



Options Vanilles et Processus de Lévy

**LAHDHIRI Mohamed Amine,
HARRABI Khalil**

Professeur : Vincent Lemaire

Table des matières

1	Introduction	2
1.1	Énoncé du sujet	2
2	EDS dirigée par un Lévy de mesure infinie	3
2.1	Cadre général	3
2.2	1er cas :	3
2.3	2ème cas :	4
2.4	Résultats	4
3	EDS dirigée par un Lévy de mesure finie	5
3.1	Modèle de Kou	5
3.1.1	Les algorithmes :	5
3.1.2	Schéma d'Euler :	5
3.1.3	1er Pricing avec la formule de Doléans Dade	6
3.1.4	2ème Pricing avec la formule de Doléans Dade	7
3.1.5	Résultats	7
3.2	Modèle de Merton :	7
3.2.1	Choix de paramètres :	8
3.2.2	Algorithme de Pricing :	8
3.2.3	Résultats	8
3.3	Réduction de variance	8
3.3.1	Modèle de Kou	8
3.3.2	Modèle de Merton	9
4	Etude de la convergence :	11
4.1	1ère étude de la convergence :	11
4.1.1	Algorithme :	11
4.1.2	Résultats :	12
4.2	2ème étude de la convergence :	12
4.2.1	Algorithme :	12
4.2.2	Résultat :	13
4.3	Etude de la convergence en moyenne :	13
4.3.1	Algorithme :	13
4.3.2	Résultat :	13
4.4	Moyennisation sur plusieurs n :	14
5	Pricing Fourier	15
5.1	Cadre général	15
5.2	Modèle de Kou :	16
5.3	Modèle de Merton :	16
5.4	Résultats	16
5.4.1	Modèle de Kou :	16
5.4.2	Modèle de Merton :	17

1 Introduction

1.1 Énoncé du sujet

Implémenter la méthode de simulation d'une EDS dirigée par un Lévy proposée dans [\[Rub03\]](#) pour différents types de payoffs et de mesures de Lévy. Étudier la convergence et la stabilité empirique de la procédure utilisée. Comparer le pricing par Monte Carlo à un pricing par Fourier dans les exemples considérés dans [\[TV09\]](#).

2 EDS dirigée par un Lévy de mesure infinie

2.1 Cadre général

Tout d'abord on a commencé par faire notre pricing avec des mesures de Lévy infinie. Dans cette catégorie de mesure nous avons eu deux cas principalement.

Pour pouvoir faire le calcul et implémenter les algorithmes, cette catégorie de mesure a engendré de nouvelles méthodes de troncature et de calcul pour faciliter l'implémentation afin d'étudier notre EDS :

$dX_t = X_t - dZ_t$ avec X un processus de Lévy de triplet caractéristique (b, c, ν)

Dans le cadre général, lors de l'étude d'un processus de Lévy de triplet (b, c, ν) nous avons principalement 4 processus à étudier suivant la troncature $f(x) = x\mathbf{1}_{|x| \leq 1}$ On a donc :

$$\begin{aligned} Z_t &= bt + \sqrt{c}W_t + Z_t^d + Z_t' \\ Z_t^d &= \sum_{s \leq t} \Delta Z_s \mathbf{1}_{|\Delta Z_s| \leq 1} - t \int_{\mathbb{R}} x \mathbf{1}_{|x| \leq 1} \nu(dx) \text{ partie martingale discontinue} \\ Z_t' &= \sum_{s \leq t} \Delta Z_s \mathbf{1}_{|\Delta Z_s| > 1} \\ Z_t &= bt + \sqrt{c}W_t + \sum_{s \leq t} \Delta Z_s \end{aligned}$$

Une des problématique avec les mesure infinie , c'est la densité des temps de sauts de notre processus dans \mathbb{R} et donc il est parfois très difficile de simuler le processus de Lévy car on ne pourra jamais se positionner entre deux temps de sauts à cause de cette densité.

