac{e^{-\frac{\xi\tau}{2}}\left(\left(-1+e^{\xi\tau}\right)M+\left(1+e^{\xi\tau}\right)\xi+\left(-1+e^{\xi\tau}\right)Q\gamma\rho\right)}{2\xi}$$

Si on diagonalise cette matrice ,
attention les matrice de passage ne sont pas orthogonale car la matrice n ' est pas symetrique.

Soit une matrice h = $\begin{pmatrix} a & b \\ c & d \end{pmatrix}$ on la diagonalise en :

$$h = \frac{1}{1+\left(\frac{-a+d+\sqrt{\Delta}}{2b}\right)^2}\begin{pmatrix} 1 & \frac{-a+d+\sqrt{\Delta}}{2b} \\ \frac{-a+d+\sqrt{\Delta}}{2b} & \left(\frac{-a+d+\sqrt{\Delta}}{2b}\right)^2 \end{pmatrix} x_1 + \frac{1}{1+\left(\frac{-a+d-\sqrt{\Delta}}{2b}\right)^2}\begin{pmatrix} 1 & \frac{-a+d-\sqrt{\Delta}}{2b} \\ \frac{-a+d-\sqrt{\Delta}}{2b} & \left(\frac{-a+d-\sqrt{\Delta}}{2b}\right)^2 \end{pmatrix} x_2$$

ou $x_1 = \dfrac{a+d+\sqrt{\Delta}}{2}$; $x_2 = \dfrac{a+d-\sqrt{\Delta}}{2}$; $\Delta = a\,d - b\,c$;

$$M1 = \begin{pmatrix} 7.1 & 2 \\ 41 & 3 \end{pmatrix}$$

{{7.1, 2}, {41, 3}}

$$M1 = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

```
Simplify[MatrixExp[M1]]
```

$$\left\{\left\{\frac{1}{2\sqrt{a^2+4bc-2ad+d^2}}e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)}\left(d-d\,e^{\sqrt{a^2+4bc-2ad+d^2}}+a\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)+\sqrt{a^2+4bc-2ad+d^2}\left(1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)\right),\frac{b\,e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)}\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)}{\sqrt{a^2+4bc-2ad+d^2}}\right\},\left\{\frac{c\,e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)}\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)}{\sqrt{a^2+4bc-2ad+d^2}},\frac{1}{2\sqrt{a^2+4bc-2ad+d^2}}e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)}\left(a-a\,e^{\sqrt{a^2+4bc-2ad+d^2}}+d\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)+\sqrt{a^2+4bc-2ad+d^2}\left(1+e^{\sqrt{a^2+4bc-2ad+d^2}}\right)\right)\right\}\right\}$$

$$\{x1,x2\}=\left\{\frac{a+d+\sqrt{(a+d)^2-4(ad-bc)}}{2},\frac{a+d-\sqrt{(a+d)^2-4(ad-bc)}}{2}\right\};$$

$$P1=\left\{\left\{-\frac{-a+d+\sqrt{a^2+4bc-2ad+d^2}}{2c},1\right\},\left\{-\frac{-a+d-\sqrt{a^2+4bc-2ad+d^2}}{2c},1\right\}\right\};$$

$$InvP1=\left\{\left\{-\frac{c}{\sqrt{a^2+4bc-2ad+d^2}},\frac{c}{\sqrt{a^2+4bc-2ad+d^2}}\right\},\left\{-\frac{-a+d-\sqrt{a^2+4bc-2ad+d^2}}{2\sqrt{a^2+4bc-2ad+d^2}},\frac{-a+d+\sqrt{a^2+4bc-2ad+d^2}}{2\sqrt{a^2+4bc-2ad+d^2}}\right\}\right\};$$

```
Simplify[Transpose[(P1)].DiagonalMatrix[{x2, x1}].Transpose[(InvP1 )]]
{{a, b}, {c, d}}
```

```
expM1 =
 Simplify[Transpose[(P1)].DiagonalMatrix[{Exp[τ x2], Exp[τ x1]}].Transpose[(InvP1 )]]
```

$$\left\{\left\{\frac{1}{2\sqrt{a^2+4bc-2ad+d^2}}e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)\tau}\left(d-d\,e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}+a\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}\right)+\sqrt{a^2+4bc-2ad+d^2}\left(1+e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}\right)\right),\frac{b\left(-e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)\tau}+e^{\frac{1}{2}\left(a+d+\sqrt{a^2+4bc-2ad+d^2}\right)\tau}\right)}{\sqrt{a^2+4bc-2ad+d^2}}\right\},\left\{\frac{c\left(-e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)\tau}+e^{\frac{1}{2}\left(a+d+\sqrt{a^2+4bc-2ad+d^2}\right)\tau}\right)}{\sqrt{a^2+4bc-2ad+d^2}},\frac{1}{2\sqrt{a^2+4bc-2ad+d^2}}e^{\frac{1}{2}\left(a+d-\sqrt{a^2+4bc-2ad+d^2}\right)\tau}\left(a-a\,e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}+d\left(-1+e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}\right)+\sqrt{a^2+4bc-2ad+d^2}\left(1+e^{\sqrt{a^2+4bc-2ad+d^2}\,\tau}\right)\right)\right\}\right\}$$

```
MatrixId[M_] := Module[{ss = Eigensystem[M], P, InvP, eigen},
  eigen = ss[[1]];
  P = ss[[2]];
  InvP = Inverse[P];
  (Transpose[P].DiagonalMatrix[eigen]).Transpose[InvP]
 ]
```

```
MatrixExp2[{{a_, b_}, {c_, d_}}] :=
 Module[{det = Sqrt[a^2 + 4 b c - 2 a d + d^2], expdet, expab}, expdet = e^det;
  expab = e^(1/2 (a+d-det));
  {{(expab (d - d expdet + a (-1 + expdet) + det (1 + expdet)))/(2 det), (b expab (-1 + expdet))/det},
   {(c expab (-1 + expdet))/det, (expab (a - a expdet + d (-1 + expdet) + det (1 + expdet)))/(2 det)}}]
```

Matrix Logarithm

```
MatrixLog2[{{a_, b_}, {c_, d_}}] :=
 Module[{logdetm, adm, adp, logdetp, log4, P, InvP, det = Sqrt[a^2 + 4 b c - 2 a d + d^2]},
  adm = a + d - det; adp = a + d + det;
  If[(Im[adm] == 0) && (Re[adm] ≤ 0), logdetm = 0, logdetm = Log[adm]];
  If[(Im[adp] == 0) && (Re[adp] ≤ 0), logdetp = 0, logdetp = Log[adp]];
  log4 = Log[4];
  {{(-det log4 + (-a + d + det) logdetm + (a - d + det) logdetp)/(2 det), (-b (logdetm - logdetp))/det},
   {(c (-logdetm + logdetp))/det, (-det log4 + (a - d + det) logdetm + (-a + d + det) logdetp)/(2 det)}}
 ]
```

```
mm = MatrixExp2[{{11, 2}, {1.2, 4.2}}]
```

```
{{80040.3, 22418.5}, {13451.1, 3817.4}}
```

```
MatrixLog2[mm]
```

```
{{11., 2.}, {1.2, 4.2}}
```

la trace du log est tres simple a calculer :

```
TrMatrixLog2[{{a_, b_}, {c_, d_}}] := Module[{},
  Log[- b c + a d]
 ]
```

