

Rapport de Stage de Fin d'Etude

Etude et Calibration d'un modèle à sauts Applications aux fonds

M'RAD Mohamed Kamel Eddine

12 septembre 2005

Préface

Ce travail s'inscrit dans le cadre de mon stage de Master Probabilités et finance de Paris VI au sein de IXIS CIB. Partant d'une étude théorique d'un modèle à sauts assez général proposée par Sepp [1], on s'est proposé de modéliser la dynamique des fonds observée sur le marché. Les données disponibles étant des rendements et non de la volatilité, la calibration du modèle en question posait un vrai problème dont les difficultés étaient principalement numériques. En effet, l'objectif final était de mettre en pratique ce modèle sur la base de données **Pertrac** avec un algorithme dont le coût est assez faible et l'erreur de calibration optimale.

Remerciements

Je tiens à remercier mon directeur de stage à IXIS CIB; Mr Olivier Croissant pour sa disponibilité, sa bonne humeur et surtout pour la confiance dont il a témoigné à mon égard, ainsi que mon directeur de stage à l'Ecole Polytechnique, Mme Nicole El Karoui pour son écoute et son suivi du déroulement de mon stage de fin d'étude.

Enfin, je ne pourrai pas oublier l'ambiance agréable dans laquelle j'ai travaillé au sein de l'équipe.

Table des matières

1	Introduction	5
1.1	Définition d'un <i>fund</i>	5
1.2	Motivation	5
1.3	Approche	6
2	Etude théorique	7
2.1	Quelques définitions utiles	7
2.1.1	Processus de Markov	7
2.1.2	Options Européennes	7
2.2	Théorème de Feynman-Kac	8
2.3	La transformée de Fourier	8
2.3.1	Modèle formel et EDP	9
2.3.2	EDP et transformée de Fourier	10
2.3.3	La transformée Fondamentale	13
3	Le modèle proposé	15
3.1	Modèle de Black-Scholes	15
3.2	Modèle à Saut	16
3.2.1	Saut de taille Log-Normale	17
3.2.2	Saut de taille Double Exponentielle	18
3.2.3	Saut de taille Double Log-Normale	18
3.3	Modèle à volatilité stochastique	18
3.4	Modèle à volatilité stochastique plus Sauts	20
3.5	Modèle à saut avec intensité stochastique et volatilité constante	20
3.6	Le modèle général	23
4	Etude Pratique	27
4.1	Intégration numérique	28
4.2	Calibration du modèle	30
4.2.1	nag-opt-nlin-lsq	30
4.2.2	specification	30

4.2.3	Description	31
4.3	Résultats numériques et commentaires	36
4.3.1	Existence de plusieurs minimums locaux	37
4.3.2	Amélioration du temps de calcul	45
5	Appendice	47
5.1	Calcul de la transformée du Saut Λ	47
5.1.1	La taille du saut suit une distribution Log-Normale	47
5.1.2	La taille du saut suit une distribution Double Exponentielle	47
5.1.3	La taille du saut suit une distribution Double Log-Normale	48
5.2	Démonstration de la proposition 3.4.1	48
5.3	Démonstration de la proposition 3.6.1	50
	Références	50

Chapitre 1

Introduction

1.1 Définition d'un *fund*

Un fond, ou encore *fund*, est une collecte de capitaux auprès de différentes sources de financement. Cet argent sera ensuite utilisé dans des opérations de capital-investissement. On parle alors de fonds d'investissement dont la gestion est assurée par des gestionnaires de fonds : le fond **Innovacom 4** est par exemple géré par la société **Innovacom gestion**. Outre ces fonds privés, il existe en France, des dispositifs particuliers gérés par des institutions financières et bénéficiant de dispositions fiscales spécifiques : les **Fonds Communs de Placement à Risque (FCPR)** et les **Fonds Communs de Placement dans l'Innovation (FCPI)**.

1.2 Motivation

L'objectif de cette étude est de mieux comprendre le ou les comportements des *funds* observés sur le marché financier et jusque là assez mal modéliser. Par une étude théorique reprenant des modèles assez classiques, Artur Sepp [1] a étudié un modèle à sauts assez général qu'on s'est proposé d'utiliser pour répliquer la dynamique des funds. Une étude précédente sur les fonds a laissé entendre qu'on peut classer ces derniers suivant leur rendements et qu'on peut modéliser leur dynamique par un simple modèle à volatilité stochastique, ce qui va être démenti par notre travail par la suite. En effet, en observant l'historique des rendements d'un fond et surtout la densité empirique extraite, on se rend compte que

- un modèle à volatilité stochastique ne tient pas compte des sauts observés sur l'historique des *funds* sur le marché.

- des paramètres supplémentaires, notamment sur la volatilité, sont nécessaires pour pouvoir répliquer la densité empirique observée.

1.3 Approche

Ce travail admet deux composantes;

- Etude théorique. Le chapitre 2 présente le coeur de la discussion théorique des différents modèles déjà proposés pour les fonds. Après un bref rappel de quelques notions mathématiques telles que la transformée de Fourier, on établit les formules fermées pour un modèle général formel de même type que celui proposé par [1].

Le chapitre 3 est une comparaison du modèle étudié avec ceux déjà proposés.

- Implémentation et calibration. Pour mettre en pratique le modèle proposé, la première étape consiste à son implémentation.

Vient ensuite l'étape cruciale de ce travail; la calibration du modèle au marché financier où, à partir de la base de données **Pertrac**, on va pouvoir accéder aux rendements historiques des fonds. Par suite, on a utilisé **Nag**, une librairie performante pour optimiser les erreurs numériques et le coût de l'algorithme en terme de temps de calcul.

Chapitre 2

Etude théorique

2.1 Quelques définitions utiles

2.1.1 Processus de Markov

Soit \mathbb{E} un ensemble fini ou dénombrable, muni de la topologie discrète et de la tribu $\mathbb{P}(\mathbb{E})$ des parties de \mathbb{E} .

Soit $(X_t)_{t \geq 0}$ un processus à valeurs dans \mathbb{E} , à trajectoires continues à droite avec limites à gauche en tout point. On note $F_t^X = \sigma(X_u, u \leq t)$ la filtration naturelle du processus X .

Définition 2.1.1 *on dit que X_t est un processus de Markov à valeurs dans \mathbb{E} si, pour tout $s < t$ et pour toute fonction boorélienne bornée f de \mathbb{E} dans \mathbb{R}*

$$\mathbb{E}(f(X_t)/F_s^X) = \mathbb{E}(f(X_t)/X_s).$$

*Si de plus la loi de X_t sachant X_s ne dépend que $t - s$, on dit que ce processus est **homogène**. Dans ce cas la loi de X_t sachant X_s est donnée par*

$$P(X_t = x/X_s = y) = Q_{t-s}(x, y),$$

où $Q_t(x, y)$ est une famille de noyaux markoviens.

Le lemme suivant est une caractérisation des processus de Markov

Lemme 2.1.1 *Soit $(X_t)_{t \geq 0}$ un processus continue à droite limité à gauche, à valeurs dans \mathbb{E} . Pour que X_t soit un processus de Markov, il suffit que, pour tout $t_0 < t_1 < \dots < t_n < t$, la loi de X_t sachant $(X_{t_0}, X_{t_1}, \dots, X_{t_n})$ soit celle de X_t sachant X_{t_n} .*

2.1.2 Options Européennes

Une option est un contrat financier qui donne le droit mais non l'obligation d'exécuter ce droit en une date future (maturité). Le *payoff* pour une

option vanille (*Call* ou *Put* européen) a maturité T est donné par:

$$F = \max(0, \varepsilon[S_T - K]),$$

où S est le prix de l'actif, K est le *Strike*, T est la maturité ou encore date d'expiration du contrat et $\varepsilon = \pm 1$, 1 pour un *Call* et -1 pour un *Put*.

2.2 Théorème de Feynman-Kac

C'est un outil mathématique adapté au *pricing* des options européennes. Pour plus de détails, voir Duffie (2000).

Théorème 2.2.1 (Feynman-Kac) *Pour tout vecteur $X \in \mathbb{R}^d$ caractérisé par la dynamique suivante:*

$$dX_t = \mu(t, X_t)dt + \sigma(t, X_t)dW_t + J(X_t)dN_t(\lambda(X_t)),$$

où les fonctionnelles μ et σ vérifient les conditions de lipshitz et de croissance, W_t et N_t sont des vecteurs de \mathbb{R}^d respectivement un mouvement brownien et un processus de Poisson. Alors sous ces hypothèses, pour $f, g \in \mathbb{R}$, l'expression:

$$F(X_t) = E_t[e^{-\int_t^T g(X_s)ds} f(X_T)],$$

est solution de l'équation

$$\frac{\partial F(t, X_t)}{\partial t} + \mathfrak{D}F(t, X_t) = g(X_t)F(t, X_t)$$

avec condition finale $F(X_T) = f(X_T)$ et \mathfrak{D} est l'opérateur différentiel défini par

$$\mathfrak{D}F(X_t) = \sum_{i=1}^d \mu_i(t, X) \frac{\partial F(X)}{\partial X_i} + \frac{1}{2} \sum_{i,j=1}^d \Lambda_{ij}(t, X) \frac{\partial^2 F(t, X)}{\partial X_i \partial X_j} + \sum_{i=1}^d \lambda_i(X_t) \mathbb{E}_t[F(X_i + J_i) - F(X_i)]$$

où $\Lambda(t, X) = \sigma(t, X)\sigma^T(t, X)$.

2.3 La transformée de Fourier

C'est l'outil mathématique le plus important dans la discussion théoriques des différents modèles proposés pour les *funds*. Il nous permettra, comme on

le verra dans la suite, de simplifier l'étude théorique d'un modèle formel assez général.

Définition 2.3.1 *On se donne une fonctionnelle f , sa transformée de Fourier notée par \widehat{f} est donnée par:*

$$\widehat{f}(z) = \int_{-\infty}^{+\infty} e^{izx} f(x) dx$$

et son inverse par

$$f(x) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{-izx} \widehat{f}(z) dz.$$

2.3.1 Modèle formel et EDP

On considère un modèle à sauts assez général caractérisé par la dynamique suivante:

$$\begin{cases} dS_t/S_t &= (r_t - d_t - \lambda_t m) S_t dt + \sqrt{V_t} S_t dW_t^s + (e^{J_t} - 1) dN_t(\lambda_t) \\ dV_t &= a(V_t) dt + b(V_t) dW_t^v \\ d\lambda_t &= c(\lambda_t) dt + d(\lambda_t) dW_t^\lambda \end{cases}$$

où

- S_t est le cours de l'actif, V_t sa variance,
- W_t^s et W_t^v sont deux browniens corrélés, la corrélation est noté par $\rho(V_t)$,
- W_t^λ est un brownien indépendant des deux premiers,
- $N_t(\lambda_t)$ est un processus de Poisson dont l'intensité ici notée par λ_t est stochastique.
- J_t représente la taille du saut, c'est une variable aléatoire où $m := E_t^Q(e^J - 1)$. On notera par $\phi(x)$ sa fonction de densité qu'on suppose vérifier la condition suivante:

$$E_t^Q(\phi(J)) < \infty,$$

Les coefficients $a(V_t)$, $b(V_t)$, $c(\lambda_t)$, $d(\lambda_t)$ et $\phi(x)$ seront choisis dans la suite de façon à pouvoir établir des formules fermées à partir de l'EDP de *pricing*. Ce modèle formel est assez général pour modéliser la dynamique des fonds sur le marché.

Corollaire 2.3.1 *En gardant les notations précédentes, et par le théorème de Feynman - Kac, le prix d'une option européenne de Strike K et de*

maturité T , vérifie l'EDP suivante:

$$\begin{aligned} \frac{\partial F}{\partial t} + (r - d - \lambda m)S \frac{\partial F}{\partial S} + \frac{1}{2}VS^2 \frac{\partial^2 F}{\partial S^2} + a(V) \frac{\partial F}{\partial V} + \frac{1}{2}b(V) \frac{\partial^2 F}{\partial V^2} \\ + \rho b(V)\sqrt{V}S \frac{\partial^2 F}{\partial S \partial V} + c(\lambda) \frac{\partial F}{\partial \lambda} + \frac{1}{2}d^2(\lambda) \frac{\partial^2 F}{\partial \lambda^2} \\ + \lambda \mathbb{E}_t^Q(F(e^J S) - F(S)) - rF = 0 \end{aligned} \quad (2.1)$$

avec la condition terminale

$$F(S_t, V_t, \lambda_t, t = T) = g(S_T, K).$$

Pour simplifier l'étude de cette EDP, on utilise le changement de variable $x = \ln(S)$ et on définit

$$f(x, V, \lambda, t) = F(S, V, \lambda, t)$$

vérifiant ainsi la nouvelle EDP:

$$\begin{aligned} \frac{\partial f}{\partial t} + (r - d - \frac{1}{2}V - \lambda m) \frac{\partial f}{\partial S} + \frac{1}{2}V \frac{\partial^2 f}{\partial x^2} + a(V) \frac{\partial f}{\partial V} + \frac{1}{2}b(V) \frac{\partial^2 f}{\partial V^2} \\ + \rho b(V)\sqrt{V} \frac{\partial^2 f}{\partial S \partial V} + c(\lambda) \frac{\partial f}{\partial \lambda} + \frac{1}{2}d^2(\lambda) \frac{\partial^2 f}{\partial \lambda^2} \\ + \lambda \mathbb{E}_t^Q(f(x + J) - f(x)) - rf = 0. \end{aligned} \quad (2.2)$$

La condition finale devient

$$f(x, V, \lambda, t = T) = g(e^x, K)$$

2.3.2 EDP et transformée de Fourier

On applique la transformée de Fourier à $f(x, V, \lambda, t)$ par rapport à sa première variable x

$$\hat{f}(z, V, \lambda, t) = \int_{-\infty}^{+\infty} e^{izx} f(x, V, \lambda, t) dx, z \in \mathbb{C}.$$

D'une manière générale, pour que cette intégrale existe et soit bien définie, $\mathcal{I}m(z)$ doit se trouver entre deux valeurs: $\alpha < \mathcal{I}m z < \beta$, où $\mathcal{I}m(z)$ désigne la partie imaginaire de z .