Or dans ce cas, on sait que pour un $M > 0$ et étant dans un intervalle de temps I borné de valeur absolue plus grande que M , les sauts arrivant dans I sont en nombre fini. Donc, si les sauts de Z sont minorés, on peut se positionner entre deux temps de sauts et donc notre EDS devient une EDS classique de la forme : $dX_t = \tilde{b}dt + \sqrt{c}dW_t$. Donc on va toujours travailler sur une fenêtre $[\frac{-1}{u_n}, \frac{1}{u_n}]$ pour isoler les sauts avec u_n une suite qui converge vers 0

$$Z_t^n = \tilde{b}_n t + \sqrt{c}W_t + \sum_{s \leq t} \Delta Z_s \mathbf{1}_{\Delta Z_s \in D_n};$$

$$\tilde{b}_n = b - \int_{D_n} x \mathbf{1}_{|x| \leq 1} \nu(dx); D_n = \mathbb{R} \setminus \left[-\frac{1}{u_n}, \frac{1}{u_n} \right]$$

$dX_t = X_t - dZ_t$ EDS Principale; $dX_t^n = X_t - dZ_t^n$ EDS auxiliaire ; \tilde{X}^n : Schéma de l'EDS auxiliaire dans la troncature, Z^n est un Lévy de mesure finie et est de loi : $\tilde{b}_n t + \sqrt{c}W_t + \sum_{i=1}^{N_t} Y_i$ avec Y_i i.i.d de loi $\frac{\mathbf{1}_{D_n}(x)\nu(dx)}{\theta_n}$ et N processus de poisson d'intensité θ_n avec $\theta_n = \nu(D_n)$.

Pour la simulation, on simule les temps de sauts entre 0 et 1 , ces derniers sont espacés par des variables de loi exponentielle de paramètre θ_n . Ensuite, on construit la grille des instants $t_j^n = \left(t_{j-1}^n + \frac{1}{k_n}\right) \wedge \inf \{T_i^n \mid i \in \mathbb{N} \text{ tel que } T_i^n > t_{j-1}^n\}$ où on calcule la solution. Enfin, entre deux temps de sauts, on a une diffusion classique avec un schéma d'Euler comme celui vu en cours :

$$\tilde{X}_{t_{j+1}^n}^n = \tilde{X}_{t_j^n}^n + \tilde{X}_{t_j^n}^n \tilde{b}_n (t_{j+1}^n - t_j^n) + \tilde{X}_{t_j^n}^n \sqrt{c} (W_{t_{j+1}^n} - W_{t_j^n})$$

puis arrivé au temps de saut $\tilde{X}_{T_{k+1}^n}^n = \tilde{X}_{T_{k+1}^{n-1}}^n + \tilde{X}_{T_{k+1}^{n-1}}^n \Delta Z_{T_{k+1}^n}^n$. Le calcul de la loi des sauts se fait par l'inversion de la fonction de répartition grâce à des lois uniformes comme on va le voir après.

2.2 1er cas :

Pour ce cas nous avons pris la mesure de Lévy suivante :

$$\nu(dx) = \mathbf{1}_{x \neq 0} \frac{1}{|x|^{1+\alpha}} dx$$

Avec $0 < \alpha < 2$ et donc les sauts peuvent prendre n'importe quelle valeur réelle et donc les sauts ne sont pas plus grands que -1 et donc la formule de Doléans Dade n'est plus valable, et on aura après les paramètres suivants :

$$b_n = b; \tilde{b} = b; \theta_n = \frac{2u_n^\alpha}{\alpha}; w_n = \sqrt{\frac{2-\alpha}{2}} u_n^{\frac{2-\alpha}{2}}$$

$$h^{-1}(U) = \begin{cases} \frac{-1}{u_n} (2U)^{-1/\alpha} & \text{si } U < \frac{1}{2} \\ \frac{1}{u_n} (2-2U)^{-1/\alpha} & \text{sinon} \end{cases}$$