Implementation

Appendix 4

Lognormal Heston Formula and 1 dim Wishart case

Explicitation of the matrix method

```
Placement1 = {{1, 0}}; Placement2 = {{0, 1}};
```

$$H = -\left(\frac{M - i\,Q\,\rho\,\gamma}{2}\right)(\text{Transpose}[\text{Placement1}].\text{Placement1}) + (-Q\,Q/2)(\text{Transpose}[\text{Placement1}].\text{Placement2}) + (-\gamma\gamma/2 + i\,\gamma/2)(\text{Transpose}[\text{Placement2}].\text{Placement1}) + \left(\frac{M - i\,Q\,\rho\,\gamma}{2}\right)(\text{Transpose}[\text{Placement2}].\text{Placement2});$$

```
{{E11, E12}, {E21, E22}} = Simplify[MatrixExp2[τ H]];
```

```
ξexp = -2 Simplify[E11[[2, 2]]] /. τ → 1
```

$$\sqrt{M^2 - 2\,i\,M\,Q\,\gamma\,\rho + Q^2\,\gamma\,(-i + \gamma - \gamma\,\rho^2)}$$

```
ξexp[[1]]
```

$$M^2 - 2\,i\,M\,Q\,\gamma\,\rho + Q^2\,\gamma\,(-i + \gamma - \gamma\,\rho^2)$$

```
ExtractCarré[exp_, var_] :=
 Module[{a, exp1 = Coefficient[exp, var, 1], exp2 = Coefficient[exp, var, 2]},
  a = exp1 / (2 exp2);
  exp2 (var + a)^2 + Simplify[exp - exp2 (var + a)^2, Evaluate[var] > 0]]
```

```
ExtractCarré[ξexp[[1]], M]
```

$$Q^2\,\gamma\,(-i + \gamma) + (M - i\,Q\,\gamma\,\rho)^2$$

$$\text{Simplify}\left[M^2 - 2\,i\,M\,Q\,\gamma\,\rho - Q^2\,\gamma^2\,(\rho^2)\right]$$

$$(M - i\,Q\,\gamma\,\rho)^2$$

$$XX = \text{Collect}\left[e^{-\frac{\xi\tau}{2}}\,\text{Simplify}[E11, \tau > 0]\ /.\ \{(\xi\text{exp}[[1]])^{n\_} \to \xi^{2n}\},\ e^{\xi\tau}\right]$$

$$\frac{e^{-\xi\tau}\,(M + \xi - i\,Q\,\gamma\,\rho)}{2\,\xi} + \frac{-M + \xi + i\,Q\,\gamma\,\rho}{2\,\xi}$$

$$\text{Log}[XX] + \frac{\tau}{2}\,(M - i\,\gamma\,\rho\,Q + \xi)$$

$$\frac{1}{2}\,(M + \xi - i\,Q\,\gamma\,\rho)\,\tau + \text{Log}\left[\frac{e^{-\xi\tau}\,(M + \xi - i\,Q\,\gamma\,\rho)}{2\,\xi} + \frac{-M + \xi + i\,Q\,\gamma\,\rho}{2\,\xi}\right]$$

```
B = Simplify[1/2 ψp τ + Log[(ψm + ψp e^(-ξ τ))/(2 ξ)] /. {ψp → - (M + I z ρ Q) + ξ, ψm → (M + I z ρ Q) + ξ}]
```

$$-\frac{1}{2}\left(M-\zeta+i\,Q\,z\,\rho\right)\tau+\text{Log}\left[\frac{M+\zeta+i\,Q\,z\,\rho+e^{-\zeta\tau}\left(-M+\zeta-i\,Q\,z\,\rho\right)}{2\,\zeta}\right]$$

```
YY = Simplify[E21, τ > 0] /. {(ξexp[[1]])^n_ → ξ^(2 n)}
```

$$-\frac{e^{-\frac{\zeta\tau}{2}}\left(-1+e^{\zeta\tau}\right)\gamma\left(-i+\gamma\right)}{2\,\zeta}$$

```
A = Simplify[e^(-ξ τ/2) YY / XX]
```

$$\frac{\left(-1+e^{\zeta\tau}\right)\gamma\left(-i+\gamma\right)}{\left(-1+e^{\zeta\tau}\right)M-\left(1+e^{\zeta\tau}\right)\zeta-i\left(-1+e^{\zeta\tau}\right)Q\,\gamma\,\rho}$$

Implementation

```
Clear[LNHestonVanillaCall, LNHestonVanillaCallIntegrand, HestonPropagator]
```

```
HestonPropagator[V_, τ_, λ_, θ_, ν_, ρ_, z_] := Module[{ψp, ψm, ξ, X},
  ξ = Sqrt[(λ + I z ρ ν)^2 + ν^2 (-I z + z^2)];
  ψp = - (λ + I z ρ ν) + ξ;
  ψm = (λ + I z ρ ν) + ξ;
  A = -θ λ / ν^2 (ψp τ + 2 Log[(ψm + ψp e^(-ξ τ))/(2 ξ)]);
  B = - (-I z + z^2) (1 - e^(-ξ τ))/(ψm + ψp e^(-ξ τ));
  e^(A+B V)]
```

```
FourierPayOffLNSepp[z_, k_] := k^(i z+1)/(z^2 - i z)
```

$\text{Max}[F-K,0]-F=(F-K)\,1_{F>K}-F\,(1_{F>K}+1_{F<K})=-K\,1_{F>K}-F\,1_{F<K}=-(K\,1_{F>K}+F\,1_{F<K})=-\text{Min}[F,K]$

on montre FourierTransform[Min[x, K], x, z] == $\frac{K^{i\,z+1}}{z^2-i\,z}$

On passe par Min car le call n' est pas borné donc la theorie ne s' aaplique pas.D' allieurs on trouve une pole pour passer du call au min

```
LNHestonVanillaCallIntegrand[F_, K_, z_, V_, τ_, θ_, λ_, ν_, ρ_] :=
 Module[{ψp, ψm, ξ, X},
  Re[e^(-Log[F] I z) HestonPropagator[V, τ, λ, θ, ν, ρ, z] × FourierPayOffLNSepp[z, K]]]
```

```
LNHestonVanillaCall[F_, K_, V0_, τ_, λ_, θ_, ν_, ρ_, limsup_] :=
 F + 1 / Pi NIntegrate[
    LNHestonVanillaCallIntegrand[F, K, k1 + I / 2, V0, τ, θ, λ, ν, ρ],
    {k1, 0, limsup}, MaxRecursion → 20]
```

```
LNHestonRiccatiVanillaCallIntegrand[F_, K_, z_, V_, τ_, θ_, λ_, ν_, ρ_] :=
 Module[{ψp, ψm, ξ, X},
  Re[ⅇ^(-Log[F] I z)
    HestonLaplaceTransform4[-λ, θ, ρ, V, -ⅈ z, ν, τ] × FourierPayOffLNSepp[z, K]]]
```

```
LNHestonRiccatiVanillaCall[F_, K_, V0_, τ_, λ_, θ_, ν_, ρ_, limsup_] :=
 F + 1 / Pi NIntegrate[
    LNHestonRiccatiVanillaCallIntegrand[F, K, k1 + I / 2, V0, τ, θ, λ, ν, ρ],
    {k1, 0, limsup}, MaxRecursion → 20]
```