Une fois cette condition respectée, il ne nous reste qu'à calculer la fonctionnelle f et essayer de trouver une formule fermée si c'est possible .

Remarquons que dans le cas du *Call*, le *payoff* $\varphi = (S_T - K)_+$ peut être décomposé sous la forme suivante

$$\varphi = S_T - \min(S_T, K)$$

et donc il suffit de trouver une solution pour le cas particulier

$$f(x, V, \lambda, t) = \min(e^x, K)$$

pour trouver une formule fermée pour le cas général.

Proposition 2.3.1 *La transformée de Fourier du payoff $f(x, V, \lambda, t) = \min(e^x, K)$ pour $0 < \text{Im}z < 1$ est donnée par:*

$$\hat{f}(z, V, \lambda, t, T) = \frac{K^{1+iz}}{z^2 - iz}.$$

Preuve

On a

$$f = \min(e^x, K)$$

donc

$$\begin{aligned} \hat{f}(z, V, \lambda, t, T) &= \int_{-\infty}^{+\infty} e^{izx} \min(e^x, K) dx = \lim_{U \rightarrow \infty} \int_{-U}^{\ln K} e^{(iz+1)x} dx + \lim_{U \rightarrow -\infty} \int_{\ln K}^U e^{izx} dx \\ &= \lim_{U \rightarrow \infty} \frac{e^{(iz+1)x}}{iz+1} \Big|_{x=-U}^{x=\ln(K)} + K \lim_{U \rightarrow \infty} \frac{e^{izx}}{iz} \Big|_{x=\ln(K)}^{x=U} \\ &= \frac{e^{(iz+1)\ln(K)}}{iz+1} - \lim_{U \rightarrow \infty} \frac{e^{-(iz+1)U}}{iz+1} + K \left(\lim_{U \rightarrow \infty} \frac{e^{izU}}{iz} - \frac{e^{iz\ln(K)}}{iz} \right) \end{aligned}$$

Or, on sait que pour $0 < \text{Im}(z) < 1$, on a $U \rightarrow +\infty$ donc

$$\frac{e^{-(iz+1)U}}{iz+1} \rightarrow 0.$$

De même, si $U \rightarrow -\infty$, on a

$$\frac{e^{izU}}{iz} \rightarrow 0.$$

Finalement on obtient que

$$\hat{f}(z, V, \lambda, t, T) = \frac{K^{iz+1}}{iz+1} - \frac{K^{iz+1}}{iz} = \frac{K^{iz+1}}{-iz+z^2}.$$

On va maintenant appliquer la transformée de Fourier à l'EDP (2.2). Sous condition d'existence vérifiée sur Imz , la nouvelle EDP vérifiée par \hat{f} est la suivante:

$$\begin{aligned} & \frac{\partial \hat{f}}{\partial t} + (-r - iz(r - d - m\lambda))\hat{f} - \frac{1}{2}(z^2 - iz)V\hat{f} \\ & + (a(V) - iz\rho(V)b(V)\sqrt{V})\frac{\partial \hat{f}}{\partial V} + \frac{1}{2}b^2(V)\frac{\partial^2 \hat{f}}{\partial V^2} \\ & + c(\lambda)\frac{\partial \hat{f}}{\partial \lambda} + \frac{1}{2}d^2(V)\frac{\partial^2 \hat{f}}{\partial \lambda^2} + \lambda\mathbb{E}(\hat{f}(e^{-izJ} - 1)) = 0 \end{aligned}$$

Peuve:

Remarquons que

$$\int_{-\infty}^{+\infty} e^{izx} \frac{\partial f}{\partial t} dx = \frac{\partial \hat{f}}{\partial t}$$

En intégrant par partie, on écrit:

$$\begin{aligned} & \int_{-\infty}^{+\infty} e^{izx} \frac{\partial^2 f}{\partial x^2} dx = e^{izx} f \Big|_{x=-\infty}^{x=+\infty} - iz \int_{-\infty}^{+\infty} e^{izx} \frac{\partial f}{\partial x} dx \\ & = -iz e^{izx} f \Big|_{x=-\infty}^{x=+\infty} - z^2 \int_{-\infty}^{+\infty} e^{izx} f dx = -z^2 \int_{-\infty}^{+\infty} e^{izx} f dx \\ & = -z^2 \hat{f} \end{aligned}$$

En plus:

$$\int_{-\infty}^{+\infty} e^{izx} \frac{\partial f}{\partial x} dx = e^{izx} f \Big|_{x=-\infty}^{x=+\infty} - iz \int_{-\infty}^{+\infty} e^{izx} f dx = -iz \hat{f}.$$

D'autre part, si on utilise le changement de variable $y = x + J$, on écrit alors:

$$\begin{aligned} & \int_{-\infty}^{+\infty} e^{izx} f(x + J, V, \lambda, t) dx = \int_{-\infty}^{+\infty} e^{izx - izJ} f(x, V, \lambda, t) dx \\ & = e^{-izJ} \int_{-\infty}^{+\infty} e^{izy} f(x, V, \lambda, t) dx = e^{-izJ} \hat{f}(z, V, \lambda, t) \end{aligned}$$

Pour appliquer la Transformée de Fourier de l'EDP (2.2), il suffit de la multiplier par e^{izx} et puis intégrer. On obtient alors l'équation:

$$\begin{aligned} & \int_{-\infty}^{+\infty} e^{izx} \frac{\partial f}{\partial t} dx + \int_{-\infty}^{+\infty} e^{izx} [(r - d - \frac{1}{2}V - \lambda m) \frac{\partial f}{\partial S} + \frac{1}{2}V \frac{\partial^2 f}{\partial x^2} \\ & + a(V) \frac{\partial f}{\partial V} + \frac{1}{2}b(V) \frac{\partial^2 f}{\partial V^2} + \rho b(V) \sqrt{V} \frac{\partial^2 f}{\partial S \partial V} \\ & + c(\lambda) \frac{\partial f}{\partial \lambda} + \frac{1}{2}d^2(\lambda) \frac{\partial^2 f}{\partial \lambda^2} + \lambda \mathbb{E}_t^Q(f(x + J) - f(x)) - rf] = 0. \end{aligned}$$

En utilisant les calculs et les égalités précédentes, on obtient finalement:

$$\begin{aligned} & \frac{\partial \hat{f}}{\partial t} + (-r - iz(r - d - m\lambda))\hat{f} - \frac{1}{2}(z^2 - iz)V\hat{f} \\ & + (a(V) - iz\rho(V)b(V)\sqrt{V})\frac{\partial \hat{f}}{\partial V} + \frac{1}{2}b^2(V)\frac{\partial^2 \hat{f}}{\partial V^2} \\ & + c(\lambda)\frac{\partial \hat{f}}{\partial \lambda} + \frac{1}{2}d^2(V)\frac{\partial^2 \hat{f}}{\partial \lambda^2} + \lambda\mathbb{E}(\hat{f}(e^{-izJ} - 1)) = 0 \end{aligned}$$

2.3.3 La transformée Fondamentale

On introduit une nouvelle variable: $\tau = T - t$ et on pose

$$\hat{h}(z, V, \lambda, \tau) = \hat{f}(z, V, \lambda, t)$$

pour la réinjecter dans L'EDP vérifier par \hat{f} dans le but de montrer que \hat{h} telle qu'elle est définie vérifie:

$$\begin{aligned} & \frac{\partial \hat{h}}{\partial \tau} + (r + iz(r - d - m\lambda))\hat{h} + \frac{1}{2}(z^2 - iz)V\hat{h} \\ & - (a(V) - iz\rho(V)b(V)\sqrt{V})\frac{\partial \hat{h}}{\partial V} - \frac{1}{2}b^2(V)\frac{\partial^2 \hat{h}}{\partial V^2} \\ & - c(\lambda)\frac{\partial \hat{h}}{\partial \lambda} - \frac{1}{2}d^2(V)\frac{\partial^2 \hat{h}}{\partial \lambda^2} - \lambda\mathbb{E}(\hat{f}(e^{-izJ} - 1)) = 0 \end{aligned}$$

avec la condition finale $\hat{h}(z, V, \lambda, \tau = 0) = 1$.

Définition 2.3.2 La solution $\hat{h}(z, V, \lambda, \tau)$ de l'EDP (2.9) vérifiant

$$\hat{h}(z, V, \lambda, \tau = 0) = 1$$

est appelée transformée fondamentale. Etant donnée cette transformée fondamentale, et en appliquant la transformée de Fourier inverse, la solution associée au payoff $f(x, V, \lambda, T)$ est donnée par:

$$f(x, V, \lambda, t) = \frac{1}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} e^{-izx} \frac{K^{1+iz}}{z^2 - iz} \hat{h}(z, V, \lambda, \tau) dz$$

Introduisons maintenant une nouvelle variable $Y = \ln(S/K)$. On aboutit alors au théorème suivant:

Théorème 2.3.1 La solution associée au payoff $f(x, V, \lambda, \tau) = \min(e^x, K)$ est donnée par:

$$f(x, V, \lambda, \tau) = \frac{1}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} e^{-izY} \frac{\hat{h}(z, V, \lambda, \tau)}{z^2 - iz} dz$$

Le prix du *Call* européen s'écrit donc facilement

$$C(S, V, \lambda, \tau) = S e^{-d_t \tau} - f(x, V, \lambda, \tau)$$

Pour le *Put* européen, on la formule suivante:

$$P(S, V, \lambda, \tau) = K e^{-r_t \tau} - f(x, V, \lambda, \tau)$$

Ainsi le calcul du prix d'une option européenne, que ce soit un *Call* ou un *Put*, est réduit à la simple évaluation de l'intégrale (2.1).

On sait que d'après les calculs précédents que ces formules intégrales ne sont définies que pour $0 < \mathcal{I}m(z) < 1$. On va donc choisir $\mathcal{I}m(z) = \frac{1}{2}$ entre les deux pôles (zéros) de $z^2 - iz$, la formule (2.2) se réécrit alors:

$$f(x, V, \lambda, \tau) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} e^{(-ik + \frac{1}{2})Y} \frac{\hat{h}(k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}} dk \quad (2.3)$$

Remarquons que dans cette équation le terme à gauche c'est f est dans \mathbb{R} donc forcément le second l'est aussi. En effet, on peut écrire:

$$\hat{h}(k + \frac{1}{2}i) = \int_{-\infty}^{+\infty} e^{-i(k + \frac{1}{2}i)x} h(x) dx.$$

Et donc

$$\bar{\hat{h}}(k + \frac{1}{2}i) = \int_{-\infty}^{+\infty} e^{-i(k + \frac{1}{2}i)x} h(x) dx = \int_{-\infty}^{+\infty} e^{ik + \frac{1}{2}x} h(x) dx = \hat{h}(-k + \frac{1}{2}i)$$

D'une autre part, on décompose l'intégrale (2.5) en deux parties comme suit

$$f(x, V, \lambda, \tau) = \frac{1}{2\pi} \int_0^{+\infty} e^{(-ik + \frac{1}{2})Y} \frac{\hat{h}(k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}} dk + \frac{1}{2\pi} \int_{-\infty}^0 e^{(-ik + \frac{1}{2})Y} \frac{\hat{h}(k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}} dk.$$

On peut réécrire cette somme à l'aide d'un changement de variable $h = -k$

$$\frac{1}{2\pi} \int_{-\infty}^0 e^{(-ik + \frac{1}{2})Y} \frac{\hat{h}(k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}} dk = \frac{1}{2\pi} \int_0^{\infty} e^{(ik + \frac{1}{2})Y} \frac{\hat{h}(-k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}} dk.$$

Utilisons l'égalité (2.6) on obtient

$$f(x, V, \lambda, \tau) = \frac{1}{2\pi} \int_0^{+\infty} \mathcal{R}[e^{(-ik + \frac{1}{2})Y} \frac{\hat{h}(k + i\frac{1}{2}, V, \lambda, \tau)}{k^2 + \frac{1}{4}}] dk$$

où \mathcal{R} désigne la partie réelle.

Tous les calculs qui suivront pour les *pricing* seront basés sur cette dernière formule. On s'intéresse dans ce qui suit à trouver une formule fermée pour exprimer la fonctionnelle h ou d'une manière équivalente \hat{h} .

Chapitre 3

Le modèle proposé

Dans ce chapitre on va proposer un modèle à saut assez général et établir les formules fermées correspondentes. En première étape, on va proposer une série de sous modèles dont on cherchera les solutions aux EDP (2.9). Ainsi, la construction du modèle général se fera récursivement partant du modèle le plus basique vers un modèle de marché.