Avec $U \stackrel{\text{loi}}{=} \mathcal{U}[0, 1]$

2.3 2ème cas :

Comme le processus précédent n'est pas nécessairement positif, alors maintenant on met la contrainte de positivité afin de pouvoir bien modéliser un prix et donc on a le même triplet caractéristique sauf un petit changement pour la mesure de Lévy :

$$\nu(dx) = \mathbf{1}_{x > -1} \frac{1}{|x|^{1+\alpha}} dx$$

De même, $0 < \alpha < 2$ et donc les sauts peuvent prendre n'importe quelle valeur réelle plus grande que -1 et donc la formule de Doléans Dade est valable, et on aura après les paramètres suivants :

$$b_n = b; \tilde{b} = b; \theta_n = \frac{1}{\alpha} (2u_n^\alpha - 1); w_n = \left(\frac{2}{2-\alpha} u_n^{\alpha-2} \right)^{-1/2}$$

$$h^{-1}(U) = \begin{cases} -(\alpha\theta_n U + 1)^{-1/\alpha} & \text{si } U < \frac{1}{\alpha\theta_n} (u_n^\alpha - 1) \\ (-\alpha\theta_n U + 2u_n^\alpha - 1)^{-1/\alpha} & \text{sinon} \end{cases}$$

Avec $U \stackrel{\text{loi}}{=} \mathcal{U}[0, 1]$

2.4 Résultats

	Strike	mean	Écart type	Borne inférieure	Borne supérieur
Première mesure infinie	80	126.993	91.194	121.341	132.646
Deuxième mesure infinie	80	127.406	87.798	121.964	132.848

TABLE 1 – Mesure infinie

Le tableau 1 présente les résultats de la simulation d'un call avec un strike de 80 et une valeur spot de 100. On peut remarquer que le prix obtenu est supérieur à la valeur de l'actif sous-jacent, ce qui peut s'expliquer par l'utilisation d'une mesure infinie. En effet, ces mesures infinies ont tendance à charger les grands sauts et à avoir des queues de distribution lourdes, ce qui peut occasionner l'apparition de sauts de grandes amplitudes.

3 EDS dirigée par un Lévy de mesure finie

3.1 Modèle de Kou

Considérons le modèle de prix suivant sous la probabilité historique :

$$\begin{aligned} dX_t &= X_{t-} dZ_t; \quad Z_t = bt + \sqrt{c} \hat{W}_t + \sum_{k=1}^{\hat{N}_t} \hat{Y}_k \\ \hat{Y}_k \text{ i.i.d} &\stackrel{\text{loi}}{=} \nu(dx); \hat{N}_t \stackrel{\text{loi}}{=} \mathcal{P}(t\lambda); \hat{W}^{\text{Phist}} m.b. \\ \nu(dx) &= \left[p \frac{\lambda_+}{(1+x)^{\lambda_++1}} 1_{x>0} + (1-p) \frac{\lambda_-}{(1+x)^{1-\lambda_-}} 1_{-1<x<0} \right] dx \\ \lambda_+ &> 1; \quad \lambda_- > 0; 0 < p < 1 \end{aligned}$$

Sous la probabilité risque neutre, qu'on justifiera juste après, on a :

$$\begin{aligned} dX_t &= X_{t-} dZ_t \\ Z_t &= \left(r - \lambda \int_{\mathbb{R}} (e^x - 1) G(dx) \right) t + \sqrt{c} W_t + \sum_{k=1}^{N_t} Y_k \end{aligned}$$