Si on inclue le drift venant du fait que l' underlying est un log d' asset

```
HestonLaplaceTransform4[M_, θ_, ρ_, Σ_, γ_, Q_, τ_] :=
 Module[{H, EXPH, A11, A1, A21, A, β, ν11, ν12, ν21, ν22, c,
    Placement1 = {{1, 0}}, Placement2 = {{0, 1}}, ξ},
  β = Sqrt[-M θ] / Q;
  H = -((M + Q ρ γ)/2) (Transpose[Placement1] .Placement1) +
     (-Q Q / 2) (Transpose[Placement1] .Placement2) +
     (γ γ / 2 - γ / 2) (Transpose[Placement2] .Placement1) +
     ((M + Q ρ γ)/2) (Transpose[Placement2] .Placement2) ;
  EXPH = MatrixExp2[τ H];
  A11 = EXPH[[1, 1]];
  A21 = EXPH[[2, 1]];
  A = A21/A11;
  c = -2 β^2 (Log[A11] + τ/2 (M + γ ρ Q));
  Exp[A Σ + c]
 ]
```

```
HestonFourierTransform4[M_, θ_, ρ_, Σ_, γ_, β_, τ_] :=
 HestonLaplaceTransform4[M, θ, ρ, Σ, -ⅈ γ, β, τ]
```

```
Module[{λ = 0.05, ν = 0.2, ρ = -0.5, Σ = 0.04, Σinf = 0.05, τ = 5, z = 1},
 {HestonFourierTransform4[-λ, Σinf, ρ, Σ, z, ν, τ],
  HestonPropagator[Σ, τ, λ, Σinf, ν, ρ, z]}]
```

```
{0.888791+ 0.0581648 ⅈ, 0.888791+ 0.0581648 ⅈ}
```

```
Module[{S = 0.05, K = 0.06, T = 5, V = 0.04,
  Vinf = 0.04, ρ = -0.5, λ = 0.05, ν = 0.2, limsup = 3000},
 {LNHestonRiccatiVanillaCall[S, K, V, T, λ, Vinf, ν, ρ, limsup],
  LNHestonVanillaCall[S, K, V, T, λ, Vinf, ν, ρ, limsup]}]
```

```
{0.00324844, 0.00324844}
```

Normal Heston case

```
HestonLaplaceTransform3[M_, θ_, ρ_, Σ_, γ_, Q_, τ_] :=
 Module[{H, EXPH, A11, A1, A21, A, ν11, ν12, ν21, ν22, c,
   Placement1 = {{1, 0}}, Placement2 = {{0, 1}}, ξ, β = Sqrt[-M θ]/Q},
  H = -((M + Q ρ γ)/2) (Transpose[Placement1] .Placement1) +
     (-Q Q / 2) (Transpose[Placement1] .Placement2) +
     (γ γ / 2) (Transpose[Placement2] .Placement1) +
     ((M + Q ρ γ)/2) (Transpose[Placement2] .Placement2) ;
  EXPH = MatrixExp2[τ H];
  A11 = EXPH[[1, 1]];
  A21 = EXPH[[2, 1]];
  A = A21/A11;
  c = -2 β^2 (Log[A11] + τ/2 (M + γ ρ Q));
  Exp[A Σ + c]
 ]
```

```
HestonFourierTransform[M_, θ_, ρ_, Σ_, γ_, β_, τ_] :=
 HestonLaplaceTransform[M, θ, ρ, Σ, -ⅈ γ, β, τ]
```

```
HestonFourierTransform2[M_, θ_, ρ_, Σ_, γ_, β_, τ_] :=
 HestonLaplaceTransform2[M, θ, ρ, Σ, -ⅈ γ, β, τ]
```

```
HestonFourierTransform3[M_, θ_, ρ_, Σ_, γ_, β_, τ_] :=
 HestonLaplaceTransform3[M, θ, ρ, Σ, -ⅈ γ, β, τ]
```

```
GaussianHestonFondamentalTransform[ρ_, M_, Σinf_, Q_, V_, γ_, τ_] :=
 Module[{ξ, ψp , ψm, A, B, argξ},
  argξ = (M + ⅈ ρ Q γ)^2 + Q^2 (γ^2);
  If[Abs[argξ] ≤ 10^(-200),
   B = (M + ⅈ ρ γ Q)/Q^2 - ((M - ⅈ ρ γ Q) Q^2)/(τ (M + ⅈ ρ γ Q) + Q^4);
   A = 1/(2 Q^2) (Σinf M (2 M τ + 2 ⅈ Q ρ τ γ -
          2 ⅈ Q^4 ArcTan[(Q ρ τ γ)/(Q^4 + M τ)] + Q^4 Log[Q^8/(Q^8 + 2 Q^4 M τ + M^2 τ^2 + Q^2 ρ^2 τ^2 γ^2)]));
   ⅇ^(A+B V),
   ξ = Sqrt[(M + ⅈ ρ Q γ)^2 + Q^2 (γ^2)] ;
   ψp  = - (M + ⅈ ρ Q γ) + ξ;
   ψm  = (M + ⅈ ρ Q γ) + ξ;
   A = - (Σinf M (τ ψp + 2 Log[(ψm+ⅇ^(-ξ τ) ψp)/(2 ξ)]))/Q^2;
   B = (- (γ^2) (1 - ⅇ^(-ξ τ)))/(ψm + ψp ⅇ^(-ξ τ));
   If[Re[A + B V] < -100, 0,
    ⅇ^(A+B V)]]]
```

```
Module[{M = -0.01, θ = 0.1, ρ = 0.5, Σ = 0.15, γ = 1, Q = 2, τ = 10},
 {HestonFourierTransform3[M, θ, ρ, Σ, γ, Q, τ],
  GaussianHestonFondamentalTransform[ρ, -M, θ, Q, Σ, γ, τ]}]
```