3.1 Modèle de Black-Scholes

L'équation différentielle stochastique utilisée dans la théorie avancée par Black-Scholes pour décrire la dynamique du marché est donnée par:

$$dS_t/S_t = (r - d)dt + \sigma dW_t$$

c'est le modèle classique dont on dispose déjà des formules fermées pour le *pricing* des options européennes. Mais en gardant les notations du chapitre précédent on a $V = \sigma^2$ et $\lambda = 0$, l'équation (2.9) devient alors

$$\frac{\partial \hat{h}}{\partial \tau} = (-r - iz(r - d))\hat{h} - \frac{1}{2}(z^2 - iz)V\hat{h} \quad (3.1)$$

Dont la solution est donnée par

$$\hat{h} = C e^{(-r - iz(r - d) - \frac{1}{2}(z^2 - iz)V)\tau} \quad (3.2)$$

En utilisant la condition initiale $\hat{h}(z, V, \tau = 0) = 1$ on conclut que $C = 1$. Injectons ce résultat dans l'égalité (2.11) en introduisant la nouvelle variable

$X = \ln(S/K) + (r - d)\tau$, on obtient alors

$$f(x, V, \tau) = \frac{Ke^{-r\tau}}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} \frac{e^{-izX - \frac{1}{2}(z^2 - iz)V\tau}}{z^2 - iz} dz$$

ou encore

$$f(x, V, \tau) = \frac{Ke^{-r\tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X - \frac{1}{2}(k^2 + \frac{1}{4})V\tau}}{k^2 + \frac{1}{4}} dk$$

On utilisera alors les formules (2.12) et (2.13) pour le *pricing* des options européennes.

3.2 Modèle à Saut

Les Sauts sont très importants car ils sont assez courants sur les marchés et on peut les observer facilement sur les fonds, c'est pour cette raison qu'il devient crucial d'en tenir compte dans nos modèles.

On considère maintenant la dynamique de S_t donnée sous la probabilité \mathbb{P} par

$$dS_t = \mu(S, t)dt + \sigma S_t dW_t^s + J(S) dN_t(\lambda)$$

ou $N_t(\lambda)$ est un processus de poisson d'intensité λ constante, indépendant du mouvement brownien W_t^s . $J(S)$ est une variable aléatoire qui joue le rôle de la taille du Saut de fonction de densité $\phi(J)$.

On suppose que la dynamique de S_t sous la probabilité \mathbb{Q} risque neutre est donnée par

$$dS_t/S_t = (r - d - \lambda m)dt + \sigma dW_t^s + (e^{J_t} - 1)dN_t(\lambda) \quad (3.3)$$

Où $m = \mathbb{E}_t^{\mathbb{Q}}(e^J - 1)$.

L'EDP (2.9) correspondent e à ce modèle s'écrit alors

$$\frac{\partial \hat{h}}{\partial \tau} = (-r - iz(r - d - \lambda m))\hat{h} - \frac{1}{2}(z^2 - iz)V\hat{h} + \lambda \mathbb{E}_t^{\mathbb{Q}}[\hat{h}(e^{-izJ} - 1)] \quad (3.4)$$

On a en particulier

$$\mathbb{E}_t^{\mathbb{Q}}[\hat{h}(e^{-izJ} - 1)] = \int_{-\infty}^{+\infty} [\hat{h}(z, V\lambda, t)(e^{-izJ} - 1)\phi(J)]dJ = \hat{h}(z, V\lambda, t) \int_{-\infty}^{+\infty} (e^{-izJ} - 1)\phi(J)dJ$$

On introduit alors une nouvelle variable très importante dans les calculs qui suivront

$$\Lambda(z) = \int_{-\infty}^{\infty} e^{-iz} \phi(J) dJ - 1 + izm \quad (3.5)$$

$$z = k - \frac{i}{2}$$

$$\Lambda(k) = \int_{-\infty}^{\infty} e^{(-ik + \frac{1}{2})J} \phi(J) dJ - 1 - m(-ik + \frac{1}{2}) \quad (3.6)$$

Finalement en intégrant (3.4) et en introduisons toujours la nouvelle variable $X = \ln(S/K) + (r - d)\tau$ on obtient

$$\hat{h} = e^{-r - iz(r-d) - \frac{1}{2}(z^2 - iz)V + \lambda\tau\Lambda(z)} \quad (3.7)$$

$$f(x, V, \lambda, t) = \int_{-\infty}^{+\infty} \frac{e^{-izX - \frac{1}{2}(z^2 - iz)V + \lambda\tau\Lambda(z)}}{z^2 - iz} dz \quad (3.8)$$

Ou encore en utilisons notre forme préférée

$$f(x, V, \lambda, t) = \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X - \frac{1}{2}(k^2 + \frac{1}{4})V + \lambda\tau\Lambda(k)}}{k^2 + \frac{1}{4}} dk \quad (3.9)$$

Dans la Partie suivante en exprimera explicitement Λ en fonction des paramètres de la loi de la taille des Sauts.

3.2.1 Saut de taille Log-Normale

La solution de l'équation (2.3) est donnée par

$$\Lambda(z) = e^{-\nu zi - \frac{\delta^2}{2}} - 1 + iz(e^{\nu + \frac{\delta^2}{2}} - 1)$$

ou encore en terme de la variable k

$$\Lambda(k) = e^{-ik(\nu + \frac{\delta^2}{2}) - (k^2 - \frac{1}{4})\delta^2 + \frac{\nu}{2}} - 1 - (-ik + \frac{1}{2})(e^{\nu + \frac{\delta^2}{2}} - 1)$$

3.2.2 Saut de taille Double Exponentielle

$$\phi(J) = p \frac{1}{\eta_u} e^{-\frac{J}{\eta_u}} 1_{J \geq 0} + q \frac{1}{\eta_d} e^{-\frac{J}{\eta_d}} 1_{J < 0}.$$

La solution de l'équation (2.3) est donnée par

$$\Lambda(z) = \frac{p}{1 + iz\eta_u} + \frac{q}{1 - iz\eta_d} - 1 + iz \left(\frac{p}{1 - \eta_u} + \frac{q}{1\eta_d} - 1 \right).$$

La transformée de Fourier est définie si $-1/\eta_d < Imz < 1/\eta_u$ et

$$\Lambda(k) = \left(\frac{p}{1 - (-ik + \frac{1}{2})\eta_u} + \frac{q}{1 + (-ik + \frac{1}{2})\eta_d} - 1 - (-ik + \frac{1}{2}) \left(\frac{p}{1 - \eta_u} + \frac{q}{1\eta_d} - 1 \right) \right).$$

3.2.3 Saut de taille Double Log-Normale

On suppose ici que l'amplitude des sauts, notée $J(s)$, est caractérisée par le fait que sa distribution est une mixture de deux distributions gaussiennes

$$\phi(J) = pN(\nu_1, \delta_1^2) + qN(\nu_2, \delta_2^2),$$

où

- p est la probabilité de l'occurrence du premier saut et q est la probabilité de l'occurrence du second saut, $p + q = 1$
- $N(\nu, \delta^2)$ est la densité normale de moyenne ν et de variance δ^2 .

La solution de l'équation (2.3) est donnée par

$$\Lambda(z) = pe^{-\nu_1 z i - \frac{\delta_1^2 z^2}{2}} + qe^{-\nu_2 z i - \frac{\delta_2^2 z^2}{2}} - 1 + iz \left(pe^{\nu_1 + \frac{\delta_1^2}{2}} + qe^{\nu_2 + \frac{\delta_2^2}{2}} - 1 \right)$$

ou encore en terme de la variable k

$$\Lambda(k) = pe^{-(\nu_1 + \frac{\delta_1^2}{2})ik + \frac{\nu_1}{2} - (k^2 - \frac{1}{4})\delta_1^2} + qe^{-(\nu_2 + \frac{\delta_2^2}{2})ik + \frac{\nu_2}{2} - (k^2 - \frac{1}{4})\delta_2^2} - 1 - (-ik + \frac{1}{2}) \left(pe^{\nu_1 + \frac{\delta_1^2}{2}} + qe^{\nu_2 + \frac{\delta_2^2}{2}} \right)$$

3.3 Modèle à volatilité stochastique

Les études empiriques et statistiques ont montré qu'on doit considérer dans certains cas des modèles à volatilité stochastique. C'est pour cette raison qu'on a va considérer en premier temps le fameux modèle de Heston (1993) qui est connu sous le nom de *square root model*.

Modèle de Heston

On considère le modèle suivant:

$$\begin{cases} dS_t &= (r_t - d_t)S_t dt + \sqrt{V_t}S_t dW_t^s \\ dV_t &= \kappa(\theta - V_t)dt + \varepsilon\sqrt{V_t}dW_t^v \end{cases}$$

où κ représente la moyenne de retour, θ est la variance long terme, ε est la vol de vol. De plus, on considère que les deux brownien W_t^s, W_t^v sont supposés corrélés, la constante de corrélation sera notée ρ par la suite.

Remarque 3.3.1 *Dans notre cas, la variance V est non observable, donc il nous faudra calculer ses paramètres à l'aide d'une calibration.*

En accord avec les notations du chapitre précédent, on a

$$a(V_t) = \kappa(\theta - V_t), \quad b(V_t) = \varepsilon\sqrt{V_t}.$$

Par suite, l'EDP satisfaite par les modèles européens $F(S, V, t)$ devient:

$$\frac{\partial F}{\partial t} + (r-d)S \frac{\partial F}{\partial S} + \frac{1}{2}VS^2 \frac{\partial^2 F}{\partial S^2} + \kappa(\theta - V) \frac{\partial F}{\partial V} + \frac{1}{2}\varepsilon^2V \frac{\partial^2 F}{\partial V^2} + \rho\varepsilonVS \frac{\partial^2 F}{\partial S \partial V} - rF = 0$$

avec la condition terminale $F(S, V, \lambda, t = T) = g(S, K)$. En considérant à nouveau la transformée de Fourier comme au chapitre précédent, $\hat{h}(x, V, t)$ satisfait l'EDP suivante:

$$\frac{\partial \hat{h}}{\partial \tau} = (-r - iz(r-d))\hat{h} - \frac{1}{2}(z^2 - iz)V\hat{h} + (\kappa(\theta - V) - iz\rho\varepsilon V) \frac{\partial \hat{h}}{\partial V} + \frac{1}{2}\varepsilon^2V \frac{\partial^2 \hat{h}}{\partial V^2}.$$

Proposition 3.3.1 *La transformée de Fourier $\hat{h}(x, V, t)$ est de la forme:*

$$\begin{cases} \hat{h}(z, V, \tau) = e^{(-r - iz(r-d))\tau + A(z, \tau) + B(z, \tau)V} \\ A(z, \tau) = -\frac{\kappa\theta}{\varepsilon^2} [\psi_+ \tau + 2 \ln(\frac{\psi_- + \psi_+ e^{-\zeta\tau}}{2\zeta})] \\ B(z, \tau) = -(z^2 - iz) \frac{1 - e^{-\zeta\tau}}{\psi_- + \psi_+ e^{-\zeta\tau}} \\ \psi_{\pm} = \mp(\kappa + iz\rho\varepsilon) + \zeta, \zeta = \sqrt{(\kappa + iz\rho\varepsilon)^2 + \varepsilon^2(z^2 - iz)} \end{cases}$$

La preuve de ce résultat est donnée à l'Appendice.

En injectant ce ci dans (3.2), on obtient que l'inverse de cette transformée de Fourier admet la forme suivante:

$$f(x, V, \tau) = \frac{Ke^{-r_t\tau}}{2\pi} \int_{iz-\infty}^{iz+\infty} \frac{e^{-izX + A(z, \tau) + B(z, \tau)V}}{z^2 - iz} dz.$$

En prenant à nouveau $z = k + i\frac{1}{2}$, on obtient la forme intégrale

$$f(x, V, \tau) = \frac{K e^{-r_t \tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X + A(k, \tau) + B(k, \tau)V}}{k^2 + \frac{1}{4}} dk$$

où on note $A(k, \tau) = A(k + i\frac{1}{2}, \tau)$ et $B(k, \tau) = B(k + i\frac{1}{2}, \tau)$ pour simplifier l'écriture. Avec $u = \kappa - \frac{\rho\varepsilon}{2}$ et

$$\psi_{\pm} = \mp(\kappa + ik\rho\varepsilon) + \zeta, \quad \zeta = \sqrt{k^2\varepsilon^2(1 - \rho^2) + 2ik\rho\varepsilon u + u^2 + \frac{\varepsilon^2}{4}}.$$

3.4 Modèle à volatilité stochastique plus Sauts

Il évident que seul le modèle à volatilité stochastique est incapable de répliquer la dynamique des marchés ainsi que certains comportements de plusieurs produits financiers tels que la dynamique des fonds. C'est pour cette raison qu'on est obligé d'ajouter des sauts à notre modèle pour le réécrire sous la forme suivante:

$$\begin{cases} dS_t &= (r_t - d_t)S_t dt + \sqrt{V_t}S_t dW_t^s + (e^{J_t} - 1)dN_t(\lambda) \\ dV_t &= \kappa(\theta - V_t)dt + \varepsilon\sqrt{V_t}dW_t^v \end{cases}$$

où $N_t(\lambda)$ est un processus de Poisson d'intensité constante λ .

En combinant ce problème avec le précédent et en suivant les même méthodes de calcul, on arrive à écrire la fonctionnelle f sous la forme suivante

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{iz-\infty}^{iz+\infty} \frac{e^{-izX + A(z, \tau) + B(z, \tau)V + \lambda\tau\Lambda(z)}}{z^2 - iz} dz,$$

où il faut spécifier A , B et Λ , ou encore

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X + A(k, \tau) + B(k, \tau)V + \lambda\tau\Lambda(k)}}{k^2 + \frac{1}{4}} dz$$

3.5 Modèle à saut avec intensité stochastique et volatilité constante

Square Root Model

Le modèle à étudier est le suivant:

$$\begin{cases} dS_t &= (r_t - d_t - \lambda_t m)S_t dt + \sqrt{V}S_t dW_t^s + (e^{J_t} - 1)dN_t(\lambda_t) \\ d\lambda_t &= \kappa_{\lambda}(\theta_{\lambda} - \lambda_t)dt + \varepsilon_{\lambda}\sqrt{\lambda_t}dW_t^{\lambda} \end{cases}$$

où κ_λ , θ_λ et ε_λ sont respectivement la *mean-reverting*, la *vol* long terme et la *vol de vol* de l'intensité λ du Saut. On suppose que dans ce modèle, les deux browniens W_t^s et W_t^λ sont indépendants.