Sous la Proba R.N. Y_k i.i.d $\stackrel{\text{loi}}{=} F(dx); N_t \stackrel{\text{loi}}{=} \mathcal{P}(t\lambda); W \mathbb{P}^{R.N.} \cdot m.b.$

$$\begin{aligned} \nu(dx) &= \left[p \frac{\lambda_+}{(1+x)^{\lambda_++1}} 1_{x>0} + (1-p) \frac{\lambda_-}{(1+x)^{1-\lambda_-}} 1_{-1<x<0} \right] dx \\ G(dx) &= \left(p \lambda_+ e^{-\lambda_+ x} 1_{x>0} + (1-p) \lambda_- e^{-\lambda_- |x|} 1_{x<0} \right) dx \\ \lambda_+ &> 1; \quad \lambda_- > 0; 0 < p < 1 \end{aligned}$$

$$\text{Simplification : } Z_t = \left(r - \lambda \left[p \frac{\lambda_+}{\lambda_+ - 1} + (p-1) \frac{\lambda_-}{\lambda_- + 1} - 1 \right] \right) t + \sqrt{c} W_t + \sum_{i=1}^{N_t} Y_k$$

On s'intéresse dans ce qui suit au pricing des calls qui sont les options européennes classiques et des log-contrat $-2 \log(\frac{S_T}{S_0})$ qui sont des contrats qui permettent l'approximation des volatilités stochastiques d'après notre cours de calibration de S.De Marco.

3.1.1 Les algorithmes :

Pour l'implémentation du modèle de Kou on a utilisé trois modèles qui se basent sur :

- Pricing avec le schéma d'Euler
- 1er Pricing avec la formule de Doléans Dade avec la grille de sauts
- 2ème Pricing avec la formule de Doléans Dade

3.1.2 Schéma d'Euler :

Ici on n'utilise pas la formule de Doléans Dade pour les processus de sauts. Notre algorithme se présente comme suit :

- 1- On construit une grille pour les sauts
- 2 On construit la fonction h^{-1} afin de pouvoir simuler des sauts ΔZ à partir des lois uniformes. Ainsi $h^{-1}(U[0,1]) \stackrel{\text{loi}}{=} F(dx)$ et on a donc :

$$h^{-1}(U) = \begin{cases} \left(\frac{U}{1-p} \right)^{\frac{1}{\lambda_-}} - 1 & \text{si } U < 1-p \\ \left(\frac{p}{1-U} \right)^{\frac{1}{\lambda_+}} - 1 & \text{sinon} \end{cases}$$

- Construction de la fonction de pricing MC avec le schéma d'Euler qui retourne le prix d'une option de payoff donnée, ainsi que l'erreur Monte Carlo $\frac{1.96\hat{\sigma}}{\sqrt{M}}$

- Poser $b = r - \lambda \left[p \frac{\lambda_+}{\lambda_+ - 1} + (p - 1) \frac{\lambda_-}{\lambda_- + 1} - 1 \right]$
- pour $j = 0$ à M :
 - ré-initialisation $x = x_0$, saut = 0, $h = 0$, ré-initialisation de la grille
 - pour $i = 1$ à taille de la grille :
 - $h = \text{grid}[i][0] - \text{grid}[i - 1][0]$
 - $x = x * \exp(b * h + \mathcal{N}(0, 1) * \sqrt{c * h})$
 - si $\text{grid}[i][1] == 1$ (t_i est un temps de saut : simuler un saut puis $x = x(1 + \text{saut})$)
 - enregistrer la valeur du payoff évalué en x
 - actualiser la moyenne empirique ainsi que l'écart type empirique
- retourner la moyenne, l'écartype, et l'intervalle de confiance

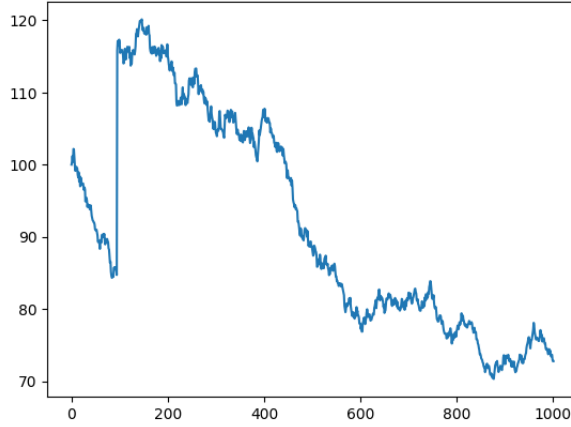


FIGURE 1 – Réalisation d'une simulation d'un schéma d'euler

3.1.3 1er Pricing avec la formule de Doléans Dade

Ici l'incrémentation normale (Brownienne) ne se fait qu'à l'instant finale ce qui est différent du schéma d'Euler. On utilise donc la formule explicite de Doléans Dade :