A11=2857.98+ 1748.79 ⅈ

$\frac{\psi m + e^{-\xi\tau}\psi p}{2\xi}$=0.503849+ 0.288656 ⅈ

{0.93296+ 0.0368795 ⅈ, 0.93296+ 0.0368795 ⅈ}

Appendix 6

Log normale spreadoption (pour comparaison)

```
phi[x_] := Exp[-x^2 / 2] / Sqrt[2 Pi]
```

```
Nd[x_] := (Erf[x/Sqrt[2]] + 1)/2
```

```
LogNormalSpreadDigitaleCall[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 NIntegrate[phi[x] × Nd[- (Log[(k + S2 E^(-1/2 sig2^2 t+sig2 Sqrt[t] x)) / (S1 E^(-1/2 sig1^2 t))] - rho sig1 Sqrt[t] x) /
     (Sqrt[1 - rho^2] sig1 Sqrt[t])], {x, -Infinity, -1, +Infinity}]
```

```
MesureQTLogNormalSpreadDigitale[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 LogNormalSpreadDigitaleCall[S1 , S2 , sig1, sig2, rho, k, t]
```

```
MesureQ1LogNormalSpreadDigitale[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 LogNormalSpreadDigitaleCall[S1 Exp[sig1^2 t],
  S2 Exp[rho sig1 sig2 t], sig1, sig2, rho, k, t]
```

```
MesureQ2LogNormalSpreadDigitale[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 LogNormalSpreadDigitaleCall[S1 Exp[rho sig1 sig2 t],
  S2 Exp[sig2^2 t], sig1, sig2, rho, k, t]
```

```
LogNormalSpreadOptionAux[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 S1 MesureQ1LogNormalSpreadDigitale[S1, S2, sig1, sig2, rho, k, t] -
  S2 MesureQ2LogNormalSpreadDigitale[S1, S2, sig1, sig2, rho, k, t] -
  k MesureQTLogNormalSpreadDigitale[S1, S2, sig1, sig2, rho, k, t]
```

```
LogNormalSpreadOption[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 LogNormalSpreadOptionAux[S1, S2, sig1, sig2, rho, k, t] /; k ≥ 0
```

```
LogNormalSpreadOption[S1_, S2_, sig1_, sig2_, rho_, k_, t_] :=
 S1 - S2 - k + LogNormalSpreadOptionAux[S2, S1, sig2, sig1, rho, -k, t] /; k < 0
```

```
Module[{S1 = 0.05, S2 = 0.05, sig1 = 0.2, sig2 = 0.3, ρ = 0.8, k = 0.01, t = 10},
 LogNormalSpreadOption[S1, S2, sig1, sig2, ρ, k, t]]
```

```
0.00596171
```

```
Module[{S1 = 0.05, S2 = 0.05, sig1 = 0.2,
  sig2 = 0.2, k1 = 0.001, k2 = 0.01, k3 = 0.03, t = 5, ρ = 0.6},
 {LogNormalSpreadOption[S1, S2, sig1, sig2, ρ, k1, t],
  LogNormalSpreadOption[S1, S2, sig1, sig2, ρ, k2, t],
  LogNormalSpreadOption[S1, S2, sig1, sig2, ρ, k3, t]}]
```

```
{0.00743791, 0.00406613, 0.00100504}
```

Appendix 7

Monte Carlo functions used to test the closed form formulas

Monte Carlo functions for the Heston case

```
HestonGeneratePath[M_, θ_, ρ1_, Σ_, S1_, β_, TimeStepsNb_, dt_, printflag_] :=
 Module[{Cov, i, j, Z, W, W11, W12, W21, W22, Z1,
   Z2, Q, Yn, Y, Σn, ΣinfM, σ, sqdt = √dt , Σperturb, alea},

  ΣinfM = θ; Y = Log[S1];
  Q = Sqrt[- (M ΣinfM)] / β;
  Σn = Σ; Yn = Y;
  Cov = dt {
     {1, ρ1},
     {ρ1, 1}
    };
  If[printflag ≥ 1, Print["C=", Cov // MatrixForm]];
  Do[alea = Random[MultinormalDistribution[{0, 0}, Cov]];
   If[printflag ≥ 2, Print["alea=", alea // MatrixForm]];
   {Z, W} = alea;
   σ = Sqrt[Σ];
   (*
   Σperturb=M (Σn-ΣinfM)dt+σ   W Q;
   Σn+=Σperturb;
   *)
   Σn *= (M (1 - ΣinfM/Σn) - 1/2 (Q/σ)^2) dt + W Q/σ;

   Yn += σ Z - dt/2 Σn;
   If[printflag ≥ 3, Print["Σperturb=", Σperturb // MatrixForm]];
   If[printflag ≥ 4, Print["Σn=", Σn // MatrixForm, " Yn=", Yn // MatrixForm]];
   , {i, 1, TimeStepsNb}];
  {Exp[Yn], Σn}]
```

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150,
   flag = 1, vol1, vol2, spdopt, strikes, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 1, dt, printflag = 0}, dt = T/TimeStepsNb;
  strikes = {0.001};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;

  HestonGeneratePath[M1, Σinf1, ρ1, Σ1, S1, β, TimeStepsNb, dt, 0]]]
```

β=2.25

{0.109, {0.0236387, 0.0582514}}

```
HestonGenerateSample[M_, θ_, ρ1_,
   Σ_, S1_, β_, TimeStepsNb_, dt_, nbSample_, printflag_] :=
  Module[{k, samples}, samples = Table[HestonGeneratePath[M, θ, ρ1, Σ, S1,
      β, TimeStepsNb, dt, printflag], {k, 1, nbSample}];
   samples]
```

```
ResSimul = Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150,
    flag = 1, vol1, vol2, spdopt, strikes, M, Q, Σ, M1, M2, ρm1, ρm2,
    TimeStepsNb = 100, nbSample = 100, dt, printflag = 0}, dt = T/TimeStepsNb;
   strikes = {0.001};
   β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
   z1max = 1; z2max = 3;
   scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ];
   M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
   M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
   HestonGenerateSample[M1, Σinf1,
    ρ1, Σ1, S1, β, TimeStepsNb, dt, nbSample, printflag]]]
```

β=2.25

```
HestonMonteCarloOption[K_, M_, θ_, ρ_, Σ11_, S1_, β_,
  TimeStepsNb_, dt_, nbSample_, printflag_] := Module[{samples},
  samples = HestonGenerateSample[M, θ, ρ, Σ11, S1,
    β, TimeStepsNb, dt, nbSample, printflag];
  Sum[If[samples⟦i, 1⟧ - K ≥ 0, samples⟦i, 1⟧ - K, 0], {i, 1, Length[samples]}] /
   Length[samples]]
```

```
HestonMonteCarloSmile[StrikeList_, M_, θ11_, ρ1_, Σ11_, S1_, β_,
  TimeStepsNb_, dt_, nbSample_, printflag_] := Module[{samples, i, k},
  samples = HestonGenerateSample[M, θ11, ρ1, Σ11,
    S1, β, TimeStepsNb, dt, nbSample, printflag];
  Table[Sum[If[samples⟦i, 1⟧ - StrikeList⟦k⟧ ≥ 0, samples⟦i, 1⟧ - StrikeList⟦k⟧, 0],
     {i, 1, Length[samples]}] / Length[samples], {k, 1, Length[StrikeList]}]]
```

Monte Carlo functions for the Bi Heston case