En accord avec les notation du chapitre précédent, on a pour ce cas particulier

$$a(V) = 0, b(V) = 0, c(\lambda_t) = \kappa_\lambda(\theta_\lambda - \lambda_t), d(\lambda_t) = \varepsilon_\lambda \sqrt{\lambda_t},$$

donc l'EDP () devient:

$$\begin{aligned} & \frac{\partial F}{\partial t} + (r - d - \lambda m)S \frac{\partial F}{\partial S} + \frac{1}{2}V S^2 \frac{\partial^2 F}{\partial S^2} + \kappa_\lambda(\theta_\lambda - \lambda_t) \frac{\partial F}{\partial \lambda} \\ & + \frac{1}{2}\varepsilon_\lambda^2 \lambda_t(\lambda) \frac{\partial^2 F}{\partial \lambda^2} + \lambda \mathbb{E}_t^Q(F(e^J S, V, \lambda, \tau) - F(S, V, \lambda, \tau)) - rF = 0 \end{aligned}$$

avec toujours la condition terminale $F(S, V, \lambda, t = T) = g(S_T, K)$.

Par suite, la transformée fondamentale associe vérifie l'EDP

$$\begin{aligned} & \frac{\partial \hat{h}}{\partial \tau} + (r + iz(r - d - m\lambda))\hat{h} + \frac{1}{2}(z^2 - iz)V\hat{h} \\ & - \kappa_\lambda(\theta_\lambda - \lambda) \frac{\partial \hat{h}}{\partial \lambda} - \frac{1}{2}\varepsilon_\lambda^2 \lambda \frac{\partial^2 \hat{h}}{\partial \lambda^2} - \lambda \hat{h} \Lambda(z) = 0 \end{aligned}$$

dont une solution est donnée par la proposition suivante

Proposition 3.5.1 *En supposant que $\Lambda(z)$ ne dépend pas de τ , la solution de l'EDP () est de la forme suivante*

$$\left\{ \begin{array}{l} \hat{h}(z, V, \tau) = e^{(-r - iz(r - d))\tau - \frac{1}{2}(z^2 - iz)V\tau + C(z, \tau) + D(z, \tau)\lambda} \\ C(z, \tau) = -\frac{\kappa_\lambda \theta_\lambda}{\varepsilon_\lambda^2} [\chi_+ \tau + 2 \ln(\frac{\chi_- + \chi_+ e^{-\xi \tau}}{2\xi})] \\ D(z, \tau) = 2\Lambda(z) \frac{1 - e^{-\xi \tau}}{\chi_- + \chi_+ e^{-\xi \tau}} \\ \chi_\pm = \mp \kappa_\lambda + \xi, \xi = \sqrt{\kappa_\lambda^2 - 2\varepsilon_\lambda^2 \Lambda(z)} \end{array} \right.$$

La preuve est donnée à l'Appendice.

En injectant cette solution dans l'égalité (2.5), on obtient la nouvelle formule spécifique pour ce modèle

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} \frac{e^{-izX - \frac{1}{2}(z^2 - iz)V\tau + C(z, \tau) + D(z, \tau)\lambda}}{z^2 - iz} dz$$

ou d'une manière équivalente

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X - \frac{1}{2}(k^2 + \frac{1}{4})V\tau + C(k, \tau) + D(k, \tau)\lambda}}{k^2 + \frac{1}{4}} dk,$$

où on a noté $C(k, \tau) = C(k + i\frac{1}{2}, \tau)$ et $D(k, \tau) = D(k + i\frac{1}{2}, \tau)$ donné par la proposition si haut.

Dans le cas particulier du modèle précédent $\varepsilon_\lambda = 0$ c-à-d

$$d\lambda_t = \kappa_\lambda(\theta_\lambda - \lambda_t)dt.$$

Etant donnée une condition initiale à l'instant s , la solution de cette EDS simple est donnée par

$$\lambda_t = (\lambda_s - \theta_\lambda)e^{-\kappa_\lambda(t-s)} + \theta_\lambda.$$

La proposition 3.6.1 devient alors

Proposition 3.5.2

$$\left\{ \begin{array}{l} \hat{h}(z, V, \tau) = e^{(-r - iz(r-d))\tau - \frac{1}{2}(z^2 - iz)V\tau + C(z, \tau) + D(z, \tau)\lambda} \\ C(z, \tau) = (\kappa_\lambda\tau - 1 + e^{-\kappa_\lambda\tau}) \frac{\Lambda(z)\theta_\lambda}{\kappa_\lambda} \\ D(z, \tau) = \Lambda(z) \frac{1 - e^{-\kappa_\lambda\tau}}{\kappa_\lambda} \end{array} \right.$$

La preuve est donnée en Appendice.

Comme précédemment on exprime alors f sous la forme

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} \frac{e^{-izX - \frac{1}{2}(z^2 - iz)V\tau + C(z, \tau) + D(z, \tau)\lambda}}{z^2 - iz} dz$$

ou encore

$$f(x, V, \lambda, \tau) = \frac{K e^{-r\tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X - \frac{1}{2}(k^2 + \frac{1}{4})V\tau + C(k, \tau) + D(k, \tau)\lambda}}{k^2 + \frac{1}{4}} dk$$

où C et D sont données par la proposition 3.6.2.

La solution du modèle général formel sera ainsi établie en fonction des formules fermées classiques pour un *Call* ou un *Put* européen.

3.6 Le modèle général

Il s'agit d'un modèle dont la dynamique est caractérisée à la fois par une volatilité et une intensité stochastiques suivant toutes les deux une dynamique *Square Root*.¹

$$\begin{cases} dS_t = (r_t - d_t - \lambda_t m)S_t dt + \sqrt{V}S_t dW_t^s + (e^{J_t} - 1)dN_t(\lambda_t) \\ dV_t = \kappa(\theta - V_t)dt + \varepsilon\sqrt{V_t}dW_t^v \\ d\lambda_t = \kappa_\lambda(\theta_\lambda - \lambda_t)dt + \varepsilon_\lambda\sqrt{\lambda_t}dW_t^\lambda \end{cases}$$

Dans le même esprit que précédemment, on écrit l'EDP vérifiée par un prix d'une option du style européenne puis on établit la formule fermée correspondante. On vérifie alors aisément que dans le cas où

$$a(V) = \kappa(\theta - V), \quad b(V) = \varepsilon\sqrt{V}, \quad c(\lambda) = \kappa_\lambda(\theta_\lambda - \lambda), \quad d(\lambda) = \varepsilon_\lambda\sqrt{\lambda},$$

F est solution de

$$\frac{\partial F}{\partial t} + (r - d - \lambda m)S \frac{\partial F}{\partial S} + \frac{1}{2}VS^2 \frac{\partial^2 F}{\partial S^2} + \kappa(\theta - V) \frac{\partial F}{\partial V} + \frac{1}{2}\varepsilon\sqrt{V} \frac{\partial^2 F}{\partial V^2} \quad (3.10)$$

$$+ \rho\varepsilon\sqrt{V}\sqrt{V}S \frac{\partial^2 F}{\partial S \partial V} + \kappa_\lambda(\theta_\lambda - \lambda) \frac{\partial F}{\partial \lambda} + \frac{1}{2}\varepsilon_\lambda^2 \lambda \frac{\partial^2 F}{\partial \lambda^2} \quad (3.11)$$

$$+ \lambda \mathbb{E}_t^Q(F(e^J S) - F(S)) - rF = 0 \quad (3.12)$$

avec la condition finale $F(S, V, \lambda, t = T) = g(S_T, K)$ où g désigne le *payoff*. La transformée de Fourier inverse est alors obtenue en se basant sur les résultats précédents. D'où le théorème suivant

Théorème 3.6.1 *La transformée de Fourier inverse correspondant au modèle () pour un payoff $f = \min(S, K)$ est définie pour $z \in \mathbb{C}$, $0 < \text{Im}z < 1$ et est donnée par la formule suivante*

$$f(x, V, \lambda, \tau) = \frac{Ke^{-r\tau}}{2\pi} \int_{iz_i - \infty}^{iz_i + \infty} \frac{e^{-izX + A(z, \tau) + B(z, \tau)V + C(z, \tau) + D(z, \tau)\lambda}}{z^2 - iz} dz$$

avec $\tau = T - t$ et $X = \ln(S/K) + (r - d)\tau$. Si on choisit $z = k + i\frac{1}{2}$, $k \in \mathbb{R}$, alors on peut récrire la formule dernière sous une autre forme plus simplifiée et surtout beaucoup plus pratique à implémenter

$$f(x, V, \lambda, \tau) = \frac{Ke^{-r\tau}}{2\pi} \int_{-\infty}^{+\infty} \frac{e^{(-ik + \frac{1}{2})X + A(k, \tau) + B(k, \tau)V + C(k, \tau) + D(k, \tau)\lambda}}{k^2 + \frac{1}{4}} dk$$

1. Pour plus de détail, voir [1]

avec la même formule pour un Call européen

$$C(S,V,\lambda,\tau) = Se^{-d_t\tau} - f(x,V,\lambda,\tau).$$

Et pour le Put européen, on a la formule suivante

$$P(S,V,\lambda,\tau) = K e^{-r_t\tau} - f(x,V,\lambda,\tau).$$

On peut à partir du modèle général et le théorème précédent déduire des résultats similaires pour plusieurs sous-modèles, qu'on peut classer suivant trois caractéristiques : la volatilité, l'intensité du Poisson et la taille du Saut

Volatilité

— Constante

$$A(z,\tau) = A(k,\tau) \equiv 0$$

$$B(z,\tau) = -\frac{1}{2}(z^2 - iz)V\tau; B(k,\tau) = -\frac{1}{2}(k^2 + \frac{1}{4})V\tau$$

— Stochastique

$$A(k,\tau) = -\frac{\kappa\theta}{\varepsilon^2}[\psi_+\tau + 2\ln(\frac{\psi_- + \psi_+e^{-\zeta\tau}}{2\zeta})]$$

$$B(k,\tau) = -(z^2 - iz)\frac{1 - e^{-\zeta\tau}}{\psi_- + \psi_+e^{-\zeta\tau}}$$

$$\psi_{\pm} = \mp(\kappa + ik\rho\varepsilon) + \zeta, \zeta = \sqrt{k^2\varepsilon^2(1 - \rho^2) + 2ik\rho\varepsilon u + u^2 + \frac{\varepsilon^2}{4}}, u = \kappa - \frac{\rho\varepsilon}{2}$$

Intensité

— Abscente

$$\lambda = 0 \Rightarrow C(z,\tau) = C(k,\tau) \equiv 0 \text{ et } D(z,\tau) = D(k,\tau) \equiv 0$$

— Constante

$$C(z,\tau) = C(k,\tau) \equiv 0, \text{ et } D(z,\tau) = \lambda\Lambda(z); D(k,\tau) = \lambda\Lambda(k)$$

— Déterministe

$$C(k,\tau) = (\kappa_\lambda\tau - 1 + e^{-\kappa_\lambda\tau})\frac{\Lambda(k)\theta_\lambda}{\kappa_\lambda}$$

$$D(k,\tau) = \Lambda(k)\frac{1 - e^{-\kappa_\lambda\tau}}{\kappa_\lambda}$$

— Stochastique

$$C(k,\tau) = -\frac{\kappa_\lambda\theta_\lambda}{\varepsilon_\lambda^2}[\chi_+\tau + 2\ln(\frac{\chi_- + \chi_+e^{-\xi\tau}}{2\xi})]$$

$$D(k,\tau) = 2\Lambda(k)\frac{1 - e^{-\xi\tau}}{\chi_- + \chi_+e^{-\xi\tau}}$$

$$\chi_{\pm} = \mp \kappa_{\lambda} + \xi, \xi = \sqrt{\kappa_{\lambda}^2 - 2\varepsilon_{\lambda}^2 \Lambda(k)}$$

Taille du Saut— *Log Normal*

$$\Lambda(k) = e^{-ik(\nu + \frac{\delta^2}{2}) - (k^2 - \frac{1}{4})\delta^2 + \frac{\nu}{2}} - 1 - (-ik + \frac{1}{2})(e^{\nu + \frac{\delta^2}{2}} - 1)$$

— *Double Log Normal*

$$\begin{aligned} \Lambda(k) = & p e^{-(\nu_1 + \frac{\delta_1^2}{2})ik + \frac{\nu_1}{2} - (k^2 - \frac{1}{4})\delta_1^2} + q e^{-(\nu_2 + \frac{\delta_2^2}{2})ik + \frac{\nu_2}{2} - (k^2 - \frac{1}{4})\delta_2^2} \\ & - 1 - (-ik + \frac{1}{2})(p e^{\nu_1 + \frac{\delta_1^2}{2}} + q e^{\nu_2 + \frac{\delta_2^2}{2}}) \end{aligned}$$

— *Double Exponentielle*

$$\Lambda(k) = \frac{p}{1 - (-ik + \frac{1}{2})\eta_u} + \frac{q}{1 + (-ik + \frac{1}{2})\eta_d} - 1 - (-ik + \frac{1}{2})\left(\frac{p}{1 - \eta_u} + \frac{q}{1 - \eta_d} - 1\right)$$

Chapitre 4

Etude Pratique

La première étape nous a permis d'établir toutes les formules fermées nécessaires pour le modèle proposé, on se pose le problème de l'implémentation et de la calibration du modèle au marché.