$$X_t = X_0 e^{\left(r - c/2 - \lambda \left[p \frac{\lambda_+}{\lambda_+ - 1} + (p - 1) \frac{\lambda_-}{\lambda_- + 1} - 1 \right] \right) t + \sqrt{c} W_t} \prod_{k=1}^{\mathcal{P}(\lambda t)} (1 + Y_k)$$

$$Y_k \text{ i.i.d } \stackrel{\text{loi}}{=} F(dx)$$

- $b = \left(r - c/2 - \lambda \left[p \frac{\lambda_+}{\lambda_+ - 1} + (p - 1) \frac{\lambda_-}{\lambda_- + 1} - 1 \right] \right)$
- pour $j = 0$ à M :
 - ré-initialisation $x = x_0$, saut = 0, N réalisation de poisson de paramètre λ
 - pour $i = 1$ à N :
 - U = uniforme sur $[0, 1]$
 - saut = $h^{-1}(U)$
 - $x = x(1 + \text{saut})$
 - $x = x * \exp(b + \mathcal{N}(0, 1) * \sqrt{c})$
 - enregistrer la valeur du payoff évalué en x
 - actualiser la moyenne empirique ainsi que l'écart type empirique
- retourner la moyenne, l'écartype, et l'intervalle de confiance

3.1.4 2ème Pricing avec la formule de Doléans Dade

Ici on travaille un peu différemment de ce qu'on a déjà fait dans l'algorithme précédent. En effet comme on a des sauts positifs avec probabilité p et des sauts négatifs avec probabilité $1 - p$ alors on décide d'utiliser une variable binomiale afin de mettre en oeuvre l'aspect aléatoire des sauts. Ainsi l'algorithme se transforme légèrement comme suit :

$$L_k \text{ i.i.d } \stackrel{\text{loi}}{=} G(dx) = \left(p\lambda_+ e^{-\lambda_+ x} 1_{x>0} + (1-p)\lambda_- e^{-\lambda_- |x|} 1_{x<0} \right) dx$$

La loi $G(dx)$ est une loi de mélange et s'écrit :

$$L_1 \stackrel{\text{loi}}{=} -H \frac{\log(U)}{\lambda_+} + (1-H) \frac{\log(U)}{\lambda_-} \stackrel{\text{loi}}{=} -1_{W < p} \frac{\log(U)}{\lambda_+} + (1 - 1_{W < p}) \frac{\log(V)}{\lambda_-}$$

- $b = \left(r - c/2 - \lambda \left[p \frac{\lambda_+}{\lambda_+ + 1} + (p-1) \frac{\lambda_-}{\lambda_- + 1} - 1 \right] \right)$
- pour $j = 0$ à M :
- ré-initialisation $x = x_0$, saut = 0, N relâsation de poisson de paramètre λ
- pour $i = 1$ à N :
 - U = uniforme sur $[0,1]$
 - H = Bernoulli de paramètre p
 - saut = saut * $\left(H * \frac{-\log(U)}{\lambda_+} + (1-H) * \frac{-\log(U)}{\lambda_-} \right)$
 - $x = x * \exp(b + \mathcal{N}(0,1) * \sqrt{c} + \text{saut})$
 - enregistrer la valeur du payoff évalué en x
 - actualiser la moyenne empirique ainsi que l'écart type empirique
- retourner la moyenne, l'écartype, et l'intervalle de confiance