```
BiHestonGeneratePath[{{M11_, M12_}, {M21_, M22_}}, Q_, ΣinfM_, {ρ1_, ρ2_},
  {Σ11_, Σ12_, Σ22_}, {S1_, S2_}, β_, TimeStepsNb_, dt_, printflag_] :=
 Module[{Cov, i, j, Z, W, W11, W12, W21, W22, Z1, Z2, M,
    Σ, Yn, Y, Σn, σ, sqdt = √dt, Σperturb, alea},
  M = {{M11, M12}, {M21, M22}};
  Σ = (Σ11  Σ12
       Σ12  Σ22); Y = {Log[S1], Log[S2]};
  Σn = Σ; Yn = Y;
  Cov = dt {
      {1, 0, ρ1, ρ2, 0, 0},
      {0, 1, 0, 0, ρ1, ρ2},
      {ρ1, 0, 1, 0, 0, 0},
      {ρ2, 0, 0, 1, 0, 0},
      {0, ρ1, 0, 0, 1, 0},
      {0, ρ2, 0, 0, 0, 1}
    };
  If[printflag == 1, Print["C=", Cov // MatrixForm]];
  Do[alea = Random[MultinormalDistribution[{0, 0, 0, 0, 0, 0}, Cov]];
   If[printflag == 2, Print["alea=", alea // MatrixForm]];
   {Z1, Z2, W11, W12, W21, W22} = alea;
   Z = {Z1, Z2}; W = {{W11, W12}, {W21, W22}};
   σ = CholeskyDecomposition[Σ];
   Σperturb = M (Σn - ΣinfM) dt + σ . W.Q;
   Σn += Σperturb + Transpose[Σperturb];
   Yn += σ.Z - dt/2 {Σn[[1, 1]], Σn[[2, 2]]};
   If[printflag == 3, Print["Σperturb=", Σperturb // MatrixForm]];
   If[printflag == 4, Print["Σn=", Σn // MatrixForm, " Yn=", Yn // MatrixForm]];
   , {i, 1, TimeStepsNb}];
  {{Exp[Yn[[1]]], Exp[Yn[[2]]]}, Σn}]
```

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150, flag = 1,
   vol1, vol2, spdopt, strikes, M, Q, Σ, M1, M2, ρm1, ρm2, TimeStepsNb = 100,
   nbSample = 1, dt, printflag = 0, ΣinfM}, dt = T/TimeStepsNb;
  strikes = {0.001};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ]; scope2 = z2max/Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)]/β;
  BiHestonGeneratePath[M, Q, ΣinfM, {ρ1, ρ2},
   {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, printflag]]]
```

β=2.25

{0.141, {{0.0227021, 0.0339462}, {{0.091216, 0.0429738}, {0.0429738, 0.125853}}}}

```
BiHestonGenerateSample[{{M11_, M12_}, {M21_, M22_}},
  Q_, ΣinfM_, {ρ1_, ρ2_}, {Σ11_, Σ12_, Σ22_}, {S1_, S2_}, β_,
  TimeStepsNb_, dt_, nbSample_, printflag_] := Module[{k, samples},
  samples = Table[BiHestonGeneratePath[{{M11, M12}, {M21, M22}}, Q, ΣinfM, {ρ1, ρ2},
     {Σ11, Σ12, Σ22}, {S1, S2}, β, TimeStepsNb, dt, printflag], {k, 1, nbSample}];
  samples]
```

```
ResSimul = Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150, flag = 1,
    vol1, vol2, spdopt, strikes, M, Q, Σ, M1, M2, ρm1, ρm2, TimeStepsNb = 100,
    nbSample = 100, dt, printflag = 0, ΣinfM}, dt = T / TimeStepsNb;
  strikes = {0.001};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 τ];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1                  ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]     M2              );
  ΣinfM = ( Σinf1                          Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12       Σinf2                    );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M]) / 2)] / β;
  BiHestonGenerateSample[M, Q, ΣinfM, {ρ1, ρ2},
    {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag]]];
ResSimul[[1]]
```

β=2.25

9.937

```
BiHestonMonteCarloOption[K_, {{M11_, M12_}, {M21_, M22_}},
  Q_, ΣinfM_, {ρ1_, ρ2_}, {Σ11_, Σ12_, Σ22_}, {S1_, S2_}, β_,
  TimeStepsNb_, dt_, nbSample_, printflag_] := Module[{samples},
  samples = BiHestonGenerateSample[{{M11, M12}, {M21, M22}}, Q, ΣinfM, {ρ1, ρ2},
    {Σ11, Σ12, Σ22}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  Sum[If[samples[[i, 1, 1]] - samples[[i, 1, 2]] - K ≥ 0, samples[[i, 1, 1]] -
      samples[[i, 1, 2]] - K, 0], {i, 1, Length[samples]}] / Length[samples]]
```

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
    Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
    ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
    ρinf12 = 0.8, β, K = 0.001, integflag = 0,
    T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150, flag = 1,
    vol1, vol2, spdopt, strikes, M, Q, Σ, M1, M2, ρm1, ρm2, TimeStepsNb = 100,
    nbSample = 100, dt, printflag = 0, ΣinfM}, dt = T / TimeStepsNb;
  strikes = {0.001};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  ΣinfM = {{Σinf1, Sqrt[Σinf1 Σinf2] ρinf12}, {Sqrt[Σinf1 Σinf2] ρinf12, Σinf2}};
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  BiHestonMonteCarloOption[K, M, Q, ΣinfM, {ρ1, ρ2},
   {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag]]]
```

β=2.25

{9.937, 0.0105103}

```
BiHestonMonteCarloSmile[StrikeList_, {{M11_, M12_}, {M21_, M22_}},
  Q_, ΣinfM_, {ρ1_, ρ2_}, {Σ11_, Σ12_, Σ22_}, {S1_, S2_}, β_,
  TimeStepsNb_, dt_, nbSample_, printflag_] := Module[{samples, i, k},
  samples = BiHestonGenerateSample[{{M11, M12}, {M21, M22}}, Q, ΣinfM, {ρ1, ρ2},
    {Σ11, Σ12, Σ22}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag];
  Table[Sum[If[samples[[i, 1, 1]] - samples[[i, 1, 2]] - StrikeList[[k]] ≥ 0,
      samples[[i, 1, 1]] - samples[[i, 1, 2]] - StrikeList[[k]], 0],
     {i, 1, Length[samples]}] / Length[samples], {k, 1, Length[StrikeList]}]]
```