Dans la base de données **Pertrac**, on peut avoir accès aux rendements historiques de chaque fond sous forme d'une valeur $\Delta S/S$ par mois. Mais on n'a aucune autre information; la volatilité est inobservable ce qui complique la calibration du modèle au marché. L'idée proposée est la suivante:

- Puisqu'on a les formules fermées, on peut donc dériver le prix du *Call* par rapport au *Strike* ce qui nous permettra de définir une courbe de densité.
- L'étape suivante consiste à définir une densité empirique c-à-d à partir des données du marché.

En pratique, on prend l'historique d'un *fund* particulier et on suppose que le processus en question est un processus de Markov homogène¹, donc on peut dire que les deux processus $\Delta S_t = S_{t+\Delta t} - S_t$ et S_t sont indépendants et que les deux processus $\Delta S_t = S_{t+\Delta t} - S_t$ et $\Delta S_{\Delta t} = S_{\Delta t} - S_0$ ont même loi, donc on pose $S_0 = 1(100)$.

Notons $\Delta S_i = \Delta S_{i\Delta t}$, et posons

$$P_i = P(\Delta S \leq \Delta S_i) \equiv \frac{\#\{j, \Delta S_j \leq \Delta S_i\}}{N}$$

où N est le nombre total des données. Notons par suite $x_i = 1 + \Delta S_i$, la famille des couples (x_i, P_i) représente alors notre densité empirique (les x_i sont les *Strikes*).

1. Voir la définition au Chapitre 2

Ainsi la stratégie de calibration devient un peu plus claire et consistera à minimiser la quantité suivante:

$$\min_{X \in \mathbb{S}} \sum_{i=1}^{i=N} \left(1 - \frac{\partial Call}{\partial K}(K_i = x_i) - P_i\right)^2,$$

où X est le vecteur de paramètres et \mathbb{S} c'est l'espace de tous les vecteurs paramètres. Des conditions et des contraintes seront dans la suite établies sur l'espace \mathbb{S} .

4.1 Intégration numérique

Avant d'attaquer la calibration, il faut d'abord savoir bien calculer numériquement

$$\frac{\partial Call}{\partial K}$$

au sens d'une erreur optimale, ce terme qui se présente dans les formules fermées établies sous forme d'une intégrale. En pratique, il y'a beaucoup de méthodes d'intégration numériques; la méthode des trapèzes, de Newton-cotes, de Gauss...

Dans ce travail, on va plutôt se pencher sur l'une des méthodes de Gauss (Gauss Hermite, Gauss Legendre, Gauss Laguerre) suivant la forme de l'intégrand; par exemple si la fonction à intégrer est de la forme $e^{-x^2}f(x)$, on utilise la méthode de Gauss Hermite qui est la plus adaptée pour ce cas.

Exemple 4.1.1 *On se propose d'approcher la somme suivante numériquement, où $f \in \mathbb{L}^1$*

$$I = \int_a^b f(x)dx$$

par la méthode de Gauss. On a l'équivalence suivante

$$I \simeq \sum_{i=1}^{i=N} W_i f(x_i),$$

où N représente le nombre de points sur lesquels on veut approcher l'intégrale. W_i et x_i sont respectivement les poids et les points de Gauss qui dépendent dans la construction de l'algorithme des paramètres N , a et b .

Remarque 4.1.1 *Pour des sommes de la forme*

$$I = \int_a^b f(x)dx, \quad -\infty < a < b < +\infty,$$

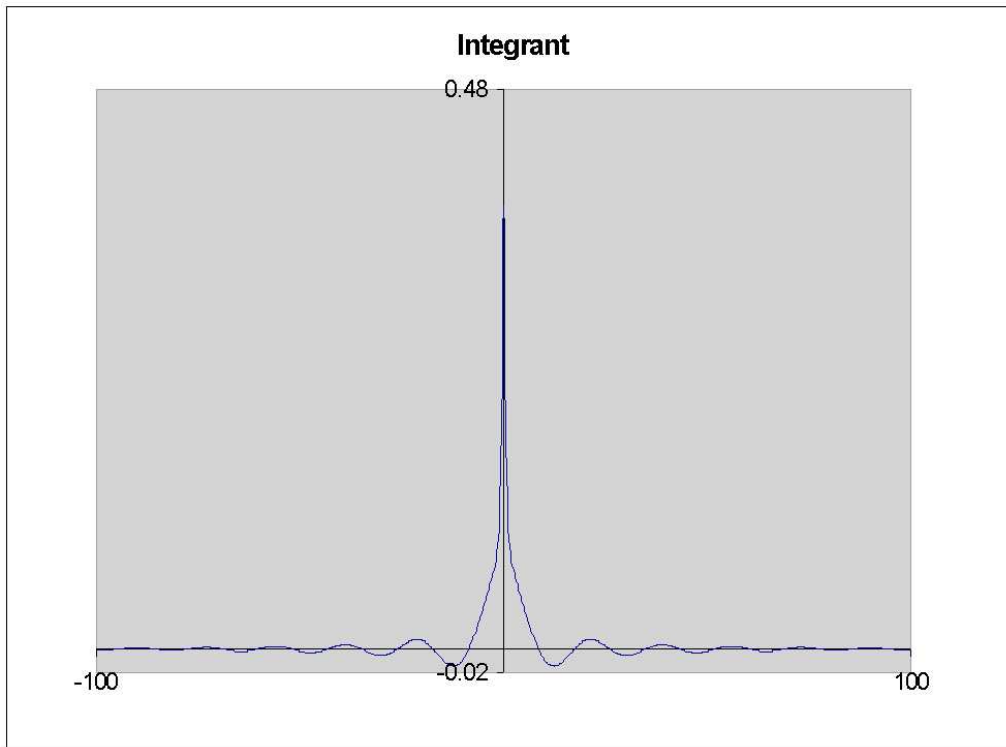


FIG. 4.1 – Oscillations observées sur des intégrales numériques

on arrive toujours assez facilement à les approcher numériquement pour $|a|$, et $|b|$ pas très grands. Mais quand l'intervalle d'intégration est \mathbb{R} , l'intégration numérique pose quelques problèmes.

- Pour avoir une bonne approche, il faut toujours assez de points donc N est infini pour le cas où $a = -\infty$, $b = +\infty$.
- Les méthodes de Gauss ne marchent plus pour $N > 170$.
- L'intégrant ne s'annule pas rapidement quand x augmente en valeur absolue, donc on se rend compte qu'on est obligé de choisir des grandes valeurs pour N .

Une méthode pour surmonter cette étape consiste à se fixer deux réels A et B assez grands de telle façon à ce que $\forall x \in [A, B]$, $f(x) \simeq 0$ et même s'ils sont très grands, on peut faire une subdivision de

$$[A, B] = \bigcup_{i=0}^{i=M-1} [A_i, A_{i+1}]$$

avec $A_0 = A$, $A_M = B$, et on obtient que

$$I = \int_{-\infty}^{+\infty} f(x)dx \simeq \int_A^B f(x)dx = \sum_{i=0}^{i=M-1} \int_{A_i}^{A_{i+1}} f(x)dx \simeq \sum_{i=0}^{i=M-1} \sum_{j=1}^{j=N} W_j^i f(x_j^i)$$

Remarque 4.1.2 Dans notre cas, les réels A et B dépendent fortement des paramètres du modèle mais ceci n'est pas un problème vu qu'on peut observer le comportement de l'intégrand en faisant varier les paramètres et en observant un minimum pour les valeurs prises par A et un maximum pour les valeurs prises par B .

4.2 Calibration du modèle

4.2.1 nag-opt-nlin-lsq

Pour calibrer le modèle proposé, on va utiliser la librairie **NAG** et plus précisément la fonction **nag – opt – nlin – lsq(e04unc)** qui est destinée à minimiser n'importe quelle somme de carrés de fonctions sous certaines contraintes (qui peuvent se présenter comme simples bandes ou encore des contraintes soit linéaires ou non linéaires sur les variables voire même une combinaison). Bien entendu, la fonction **nag – opt – nlin – lsq(e04unc)** peut être utilisée pour un problème d'optimisation avec une combinaison de contraintes linéaires et non linéaires ou encore sans contraintes.

4.2.2 specification

```
#include <nag.h>
#include <nage04.h>
```

```
void nag_opt_lin_lsq(Integer m, Integer n, Integer nclin, double a[],
    Integer tda, double bl[], double bu[], double cvec[],
    double b[], double h[], Integer tdh, Integer kx[],
    double x[], double *objf, Nag_E04_Opt *options,
    Nag_Comm *comm, NagError *fail)
```

4.2.3 Description

Le problème que la fonction **nag – opt – nlin – lsq(e04unc)** peut résoudre admet la forme suivante

$$\min_{X \in \mathbb{S}} F(x) = \frac{1}{2} \sum_{i=1}^{i=m} [f_i(x) - y_i]^2$$

où

$$\mathbb{S} = \left\{ x, l \leq \begin{pmatrix} x \\ A_L x \\ c(x) \end{pmatrix} \leq u \right\} \cap \mathbb{R}^n$$

et F est la fonction à minimiser qu'on appellera la fonction objective et qui est une fonction non linéaire qui se présente sous la forme de somme de carrés de N fonctions

$$(f_1(x) - y_1, f_2(x) - y_2, \dots, f_m(x) - y_m),$$

où les $(y_i)_{1 \leq i \leq m}$ sont des constantes, A_L est une matrice $n_L \times n$ c'est la matrice des contraintes linéaires, et $c(x)$ est le vecteur des contraintes non linéaire dans \mathbb{R}^{n_N} . La fonction objective et la fonction des contraintes non linéaires $c(x)$ doivent être continûment différentiables (la méthode de **nag – opt – nlin – lsq(e04unc)** peut résoudre le problème (4.1) si et seulement si il n'y a que quelques points de singularités isolés de la solution optimale.)

Remarque 4.2.1 — *Dans le cas où on a pas de contraintes linéaires, il suffit de prendre la matrice A_L comme la matrice nulle, $c(x) = 0$ dans le cas d'absence de contraintes non linéaires, et d'imposer aux coefficients du vecteur l (lower boundary) de prendre la valeur $-\infty$ et $+\infty$ pour celles de u (upper boundary) si on cherche une solution sans conditions sur les coefficients de x .*

— *Si on a pas de contraintes non linéaires dans (4.1) et si la fonction objective F est linéaire ou quadratique, alors il vaudra mieux utiliser l'une des fonctions*

nag – opt – lp(e04mfc), nag – opt – nlin – lsq(e04ncc), nag – opt – qp(e04nfc)

car elles seront les mieux adaptées et beaucoup plus efficaces.

— *En utilisant **NAG**, il est très utile de lui fournir un vecteur initial x_0 , aussi proche de la solution que possible.*

Exemple 4.2.1 Dans le cas du modèle général où la taille du saut suit une loi double exponentielle², on a

$$X = \begin{pmatrix} V_0 \\ \kappa \\ \varepsilon \\ \theta \\ \lambda \\ \kappa_\lambda \\ \varepsilon_\lambda \\ \theta_\lambda \\ \rho \\ \nu_1 \\ \nu_2 \\ p \end{pmatrix}$$

avec $(\lambda, \kappa, \kappa_\lambda, \varepsilon, \varepsilon_\lambda, \theta, \theta_\lambda, \nu_1, \nu_2, \rho, p, V_0) \in \mathbb{R}_+^9 \times [-1, 1] \times [0, 1]^2$

\Updownarrow

$$l_1 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ -1 \\ 0 \\ 0 \\ 1 \end{pmatrix} \leq X = \begin{pmatrix} V_0 \\ \kappa \\ \varepsilon \\ \theta \\ \lambda \\ \kappa_\lambda \\ \varepsilon_\lambda \\ \theta_\lambda \\ \rho \\ \nu_1 \\ \nu_2 \\ p \end{pmatrix} \leq \begin{pmatrix} 1 \\ +\infty \\ +\infty \\ +\infty \\ +\infty \\ +\infty \\ +\infty \\ +\infty \\ 1 \\ +\infty \\ +\infty \\ 1 \end{pmatrix} = u_1$$

D'autre part pour assurer la positivité du processus de la volatilité et celui de l'intensité du saut, on impose les conditions $2\kappa\theta - \varepsilon^2 < 0$ et $2\kappa_\lambda\theta_\lambda - \varepsilon_\lambda^2 < 0$

2. Voir la définition du modèle au Chapitre 3

et donc $c(x) = \begin{pmatrix} 2\kappa\theta - \varepsilon^2 \\ 2\kappa_\lambda\theta_\lambda - \varepsilon_\lambda^2 \end{pmatrix}$, et en posant $l_2 = \begin{pmatrix} -\infty \\ -\infty \end{pmatrix}$, $u_2 = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$

$$\mathbb{S} = \left\{ x, l \leq \begin{pmatrix} x \\ A_L x \\ c(x) \end{pmatrix} \leq u \right\} \cap \mathbb{R}^n$$

avec $l = \begin{pmatrix} l_1 \\ l_2 \end{pmatrix}$, $u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix}$.

Il est clair qu'il suffit de prendre

$$f_i(X) = 1 - \frac{\partial \text{Call}}{\partial K}(K_i = x_i)(X)$$

et $y_i = P_i$ pour pouvoir utiliser la fonction **NAG** qu'on vient de décrire si haut.

Cette fonction va être intégrée dans un code *C++* qui prendra comme *Inputs* la courbe de densité empirique $(K_i, P_i)_i$, la maturité, le taux court, les dividendes (d dans le modèle), un entier n pour calculer les poids et les points de Gauss Legendre nécessaires pour calculer les intégrales et finalement un vecteur initial de paramètres X_0 .