```
Timing[Module[{ν1 = 0.2, ν2 = 0.2, χ1 = 0.15, χ2 = 0.15,
   Σ1 = 0.04, Σ2 = 0.04, Σinf1 = 0.15, Σinf2 = 0.15, S1 = 0.04, S2 = 0.040,
   ρ1 = 0.5, ρ2 = 0.5, ρs1 = -0.6, ρs2 = -0.6, ρ12 = 0.8,
   ρinf12 = 0.8, β, K = 0.001, integflag = 0,
   T = 5, zmax, ω1 = 1, λ1 = 1.1, λ2 = 1.2, z1max, z2max, Nb = 150,
   flag = 1, vol1, vol2, spdopt, StrikeList, M, Q, Σ, M1, M2, ρm1, ρm2,
   TimeStepsNb = 100, nbSample = 100, dt, printflag = 0, ΣinfM},
  dt = T/TimeStepsNb;
  StrikeList = {-0.02, -0.015, -0.01, -0.0075, -0.005,
    -0.003, -0.001, 0.001, 0.003, 0.005, 0.0075, 0.01, 0.015, 0.02};
  β = βOptimal2[ν1, χ1, Σinf1, ν2, χ2, Σinf2]; Print["β=", β];
  z1max = 1; z2max = 3;
  scope1 = z1max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T]; scope2 = z2max / Sqrt[(Σ1+Σ2+Σinf1+Σinf2)/4 T];
  M1 = -0.075; M2 = -0.075; ρm1 = 0.0; ρm2 = 0;
  M = ( M1                ρm1 Sqrt[M1 M2]
        ρm2 Sqrt[M1 M2]   M2              );
  ΣinfM = ( Σinf1                        Sqrt[Σinf1 Σinf2] ρinf12
            Sqrt[Σinf1 Σinf2] ρinf12     Σinf2                    );
  Q = CholeskyDecomposition[-((M.ΣinfM + ΣinfM.Transpose[M])/2)] / β;
  BiHestonMonteCarloSmile[StrikeList, M, Q, ΣinfM, {ρ1, ρ2},
   {Σ1, Sqrt[Σ1 Σ2] ρ12, Σ2}, {S1, S2}, β, TimeStepsNb, dt, nbSample, printflag]]]
```

```
β=2.25
{10.031, {0.0246112, 0.0197767, 0.0154796, 0.0135241, 0.0117236, 0.01044, 0.00936011,
  0.0083956, 0.00748358, 0.00666864, 0.00587163, 0.00522334, 0.004184, 0.003284}}
```

```
Timing[
 Module[{S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.05, M2 = -0.05, θ1 = 0.03, θ2 = 0.041,
   ρs = 0.6, ρsinf = 0.8, ρm1, ρm2, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
```

```
period1 = scope1;
period1n = scope1;
ϵ1 = 0.00001;
Nb2 = 10;
LegendreCoef2 = LegendreCoeffs[Nb2];
period2 = scope2;
period2n = scope2 / 10;
ϵ2 = 0.00001;
```

$$\Sigma inf = \begin{pmatrix} \theta 1 & \sqrt{\theta 1\,\theta 2}\ \rho sinf \\ \sqrt{\theta 1\,\theta 2}\ \rho sinf & \theta 2 \end{pmatrix};\ \Sigma = \begin{pmatrix} \Sigma 1 & \sqrt{\Sigma 1\,\Sigma 2}\ \rho s \\ \sqrt{\Sigma 1\,\Sigma 2}\ \rho s & \Sigma 2 \end{pmatrix};$$

```
strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0,
   0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02};
ρm1 = -0.5; ρm2 = -0.5;
```

$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\,M2} \\ \rho m2\ \sqrt{M1\,M2} & M2 \end{pmatrix};$$

```
smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter000 = Interpolation[smile000, InterpolationOrder → 2];
ρm1 = -0.5; ρm2 = 0;
```

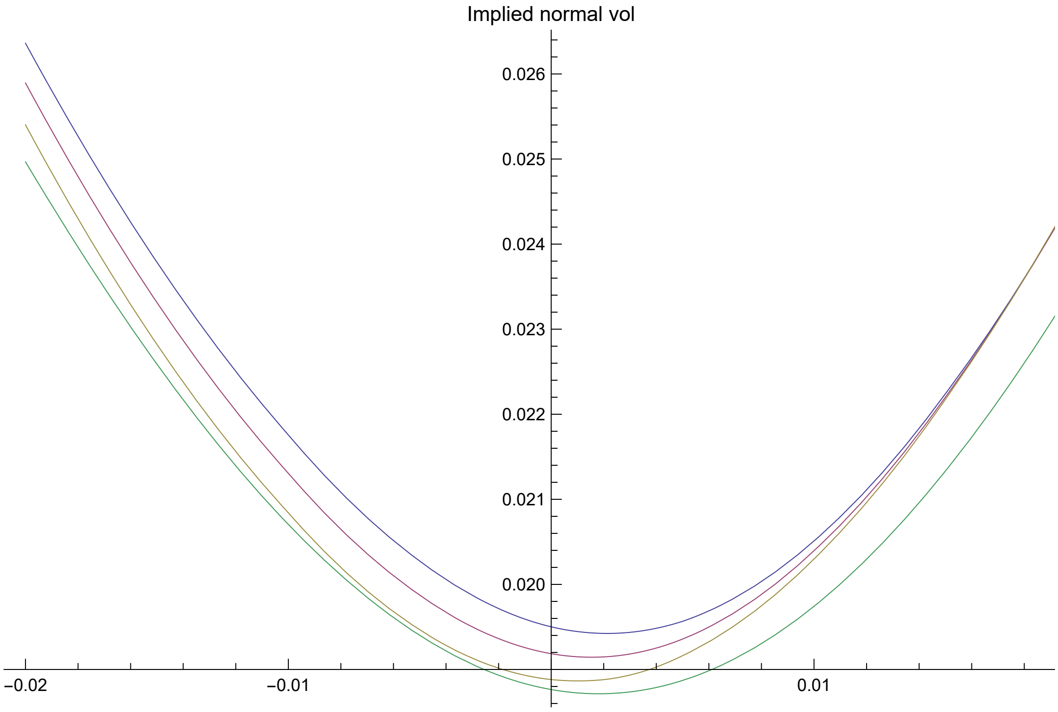
$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\,M2} \\ \rho m2\ \sqrt{M1\,M2} & M2 \end{pmatrix};$$

```
smile001 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter001 = Interpolation[smile001, InterpolationOrder → 2];
ρm1 = -0.5; ρm2 = 0.5;
```

$$M = \begin{pmatrix} M1 & \rho m1\ \sqrt{M1\,M2} \\ \rho m2\ \sqrt{M1\,M2} & M2 \end{pmatrix};$$

```
smile002 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter002 = Interpolation[smile002, InterpolationOrder → 2];
vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
smile2 =
 Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
     S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
inter2 = Interpolation[smile2];
Plot[{inter000[x], inter001[x], inter002[x], inter2[x]},
 {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
 PlotLegend → {"biheston --", "biheston -0", "biheston -+", "bilog"},
 LegendPosition → {1, 0}]
]]
```

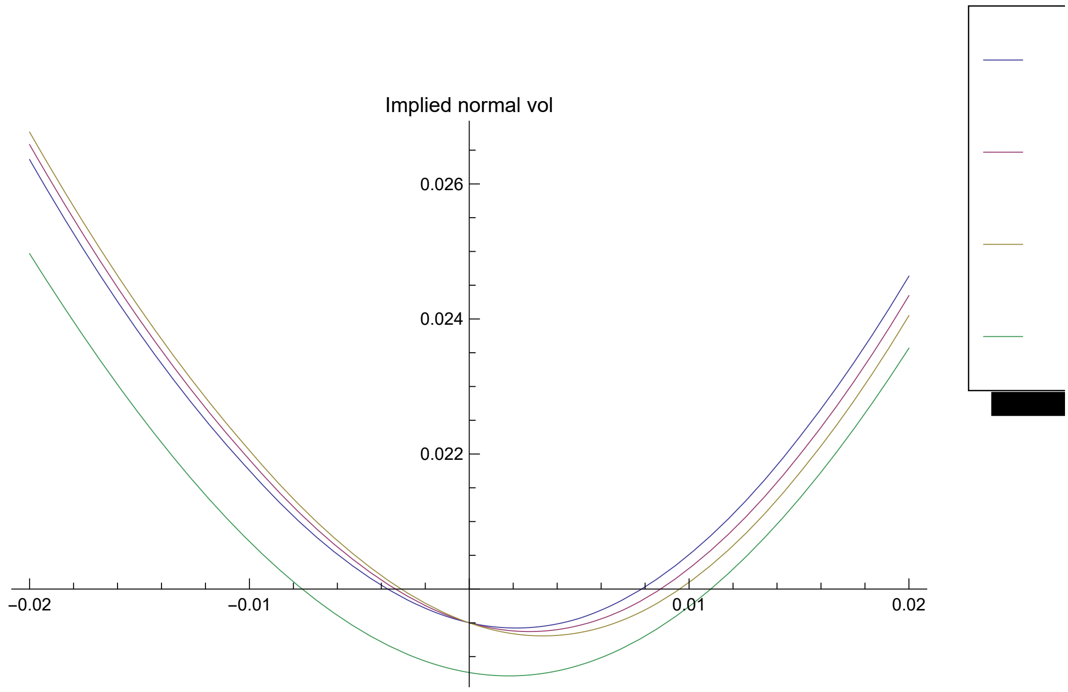
{55.099,




```
Timing[
 Module[{S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.05, M2 = -0.05, θ1 = 0.03, θ2 = 0.041,
   ρs = 0.6, ρsinf = 0.8, ρm1, ρm2, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
```

```
Σinf = ( θ1                 √(θ1 θ2) ρsinf
         √(θ1 θ2) ρsinf     θ2             ); Σ = ( Σ1              √(Σ1 Σ2) ρs
                                                    √(Σ1 Σ2) ρs     Σ2          );
strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0,
   0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02};
ρm1 = -0.5; ρm2 = -0.5;
M = ( M1               ρm1 √(M1 M2)
      ρm2 √(M1 M2)     M2           );
smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter000 = Interpolation[smile000, InterpolationOrder → 2];
ρm1 = 0.; ρm2 = -0.5;
M = ( M1               ρm1 √(M1 M2)
      ρm2 √(M1 M2)     M2           );
smile001 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter001 = Interpolation[smile001, InterpolationOrder → 2];
ρm1 = 0.5; ρm2 = -0.5;
M = ( M1               ρm1 √(M1 M2)
      ρm2 √(M1 M2)     M2           );
smile002 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter002 = Interpolation[smile002, InterpolationOrder → 2];
vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
smile2 =
 Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
     S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
inter2 = Interpolation[smile2];
Plot[{inter000[x], inter001[x], inter002[x], inter2[x]},
 {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
 PlotLegend → {"biheston --", "biheston 0-", "biheston +-", "bilog"},
 LegendPosition → {1, 0}]
]]
```

{56.254,




```
Timing[
 Module[{S1 = 0.05, S2 = 0.05, K = 0.00001, M1 = -0.2, M2 = -0.2, θ1 = 0.03, θ2 = 0.041,
   ρs = 0.6, ρsinf = 0.8, ρm1, ρm2, ρ1 = 0.5, ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5,
   τ = 5, λ1 = 1.1, λ2 = 1.2, scope1, scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n,
   period1, period1n, ϵ1, ν1 = 0.01, ν2 = 0.01, Lcoefs = LegendreCoeffs[40],
   Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 0, M, Σinf, Σ},
```

$$\text{scope1} = \frac{2}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};$$

$$\text{scope2} = \frac{3}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};$$

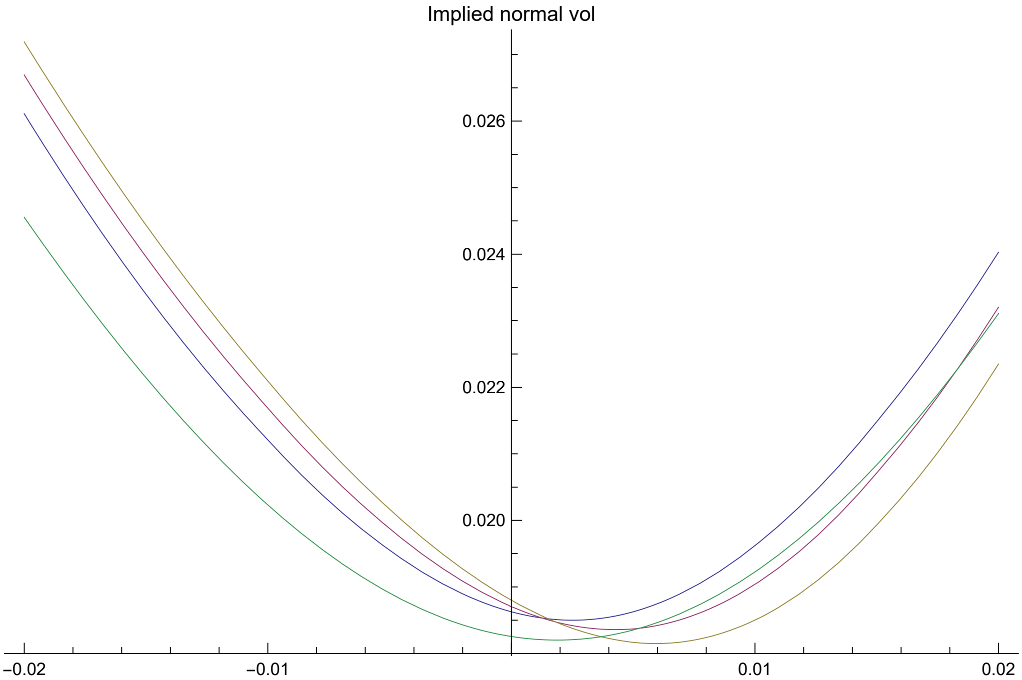
```
  Nb1 = 12;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
```

$$\Sigma\text{inf} = \begin{pmatrix} \theta1 & \sqrt{\theta1\,\theta2}\;\rho\text{sinf} \\ \sqrt{\theta1\,\theta2}\;\rho\text{sinf} & \theta2 \end{pmatrix}; \; \Sigma = \begin{pmatrix} \Sigma1 & \sqrt{\Sigma1\,\Sigma2}\;\rho\text{s} \\ \sqrt{\Sigma1\,\Sigma2}\;\rho\text{s} & \Sigma2 \end{pmatrix};$$