Remarque 4.2.2 Avec la librairie **NAG**, on a l'avantage de suivre les calculs que **NAG** fait pas par pas dans un fichier *.txt*, et on a ainsi tous les renseignements nécessaires sur la convergence, la rapidité et surtout l'erreur ou encore la distance entre densité empirique et celle théorique à chaque majoration et minoration comme le montre l'exemple suivant de fichier où l'on a lancé une calibration sur 5 paramètres pour 30 itérations maximum.

Parameters to e04unc

Number of variables.....	5		
Linear constraints.....	0	Nonlinear constraints.....	2
start.....	Nag_Cold		
step_limit.....	2.00e+000	machine precision.....	1.11e-016
lin_feas_tol.....	1.05e-008	nonlin_feas_tol.....	1.05e-008
optim_tol.....	1.00e-009	linesearch_tol.....	9.00e-001
crash_tol.....	1.00e-002	f_prec.....	4.37e-015
inf_bound.....	1.00e+020	inf_step.....	1.00e+020

```

max_iter.....          30      minor_max_iter.....          50
hessian.....          FALSE    h_reset_freq.....            2
h_unit_init.....      FALSE
f_diff_int.....Automatic    c_diff_int.....Automatic
obj_deriv.....        FALSE    con_deriv.....              TRUE
verify_grad.....Nag_CheckObjCon  print_deriv.....Nag_D_Full
print_level.....Nag_Soln_Iter    minor_print_level.....Nag_NoPrint
outfile.....C:\Documents and Settings\mmrad\My Documents\VolSto.txt

```

Verification of the objective gradients.

All objective gradient elements have been set.
Derivative level changed to Nag_AllG_AllJ.

The Jacobian seems to be ok.

The largest relative error was 2.12e-006 in subfunction 1

Component-wise check:

Subfunction 1

Elem.	x[j]	dx[j]	Jacobian value	Difference approxn.	Itns.
1	8.00e-004	9.55e-009	7.32755150e+000	7.32758815e+000 OK	2
2	8.28e-001	4.33e-007	1.21001735e-001	1.21002419e-001 OK	1
3	2.00e-002	7.45e-007	-2.95798840e-001	-2.95799523e-001 OK	2
4	3.91e-001	2.57e-007	2.62525234e-001	2.62526546e-001 OK	1
5	-7.64e-001	1.84e-005	7.66657000e-003	7.66659896e-003 OK	2

5 Jacobian elements out of the 5
set in cols 1 through 5 seem to be ok.

The largest relative error was 4.40e-006 in row 1 column 1
Minor itn 4 -- Re-solve QP subproblem.

Maj	Mnr	Step	Objective	Norm Gz	Cond Hz	C
0	5	0.0e+000	1.450655e-001	1.6e-001	1.0e+000	C
1	3	1.0e+000	9.847597e-002	2.8e-001	1.0e+000	
2	3	1.0e+000	7.485691e-002	0.0e+000	1.0e+000	
3	3	3.9e-001	4.235816e-002	6.3e+000	1.2e+007	

4	4	3.1e-001	9.776397e-003	6.8e-002	1.0e+000
5	2	1.0e-001	8.553772e-003	1.0e+000	1.9e+001
6	3	1.0e+000	7.887440e-003	0.0e+000	1.0e+000
7	3	8.6e-002	5.717580e-003	4.3e-002	1.0e+000
8	3	4.2e-002	5.532398e-003	3.8e-002	4.0e+000
9	5	8.1e-002	5.126903e-003	2.1e-002	4.0e+000
10	4	4.1e-001	1.629430e-003	3.2e-002	1.0e+000
11	5	1.2e-001	3.152801e-004	3.2e-003	4.0e+000
12	4	3.1e-002	2.161412e-004	5.4e-003	1.0e+000
13	3	3.7e-003	2.128130e-004	6.2e-003	4.0e+000
14	4	5.2e-002	2.084082e-004	9.4e-003	1.0e+000
15	4	2.0e-002	1.793554e-004	5.1e-003	4.0e+000
16	4	6.0e-002	1.720905e-004	8.6e-003	1.0e+000
17	4	1.9e-002	1.451589e-004	4.5e-003	4.0e+000
18	4	5.0e-002	1.401739e-004	7.3e-003	1.0e+000
19	4	1.5e-002	1.228853e-004	4.2e-003	4.0e+000
20	4	4.1e-002	1.194376e-004	6.4e-003	1.0e+000
21	4	1.2e-002	1.076609e-004	3.8e-003	4.0e+000
22	4	3.6e-002	1.050863e-004	5.7e-003	1.0e+000
23	4	1.0e-002	9.644457e-005	3.5e-003	4.0e+000
24	4	3.2e-002	9.442260e-005	5.2e-003	1.0e+000
25	4	9.2e-003	8.776333e-005	3.2e-003	4.0e+000
26	4	2.9e-002	8.612161e-005	4.7e-003	1.0e+000
27	4	8.2e-003	8.081588e-005	3.0e-003	4.0e+000
28	4	2.7e-002	7.945015e-005	4.3e-003	1.0e+000
29	4	7.4e-003	7.511797e-005	2.8e-003	4.0e+000
30	4	2.5e-002	7.395997e-005	3.9e-003	1.0e+000

Exit from NP problem after 30 major iterations, 118 minor iterations.

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Residual
V 1	FR	1.31205e-002	8.00000e-004	2.00000e-002	4.6138e-002	6.8795e-003
V 2	FR	5.00600e-004	1.00000e-009	1.00000e+002	3.3854e-005	5.0060e-004
V 3	FR	1.34305e+000	1.00000e-009	2.00000e+000	-5.7326e-004	6.5695e-001
V 4	FR	6.50713e-003	1.00000e-009	1.00000e+000	2.8238e-006	6.5071e-003
V 5	FR	-6.28408e-001	-1.00000e+000	1.00000e+000	0.0000e+000	3.7159e-001

4.3 Résultats numériques et commentaires

Les premiers résultats de calibration qu'on a eu pour un modèle à volatilité stochastique et une taille de Saut qui suit une loi double exponentielle

$$\phi(J) = p \frac{1}{\eta_u} e^{-\frac{J}{\eta_u}} 1_{\{J \geq 0\}} + q \frac{1}{\eta_d} e^{-\frac{J}{\eta_d}} 1_{\{J < 0\}}$$

sont les suivantes

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Residual
V	1 FR	9.18596e-002	0.00000e+000	1.20000e+001	1.1496e-004	9.1860e-002
V	2 FR	2.88541e-001	0.00000e+000	1.00000e+002	9.7138e-004	2.8854e-001
V	3 FR	2.77167e+001	0.00000e+000	1.00000e+002	-1.9736e-005	2.7717e+001
V	4 FR	8.41869e+000	0.00000e+000	1.00000e+002	3.1951e-005	8.4187e+000
V	5 FR	-4.51879e-001	-9.99000e-001	9.99000e-001	1.0738e-004	5.4712e-001
V	6 FR	1.81088e-001	0.00000e+000	1.00000e+002	5.7148e-004	1.8109e-001
V	7 FR	1.19722e-008	1.00000e-014	1.00000e+000	5.9312e-010	1.1972e-008
V	8 A LL	1.62989e-010	1.00000e-014	1.00000e+002	6.2476e-010	1.6298e-010
V	9 FR	1.72699e-002	0.00000e+000	1.00000e+000	-2.3725e-011	1.7270e-002

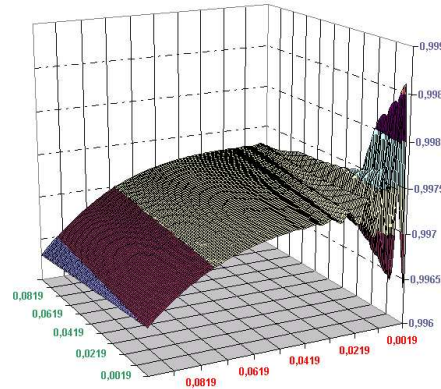
Erreur 3.495655e-003

On remarque ici un fait inadmissible; c'est le fait que la deuxième et la troisième valeurs des résultats si haut correspondant à $V_0 = 2.88541e - 001$ et $\varepsilon = 2.77167e + 001$ sont très grandes par rapport à ce qu'on observe sur le marché.

On peut penser que ceci est dû au modèle et qu'il faut peut-être essayer de voir s'il n'y a pas de corrélation entre certains paramètres ou changer la densité de la taille du saut en une densité Log Normale ou encore Double Log Normale, mais on a des comportements similaires si on modifie un peu les valeurs initiales des paramètres X_0 . Les explications avancées ne résoud pas le problème, donc il faut d'avantage imposer des limites sur ces paramètres pour ne pas avoir à aller chercher loin le minimum cherché pour la calibration.

D'autre part, le fait de changer un peu les valeurs initiales change totalement les résultats et ceci n'est pas dû qu'au seul fait de l'existence de plusieurs minimums locaux, c'est cela qui va rendre le problème beaucoup plus difficile et demandera d'étudier de plus prêt le comportement de la fonction objective F en fonction des paramètres du modèle.

theta versus V (kappa=1.863, epsilon= 0.411459, rho=-1.0)

FIG. 4.2 – *theta Versus V*

4.3.1 Existence de plusieurs minimums locaux

Dans le paragraphe précédent, on a évoqué la possibilité de l'existence de plusieurs minimums locaux. Ici, on va étudier de plus près l'évolution de la fonction objective en fonction des paramètres et on va essayer de trouver si c'est possible de construire une stratégie fiable qui va permettre de faire converger l'algorithme vers les bonnes valeurs.

Sur les figures 4.2, 4.3 et 4.4, on a représenté la surface de F en fonction de certains couples de paramètres significatifs. On sait que pour avoir de meilleurs résultats de calibration avec la fonction `nag – opt – nlin – lsq(e04unc)`, il est très important de lui fournir déjà une bonne approximation pour deux raisons majeure:

On observe facilement sur ces graphes, et les nombreuses tentatives de calibration qu'on a lancées, que le problème est beaucoup plus complexe que ce qu'on a pensé, car il ne s'agit pas de 2 ou 3 minimums possibles dont il faut en choisir un, mais au contraire il est question d'un nombre assez important dans un petit domaine. Alors, quels seront les critères pour le choix de ce minimum? Ce problème est fondamental vu que le choix d'un minimum est équivalent au choix des paramètres du modèle calibré.

Partant des observations sur le marché et des données historiques disponibles, on peut limiter les intervalles de définition de V_0 et ε car on sait qu'on a généralement

$$V_0 \in [0.03, 0.2], \varepsilon \in [0.1, 1.4],$$

et ce sont les deux paramètres les plus significatifs dans le modèle vu que les prix sont très sensibles aux moindres perturbations sur ces deux derniers.

V versus Epsilon ($\kappa=0.8$, $\theta=0.0025$, $\rho=-0.2$)

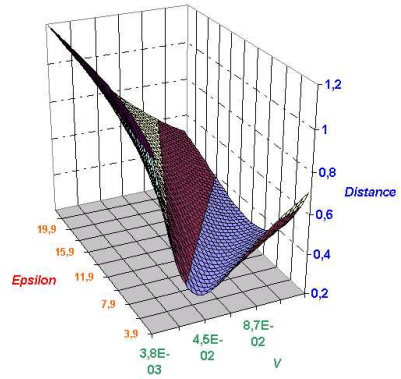


FIG. 4.3 – *V versus epsilon*

V versus theta ($\epsilon = 1.1$, $\kappa = 1.863$, $\rho = -0.5$)

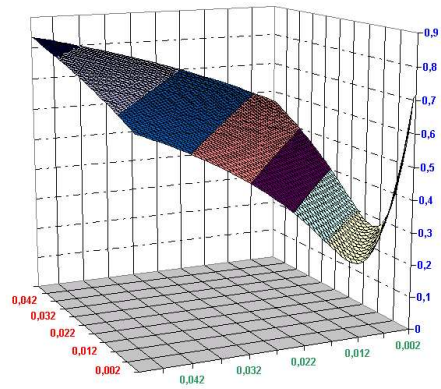


FIG. 4.4 – *V versus theta*

Mais ceci n'est pas vraiment concluant car on n'a aucune idée sur presque la totalité du reste des paramètres où même d'une éventuelle corrélation. D'autre part, il est encore question de non unicité de la solution.

En terme de code $C++$, il n'y a pas de vraies difficultés et on arrive à bien faire marcher l'algorithme. Par contre, les résultats de la calibration restent très dépendants des paramètres initialement fixés. Le problème qui se pose à ce niveau est le suivant:

- Comment peut-on trouver une bonne approximation des donnée initiales?
- Y'a t'il une approche différente qui peut résoudre se problème?