```
  strikes = {-0.02, -0.01, -0.005, -0.002, -0.001, -0.0005, -0.0002, -0.0001, 0,
    0.0001, 0.0002, 0.0005, 0.001, 0.002, 0.005, 0.0075, 0.01, 0.015, 0.02};
```

```
ρm1 = -0.5; ρm2 = -0.5;
M = ( M1              ρm1 Sqrt[M1 M2]
      ρm2 Sqrt[M1 M2] M2             );
smile000 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter000 = Interpolation[smile000, InterpolationOrder → 2];
ρm1 = 0.; ρm2 = -0.5;
M = ( M1              ρm1 Sqrt[M1 M2]
      ρm2 Sqrt[M1 M2] M2             );
smile001 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter001 = Interpolation[smile001, InterpolationOrder → 2];
ρm1 = 0.5; ρm2 = -0.5;
M = ( M1              ρm1 Sqrt[M1 M2]
      ρm2 Sqrt[M1 M2] M2             );
smile002 = Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ,
     NewSuperBiHestonVanilla[strikes[[i]], τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
      {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
      {LegendreCoef2, period2, period2n, ϵ2}, printflag]]}, {i, 1, Length[strikes]}];
inter002 = Interpolation[smile002, InterpolationOrder → 2];
vol1 = ImpVolHeston2[S1, S1, τ, Σ1, θ1, ρ1, -M1, ν1, Lcoefs];
vol2 = ImpVolHeston2[S2, S2, τ, Σ2, θ2, ρ2, -M2, ν2, Lcoefs]; ρsmod = ρs;
smile2 =
 Table[{strikes[[i]], NormalImplicitVol[S1 - S2, strikes[[i]], τ, LogNormalSpreadOption[
     S1, S2, vol1, vol2, ρsmod, strikes[[i]], τ]]}, {i, 1, Length[strikes]}];
inter2 = Interpolation[smile2];
Plot[{inter000[x], inter001[x], inter002[x], inter2[x]},
 {x, strikes[[1]], Last[strikes]}, PlotLabel → "Implied normal vol",
 PlotLegend → {"biheston --", "biheston 0-", "biheston +-", "bilog"},
 LegendPosition → {1, 0}]
]]
```

{65.115,



```
Timing[Module[{S1 = 0.05, S2 = 0.005, K = 0.0000, M1 = -0.01, M2 = -0.02,
    θ1 = 0.03, θ2 = 0.041, ρs = 0.6, ρsinf = 0.8, ρm1 = 0.3, ρm2 = -0.3, ρ1 = 0.5,
    ρ2 = 0.8, Σ1 = 0.04, Σ2 = 0.05, β = 5, τ = 5, λ1 = 1.1, λ2 = 1.2, scope1,
    scope2, Nb1, LegendreCoef1, Nb1n, LegendreCoef1n, period1, period1n, ϵ1,
    Nb2, LegendreCoef2, period2, period2n, ϵ2, printflag = 2, M, Σinf, Σ},
  scope1 = 2 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  scope2 = 3 / Sqrt[(Σ1+Σ2+θ1+θ2)/4 τ];
  Nb1 = 20;
  LegendreCoef1 = LegendreCoeffs[Nb1];
  Nb1n = 8;
  LegendreCoef1n = LegendreCoeffs[Nb1n];
  period1 = scope1;
  period1n = scope1;
  ϵ1 = 0.00001;
  Nb2 = 10;
  LegendreCoef2 = LegendreCoeffs[Nb2];
  period2 = scope2;
  period2n = scope2 / 10;
  ϵ2 = 0.00001;
  M = {{M1, ρm1 Sqrt[M1 M2]}, {ρm2 Sqrt[M1 M2], M2}};
  Σinf = {{θ1, Sqrt[θ1 θ2] ρsinf}, {Sqrt[θ1 θ2] ρsinf, θ2}};
  Σ = {{Σ1, Sqrt[Σ1 Σ2] ρs}, {Sqrt[Σ1 Σ2] ρs, Σ2}};
  {NewSuperBiHestonVanilla[K, τ, M, Σinf, {ρ1, ρ2}, Σ, {S1, S2}, β, λ1, λ2,
    {LegendreCoef1, LegendreCoef1n, period1, period1n, ϵ1},
    {LegendreCoef2, period2, period2n, ϵ2}, printflag]}]]
```

$$\text{scope1} = \frac{2}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};\qquad \text{scope2} = \frac{3}{\sqrt{\frac{\Sigma1+\Sigma2+\theta1+\theta2}{4}\,\tau}};$$

$$M = \begin{pmatrix} M1 & \rho m1\sqrt{M1\,M2} \\ \rho m2\sqrt{M1\,M2} & M2 \end{pmatrix};\quad \Sigma inf = \begin{pmatrix} \theta1 & \sqrt{\theta1\,\theta2}\,\rho sinf \\ \sqrt{\theta1\,\theta2}\,\rho sinf & \theta2 \end{pmatrix};\quad \Sigma = \begin{pmatrix} \Sigma1 & \sqrt{\Sigma1\,\Sigma2}\,\rho s \\ \sqrt{\Sigma1\,\Sigma2}\,\rho s & \Sigma2 \end{pmatrix};$$

```
{NbLegendreCoef1,NbLegendreCoef1n,period1,period1n,∈1}={20, 8, 4.45823, 4.45823, 0.00001}
{NbLegendreCoef2,period2,period2n,∈2}={10, 6.68734, 0.668734, 0.00001}
Integ_2={2, 2.31505}
Integ_2={2, 1.77644}
Integ_2={2, -0.0614284}
Integ_2={3, -0.69629}
Integ_2={3, 0.169008}
Integ_2={3, 0.0136468}
Integ_2={3, -0.0279515}
Integ_2={3, 0.0118366}
Integ_2={2, 0.00183024}
Integ_2={3, -0.00205849}
Integ_2={3, -0.00188577}
Integ_2={3, 0.00029081}
Integ_2={3, 0.000192339}
Integ_2={1, -0.0001716}
Integ_2={3, -7.4355×10^-6}
Integ_1={5, 0.895297}
{2.403, {0.0453563}}
```