L'idée la plus naturelle est de chercher une nouvelle approche qui ne soit pas basée sur les formules fermées, car ce sont des fonctions qui oscillent beaucoup et sont à l'origine de la non unicité du minimum. C'est pour cette raison et vu les informations dont on dispose, qu'on était amené à essayer de calculer théoriquement en fonction des paramètres du modèle les quantités suivantes

$$[Cov_t(\Delta S_{t+i\Delta}^2 \Delta S_{t+j\Delta}^2)]_{i,j}.$$

Toute fois, en simplifiant un peu la dynamique du modèle pour l'écrire sous la forme

$$\left\{ \begin{array}{l} dS_t = (r_t - d_t - \lambda_t m) S_t dt + \sqrt{V} S_t dW_t^s + \ln(J_t) dN_t(\lambda_t), J_t \sim \mathcal{N}(\nu, \delta) \\ dV_t = \kappa(\theta - V_t) dt + \varepsilon \sqrt{V_t} dW_t^v \\ \lambda_t = \lambda_2 V_t + \lambda_1, \lambda_1, \lambda_2 \geq 0 \\ dW_t^s dW_t^v = \rho dt, t \in [0, \Delta] \end{array} \right.$$

on obtient les égalités suivantes

$$Cov_t(\Delta S_{t+i\Delta}^2 \Delta S_{t+j\Delta}^2)_{i,j} = d_1 e^{-(j+i)\kappa\Delta} + d_2 e^{-j\kappa\Delta} + d_3 e^{-(j-i)\kappa\Delta},$$

où $j > i$ et donc

$$\left\{ \begin{array}{l} d_1 = (1 + \lambda_1(\nu^2 + \delta^2))^2 (e^{\kappa\Delta} - 1)^2 \frac{\theta - 2V_t}{2\kappa^3} \\ d_2 = (1 + \lambda_1(\nu^2 + \delta^2))(1 - e^{\kappa\Delta})(1 + \lambda_1(\nu^2 + \delta^2) + \kappa\Delta\rho^2) \frac{(\theta - V_t)\varepsilon^2\Delta}{\kappa^2} \\ d_3 = (1 + \lambda_1(\nu^2 + \delta^2))\theta\varepsilon^2 \frac{(1 + \lambda_1(\nu^2 + \delta^2) + 4\rho^2) + 4\kappa\Delta\rho^2(e^{-\kappa\Delta} - 1)}{2\kappa^3} \end{array} \right.$$

Les résultats numériques obtenus avec cette nouvelle méthode sont les suivants

Parameters to e04unc

Number of variables.....	8	
Linear constraints.....	0	Nonlinear constraints..... 1
start.....	Nag_Cold	
step_limit.....	2.00e+000	machine precision.....1.11e-016
lin_feas_tol.....	1.05e-008	nonlin_feas_tol.....1.05e-008
optim_tol.....	1.00e-009	linesearch_tol.....9.00e-001
crash_tol.....	1.00e-002	f_prec.....4.37e-015
inf_bound.....	1.00e+020	inf_step.....1.00e+020
max_iter.....	30	minor_max_iter..... 50
hessian.....	FALSE	h_reset_freq..... 2
h_unit_init.....	FALSE	
f_diff_int.....	Automatic	c_diff_int.....Automatic
obj_deriv.....	FALSE	con_deriv..... TRUE
verify_grad.....	Nag_CheckObjCon	print_deriv.....Nag_D_Full
print_level.....	Nag_Soln_Iter	minor_print_level.....Nag_NoPrint
outfile.....	C:\Documents and Settings\mmrad\My Documents\M2.txt	

Verification of the objective gradients.

All objective gradient elements have been set.
Derivative level changed to Nag_AllG_AllJ.

The Jacobian seems to be ok.

The largest relative error was 3.59e-010 in sub function 1

Component-wise check:

Subfunction 1

Elem.	x[j]	dx[j]	Jacobian value	Difference approxn.	Itns.
2	5.00e-001	7.43e-004	-5.23630167e-006	-5.23632925e-006	OK 3
4	4.00e-001	5.26e-004	-3.55625078e-006	-3.55626733e-006	OK 3
5	2.00e-001	2.06e-006	-9.09225355e-004	-9.09229905e-004	OK 2
6	-6.00e-001	3.64e-005	8.70031477e-006	8.70035881e-006	OK 3
7	1.00e-001	1.16e-005	-1.42250031e-005	-1.42250694e-005	OK 2
8	1.00e-001	1.16e-005	-1.42250031e-005	-1.42250694e-005	OK 2

The 2 zero assigned gradient elements have been checked and confirmed as constants. Elements are:

The zero elements are:
(1,1), (1,3).

8 Jacobian elements out of the 8
set in cols 1 through 8 seem to be ok.

The largest relative error was 4.55e-009 in row 1 column 5
Minor itn 6 -- Re-solve QP subproblem.

Maj	Mnr	Step	Objective	Norm Gz	Cond Hz
0	6	0.0e+000	1.196476e-001	0.0e+000	1.0e+000
1	0	2.6e-002	7.655784e-002	0.0e+000	0.0e+000

Exit from NP problem after 2 major iterations, 6 minor iterations.

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Residual
V 1	A UL	9.81000e-003	1.00000e-009	1.00000e-002	0.0000e+000	0.0000e+000
V 2	FR	7.43947e-001	1.00000e-008	1.00000e+001	-2.0120e-012	7.4395e-001
V 3	FR	1.09548e+000	1.00000e-009	1.00000e+001	0.0000e+000	8.0000e-001
V 4	FR	2.95760e+000	1.00000e-010	1.00000e+002	-5.7471e-010	2.9576e+000
V 5	FR	2.20543e-001	1.00000e-009	1.00000e+000	-7.8899e-009	2.2054e-001
V 6	FR	-6.10271e-001	-1.00000e+000	1.00000e+000	3.0790e-012	3.8973e-001
V 7	FR	2.66530e+000	1.00000e-009	1.00000e+002	-6.3225e-010	2.6653e+000
V 8	FR	2.66530e+000	1.00000e-009	1.00000e+002	-6.3225e-010	2.6653e+000

Final objective value = 7.6557842e-002

Analyse des résultats numériques:

— Convergence rapide et Stabilité: L'avantage de cette méthode et qu'on peut observer sur les résultats ci-dessus une convergence très rapide de

l'ordre de 2 majoration et ce en un temps de 3 secondes en moyenne et toujours vers le même point optimal quelque soit la donnée initiale .

— Erreur non optimale: Néanmoins, l'erreur n'est pas du tout satisfaisante

Final objective value = 7.6557842e-002

On peut alors conclure qu'on ne peut pas espérer plus de ce modèle déjà assez restreint; le minimum est atteint, il est unique mais on peut pas admettre une telle erreur de calibration.

Remarque 4.3.1 — *En terme de temps de calcul, si la dimension du vecteur X dépasse 4, alors l'algorithme devient lourd et met beaucoup plus de temps pour converger sur un intervalle plus grand.*

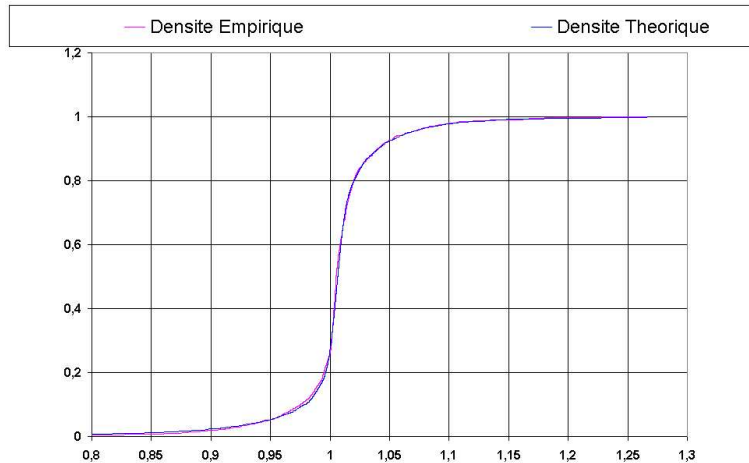
— *La première étape de l'algorithme est de calculer les gradients puis il va suivre la direction où les dérivées sont le plus négatives possible: donc il y'a plus de risque si on ai proche d'un minimum local de converger vers lui que d'aller chercher le bon minimum. Conclusion; on risque d'avoir de mauvais résultats similaires au cas traité ci-dessus.*

On a commencé par réfléchir sur un modèle **GARCH** pour avoir au moins une première approximation pour les paramètres de Vol; ϵ, κ et θ , bien que la volatilité n'était pas observable sur les données historiques. Mais les résultats numériques obtenus n'étaient pas satisfaisants. En effet, l'idée était d'estimer en première étape la vol puis de poser un modèle **GARCH** et le calibrer sur ces dernières estimations, mais les erreurs étaient aussi assez importante.

Une alternative consiste à commencer par des calibrations partielles avant de lancer celle pour calculer tous les paramètres à la fois, c-à-d on va calibrer en un premier temps seulement les paramètres de la volatilité stochastique et par suite corriger avec ceux du Saut. Pourquoi cette stratégie?

En effet, on s'est rendu compte au bout de plusieurs calibrations du modèle à volatilité stochastique seulement que l'algorithme converge toujours vers une même solution assez réelle et ce quelque soit la donnée initiale. L'erreur n'était pas très petite certes, mais en traçant les deux densités théorique et empirique il est claire qu'au voisinage et loin des deux cotés de la monnaie, les deux graphes sont assez proches comme le montre la figure si dessous.

En fixant les paramètres donnés par cette première calibration partielle et en faisant varier ceux du saut, on ce rend compte qu'on ne perd pas les régions ou les deux courbes (i.e. le modèle et le maché) sont assez proches et que sur le reste, cette deuxième calibration modifie totalement la courbure de la densité théorique; intégrer les paramètres du saut corrige ainsi l'erreur de calibration.

FIG. 4.5 – *Densité empirique et celle théorique*

La stratégie finale est alors:

- 1. Estimer V_0 par un modèle de Black Sholes.
- 2. Injecter ce résultat dans une nouvelle estimation d'un modèle à Volatilité stochastique, sans fixer V_0 .
- 3. Injecter une nouvelle fois le résultat de l'étape précédente, sachant qu'on est assez proche du bon minimum, et essayer de corriger l'erreur numérique en rajoutant les paramètres du saut lors de la calibration finale.

Les résultats qu'on obtient cette fois sont très satisfaisants comme on peut le constater sur ce graphe représentant la courbe de densité empirique à partir de la base de données historiques du marché et celle théorique fournie par le modèle ainsi calibré

Le fichier .txt ci-dessous est un deuxième argument:

Parameters to e04unc

```

Number of variables..... 12
Linear constraints..... 0      Nonlinear constraints..... 2

start..... Nag_Cold
step_limit.....2.00e+000      machine precision.....1.11e-016
lin_feas_tol.....1.05e-008    nonlin_feas_tol.....1.05e-008

```

```

optim_tol.....1.00e-009      linesearch_tol.....9.00e-001
crash_tol.....1.00e-002      f_prec.....4.37e-015
inf_bound.....1.00e+020      inf_step.....1.00e+020
max_iter.....      5000      minor_max_iter.....      50
hessian.....      FALSE      h_reset_freq.....      2
h_unit_init.....      FALSE
f_diff_int.....Automatic      c_diff_int.....Automatic
obj_deriv.....      FALSE      con_deriv.....      TRUE
verify_grad.....Nag_CheckObjCon      print_deriv.....Nag_D_Full
print_level.....Nag_Soln_Iter      minor_print_level.....Nag_NoPrint
outfile.....C:\Documents and Settings\mmrad\My Documents\FullDE.txt

```

Verification of the objective gradients.

All objective gradient elements have been set.
Derivative level changed to Nag_AllG_AllJ.

The Jacobian seems to be ok.

The largest relative error was 4.41e-006 in subfunction 1

Component-wise check:

Subfunction 1

Elem.	x[j]	dx[j]	Jacobian value	Difference approxn.	Itns.
1	2.12e+000	1.52e-006	-2.24599671e-002	-2.24599690e-002 OK	2
2	5.00e-003	2.20e-009	-3.21019581e+001	-3.21019587e+001 OK	2
3	6.89e-001	2.74e-007	-2.30262560e-001	-2.30262568e-001 OK	1
4	1.87e+001	4.01e-005	2.87267016e-004	2.87265988e-004 OK	2
5	1.71e+000	9.11e-007	1.69163515e-001	1.69163506e-001 OK	1
6	4.26e+000	3.70e-005	4.65322225e-005	4.65320408e-005 OK	2
7	1.00e-001	4.01e-008	-1.59892170e+000	-1.59892171e+000 OK	2
8	1.70e+000	1.55e-006	-2.19054315e-002	-2.19054325e-002 OK	2
9	-5.82e-001	6.42e-008	-6.20641502e-002	-6.20641493e-002 OK	2
10	4.52e-002	2.91e-009	3.37384821e+000	3.37384825e+000 OK	2
11	5.00e+000	1.94e-006	-4.91900104e-003	-4.91899954e-003 OK	1
12	9.54e-001	3.51e-009	3.40959516e+000	3.40959530e+000 OK	2

12 Jacobian elements out of the 12
set in cols 1 through 12 seem to be ok.

The largest relative error was 3.17e-008 in row 1 column 12
Minor itn 12 -- Re-solve QP subproblem.

Maj	Mnr	Step	Objective	Norm Gz	Cond Hz	
0	13	0.0e+000	8.377965e-003	4.4e-001	1.0e+000	C
1	11	1.9e-003	2.431792e-003	1.6e-002	4.0e+000	
2	10	1.1e-003	2.417476e-003	4.5e-002	1.0e+000	
		...				
		...				
		...				
		...				
998	11	4.9e-004	1.187958e-004	0.0e+000	1.0e+000	
999	12	1.5e-009	1.187958e-004	4.0e-003	1.0e+000	
0	12	7.6e-004	1.187950e-004	0.0e+000	1.0e+000	

Exit from NP problem after 5000 major iterations, 56871 minor iterations.

Varbl	State	Value	Lower Bound	Upper Bound	Lagr Mult	Residual
V 1	FR	3.53685e+000	0.00000e+000	1.20000e+001	-1.2475e-005	3.5369e+000
V 2	FR	9.89914e-003	5.00000e-003	1.00000e+000	-2.2011e-004	4.8991e-003
V 3	FR	5.42341e-001	5.00000e-005	1.00000e+002	8.2461e-005	5.4229e-001
V 4	FR	1.56840e+000	0.00000e+000	1.00000e+002	-1.3860e-008	4.5684e+001
V 5	FR	1.31803e+000	0.00000e+000	2.00000e+000	1.8262e-004	6.8197e-001
V 6	FR	2.68488e+000	0.00000e+000	1.00000e+002	3.4396e-006	2.6849e+000
V 7	FR	5.20049e-002	0.00000e+000	1.00000e+001	6.7754e-004	5.2005e-002
V 8	FR	3.00103e+000	0.00000e+000	1.00000e+001	-3.6641e-005	3.0010e+000
V 9	FR	-7.70284e-001	-1.00000e+000	1.00000e+000	1.5315e-003	2.2972e-001
V 10	FR	3.28247e-002	0.00000e+000	1.00000e+002	1.0865e-003	3.2825e-002
V 11	FR	3.64574e+000	0.00000e+000	1.00000e+002	-3.4939e-006	3.6457e+000
V 12	FR	9.76326e-001	0.00000e+000	1.00000e+000	4.8432e-004	2.3674e-002

4.3.2 Amélioration du temps de calcul

D'un point de vue pratique, il est très important d'optimiser le temps de calcul. Une fois le code optimié, il faut faire très attention à la manière dont on utilise la fonction `nag - opt - nlin - lsq(e04unc)`, car on peut facilement diviser le temps qu'une calibration prend par 5; il suffit de:

- 1. Au lieu de lui fournir comme *Inputs* le vecteur de fonctions $(f_i)_{1 \leq i \leq m}$, un vecteur de données $(y_i)_{1 \leq i \leq m}$ et un autre de *Strikes* $(K_i)_{1 \leq i \leq m}$ et

l'entier m , il est fortement recommandé de lui donner la simple fonction

$$F(x) = \frac{1}{2} \sum_{i=1}^{i=m} [f_i(x) - y_i]^2.$$

Le vecteur des fonctions initialement de taille m devient alors de taille 1, on évitera ainsi de fournir le reste dans l'appel de la fonction **NAG** car toute l'information est déjà dans F .

- 2. Il est important de noter qu'on a pas besoin d'utiliser beaucoup de points du marché pour calibrer le modèle (i.e. m grand) vu qu'il suffit de choisir astucieusement quelques points de la courbe qui réplique bien l'allure générale. Ce-ci évitera de faire d'énormes calculs qui ne rapportent rien (en général, si on calibre n paramètres, il est largement suffisant de prendre $n + 2$ données mais pas moins de n).

Chapitre 5

Appendice

5.1 Calcul de la transformée du Saut Λ

5.1.1 La taille du saut suit une distribution Log-Normale

Calculons d'abord la fonction génératrice G correspondant

$$\begin{aligned} G_J(z) &= \mathbb{E}[e^{zJ}] = \int_{-\infty}^{+\infty} e^{zJ} \phi(J) dJ = \int_{-\infty}^{+\infty} e^{zJ} \frac{1}{\sqrt{2\pi\delta^2}} e^{-\frac{(J-\nu)^2}{2\delta^2}} dJ \\ &= \int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi\delta^2}} e^{\frac{J^2 - J(2\nu + 2\delta^2 z) + \nu^2}{2\delta^2}} dJ = \frac{e^{\nu z + \frac{\delta^2 z^2}{2}}}{\sqrt{2\pi\delta^2}} \int_{-\infty}^{+\infty} e^{\frac{(J - (\nu - \delta^2 z))^2}{2\delta^2}} dJ \\ &= e^{\nu z + \frac{\delta^2 z^2}{2}} \end{aligned}$$

On en déduit le résultat en remarquons: $\Lambda(z) = G_J(-iz) - 1 + izm$
et $m = \mathbb{E}(e^J) - 1 = G_J(1) - 1$.

5.1.2 La taille du saut suit une distribution Double Exponentielle

$$\begin{aligned} G_J(z) &= \mathbb{E}[e^{zJ}] = \int_{-\infty}^{+\infty} e^{zJ} \phi(J) dJ = \int_{-\infty}^{+\infty} e^{zJ} \phi(J) dJ = \int_{-\infty}^{+\infty} e^{zJ} [p \frac{1}{\eta_u} e^{-\frac{J}{\eta_u}} 1_{J \geq 0} + q \frac{1}{\eta_d} e^{-\frac{J}{\eta_d}} 1_{J < 0}] dJ \\ &= \frac{p}{\eta_u} \int_0^{+\infty} e^{-(z + \frac{1}{\eta_u})J} dJ + \frac{q}{\eta_d} \int_{-\infty}^0 e^{-(z + \frac{1}{\eta_d})J} dJ \\ &= \frac{p}{\eta_u} \frac{1}{(-z + \frac{1}{\eta_u})} \int_0^{+\infty} e^{-(z + \frac{1}{\eta_u})J} (-z + \frac{1}{\eta_u}) dJ + \frac{q}{\eta_d} \frac{1}{(z + \frac{1}{\eta_d})} \int_{-\infty}^0 e^{-(z + \frac{1}{\eta_d})J} (-z + \frac{1}{\eta_d}) dJ \end{aligned}$$

$$= \frac{p}{1 - z\eta_u} + \frac{q}{1 + z\eta_d}$$

On en déduit le résultat comme dans le cas précédent: $\Lambda(z) = G_J(-iz) - 1 + izm$

et $m = \mathbb{E}(e^J) - 1 = G_J(1) - 1$.

Remarque 5.1.1 *La fonction génératrice est définie ici pour $-\frac{1}{\eta_d} < Imz < \frac{1}{\eta_u}$*

5.1.3 La taille du saut suit une distribution Double Log-Normale

Le calcul est similaire à celui des deux cas précédents.

5.2 Démonstration de la proposition 3.4.1

On écrit la solution \hat{h} sous la forme suivante $\hat{h}(z, \tau) = e^{(-r - iz(r-d))\tau + A(z, \tau) + B(z, \tau)V}$ sous la condition initiale $A(z, \tau = 0) = B(z, \tau = 0) = 0$, et on va intégrer ceci dans l'EDP vérifié par \hat{h} pour essayer de trouver des équations différentielles dont les fonctionnelles A et B sont les solutions

$$\begin{aligned} \Rightarrow \frac{\partial \hat{h}}{\partial \tau} &= [(-r - iz(r-d)) + \frac{\partial A}{\partial \tau} + \frac{\partial B}{\partial \tau} V] \hat{h}, \frac{\partial \hat{h}}{\partial V} = B \hat{h}, \frac{\partial^2 \hat{h}}{\partial V^2} = B^2 \hat{h} \\ \Rightarrow \frac{\partial A}{\partial \tau} + \frac{\partial B}{\partial \tau} V &= -\frac{1}{2}(z^2 - iz)V + (\kappa(\theta - V) - iz\rho\varepsilon V)B + \frac{1}{2}\varepsilon^2 B^2 \end{aligned}$$

En identifiant les coefficients du polynôme nul pour tout V obtenu par cette dernière équation on obtient le système

$$\left\{ \begin{array}{l} \frac{\partial A}{\partial \tau} = \kappa\theta B \\ \frac{\partial B}{\partial \tau} = -\frac{1}{2}(z^2 - iz) - (\kappa + iz\rho\varepsilon)B + \frac{1}{2}\varepsilon^2 B^2 \end{array} \right.$$

On se rend compte tout de suite que la seconde équation est en fait une équation différentielle de Riccati facile à résoudre en écrivant la solution sous la forme

$$B = -\frac{E'(\tau)}{\frac{\varepsilon^2}{2}E(\tau)} \quad (5.1)$$

On injecte ce ci dans cette dernière on trouve que E vérifie la nouvelle équation différentielle

$$E''(\tau) + (\kappa + iz\rho\varepsilon)E'(\tau) - \frac{1}{4}\varepsilon^2(z^2 - iz)E(\tau) = 0$$

Dont la solution générale s'écrit

$$E(\tau) = A_1 e^{-\frac{1}{2}\psi-\tau} + A_2 e^{-\frac{1}{2}\psi+\tau}$$

$$\psi_{\pm} = \mp(\kappa + iz\rho\varepsilon) + \zeta, \zeta = \sqrt{(\kappa + iz\rho\varepsilon)^2 + \varepsilon^2(z^2 - iz)}$$

Avec la condition initiale

$$\begin{cases} E(0) = C_1 + C_2 \\ E'(0) = -\frac{1}{2}C_1\psi_- + \frac{1}{2}C_2\psi_+ \end{cases}$$

Ce qui nous donne

$$\begin{cases} C_1 = \frac{\psi_+ E(0)}{2S} \\ C_2 = \frac{\psi_- E(0)}{2S} \end{cases}$$

En remplaçant dans l'égalité (4.1) on écrit

$$B = -\frac{E'(\tau)}{\frac{\varepsilon^2}{2}E(\tau)} = \frac{-\frac{1}{2}\psi_- \frac{\psi_+ E(0)}{2S} e^{-\frac{1}{2}\psi-\tau} + \frac{1}{2}\psi_+ \frac{\psi_- E(0)}{2S} e^{\frac{1}{2}\psi+\tau}}{\frac{\varepsilon^2}{2} \left(\frac{\psi_+ E(0)}{2S} e^{-\frac{1}{2}\psi-\tau} + \frac{\psi_- E(0)}{2S} e^{\frac{1}{2}\psi+\tau} \right)}$$

$$= -(z^2 - iz) \frac{-e^{-\frac{1}{2}\psi-\tau} + e^{-\frac{1}{2}\psi+\tau}}{\psi_+ e^{-\frac{1}{2}\psi-\tau} + \psi_- e^{\frac{1}{2}\psi+\tau}}$$

Finalement la fonctionnelle A est obtenu en écrivant a partir du système d'équations différentielles si haut

$$A(z, \tau) = \kappa\theta \int_0^\tau B(s) ds$$

$$\Rightarrow A(z, \tau) = -\frac{2\kappa\theta}{\varepsilon^2} \int_0^\tau \frac{E'(s)}{\frac{\varepsilon^2}{2}E(s)} ds = -\frac{2\kappa\theta}{\varepsilon^2} (\ln(E(S)) |_{s=0}^{s=\tau}) = -\frac{2\kappa\theta}{\varepsilon^2} \ln\left[\frac{E(\tau)}{E(0)}\right]$$

$$= -\frac{2\kappa\theta}{\varepsilon^2} \ln\left[\frac{\psi_+ e^{-\frac{1}{2}\psi-\tau} + \psi_- e^{\frac{1}{2}\psi+\tau}}{2\zeta}\right] = -\frac{\kappa\theta}{\varepsilon^2} \left[\psi_+ \tau + 2 \ln\left(\frac{\psi_- + \psi_+ e^{-\zeta\tau}}{2\zeta}\right) \right]$$

Pour obtenir les expressions de $A(k, \tau)$ et $B(k, \tau)$ il suffit de remplacer z par $k+i/2$.

5.3 Démonstration de la proposition 3.6.1

Comme dans démonstration précédente on écrit la solution sous la forme

$$\hat{h}(z, \tau) = e^{(-r - iz(r-d))\tau - \frac{1}{2}(z^2 - iz)V\tau + C(z, \tau) + D(z, \tau)\lambda}$$

Avec les conditions initiales $C(z, \tau = 0) = D(z, \tau = 0) = 0$, on a en plus

$$\frac{\partial \hat{h}}{\partial \tau} = ((-r - iz(r-d)) - \frac{1}{2}(z^2 - iz)V + \frac{\partial C}{\partial \tau} + \frac{\partial D}{\partial \tau}\lambda)\hat{h}, \quad \frac{\partial \hat{h}}{\partial \lambda} = D\hat{h}, \quad \frac{\partial^2 \hat{h}}{\partial \lambda^2} = D^2\hat{h}$$

En injectons ce ci dans l'EDP vérifier par \hat{h} on a:

$$\frac{\partial C}{\partial \tau} + \frac{\partial D}{\partial \tau}\lambda = \kappa_\lambda(\theta_\lambda - \lambda)D + \frac{1}{2}\varepsilon_\lambda^2 D^2 + \lambda\Lambda(z)$$

En identifiant les termes en λ on aboutit au système

$$\begin{cases} \frac{\partial C}{\partial \tau} = \kappa_\lambda\theta_\lambda D \\ \frac{\partial D}{\partial \tau} = -\kappa_\lambda D + \frac{1}{2}\varepsilon_\lambda^2 D^2 + \Lambda(z) \end{cases}$$

Qui sera résolu de la même manière que dans la démonstration de la proposition 3.4.1

Bibliographie

- [1] Artur Sepp:: Fourier inversion Methods for Option Pricing under Jump-Diffusion Stochastic Volatility and Lévy Process: Master's thesis [Draft,27 October 2002]
- [2] Darrel Duffie, Jan Pan and kenneth Singleton:: Transform Analysis and Asset Pricing for Affine Jump-Diffusions (November 26,1997, August 1,1999)
- [3] George J.Jiang and Roel C.A.Oomen:: Estimating Spot and Integrated Volatility Of Jump-Diffusion Models. (October 24,2003)
- [4] Peter Carr, Hélyette Geman, Marc Yor,Dilip B.Madan:: the fine structure of Asset Returns:An empirical investigation.
- [5] Michael Johannes, Nicholas polson, Jonathan Stroud::Sequentiel parameter Estimation in stochastic Volatility Model wirh Jumps (April 29,2004)
- [6] N.Bellamy, M.Jeanblanc:: Incompleteness of markets driven by a mixed diffusion(1991).
- [7] Jump Diffusion process by Olivier Croissant
- [8] Documentation sur la librairie Nag.
- [9] Numerical recipies