3.1.5 Résultats

	Strike	mean	Écart type	Borne inférieure	Borne supérieur
Schéma d'Euler	80	45.195	200.645	41.262	49.127
	100	36.739	207.419	32.673	40.804
	120	31.819	203.367	27.833	35.805
Doléans-Dade	80	44.734	248.195	39.869	49.599
	100	35.353	150.469	32.403	38.302
	120	30.734	173.841	27.326	34.141
Doléans-Dade méthode 2	80	41.169	164.588	37.943	44.395
	100	37.123	178.756	33.619	40.627
	120	31.617	192.773	27.838	35.395

TABLE 2 – Résultats de simulation call européen

Le tableau 2 regroupe les résultats des 3 méthodes décrites dans cette section. Nous avons remarqué que la méthode du schéma d'Euler est la plus coûteuse en termes de temps de calcul. Ceci est prévisible car on simule le schéma à chaque itération.

3.2 Modèle de Merton :

Dans le modèle de merton on considère la dynamique suivante :

$$X_t = X_0 e^{\left(r - \frac{\varepsilon}{2} - \lambda \int_{\mathbb{R}} (e^x - 1) \mathcal{N}(\mu, \delta^2)(dx) \right) t + \sqrt{c} W_t + \sum_{k=1}^{N_t} Y_k}$$

$$Y_k \text{ i.i.d } \stackrel{\text{loi}}{=} \mathcal{N}(\mu, \delta^2); N_t \stackrel{\text{loi}}{=} \mathcal{P}(\lambda t)$$

Ainsi on aura la formule suivante pour la dynamique :

$$X_t = X_0 e^{\left(r - \frac{c}{2} + \lambda - \lambda e^{\delta^2/2}\right)t + \sqrt{c}W_t + \sum_{k=1}^{N_t} Y_k}$$

Cette formule simplifié nous donne que $e^{-rt}X_t$ est une martingale et donc on est dans une vision risque-neutre ce qui est cohérent avec nos modalités de pricing.

3.2.1 Choix de paramètres :

Après avoir défini le modèle il nous reste donc la fixation de certains paramètres. En effet, comme on a choisit une distribution normale des sauts alors ceux ci vont avoir une certaine moyenne et une variance.

Pour la moyenne on peut dire qu'on voudrais avoir avec les mêmes probabilités des sauts négatifs et positifs et donc on peut prendre $\mu = 0$

Pour la variance, comme on a une exponentielle des sauts alors pour avoir une valeur raisonnable des sauts on peut prendre $\delta = 0.2$ car si on fait un peu de calcul sur les sauts on a :

$$\Delta X_t = X_t - X_{t-} = X_{t-}(e^{\mathcal{N}(0,1)} - 1)$$

Et donc pour éviter des sauts très grand et avoir des valeurs raisonnables on peut prendre une variance assez faible et donc $\delta = 0.2$ serait une bonne approximation

3.2.2 Algorithme de Pricing :

Ici on simule toutes les variables à la maturité et donc on a une seule boucle qui concerne les simulations Monte-Carlo. Le modèle de Merton présente une particularité qui consiste à avoir des variables de sauts de loi normale centré mais avec une certaine variance δ .

Une autre particularité du modèle est que la somme $\sum_{k=1}^{N_t} Y_k$ peut être simulé grâce à $\delta\sqrt{N}\mathcal{N}(0,1)$ ce qui diminue la complexité en mémoire et en calcul. L'algorithme se présente comme suit :

- $b = r - \frac{c}{2} + \lambda - \lambda e^{\delta^2/2}$
- pour $j = 0$ à M :
 - ré-initialisation $x = x_0$, saut = 0, N réalisation de poisson de paramètre λ , $Z1$ et $Z2$ réalisations $\mathcal{N}(0,1)$
 - $x = x * \exp(b + \sqrt{c}Z1 + \delta * \sqrt{N}Z2)$
 - enregistrer la valeur du payoff évalué en x
 - actualiser la moyenne empirique ainsi que l'écart type empirique
 - retourner la moyenne, l'écartype, et l'intervalle de confiance

3.2.3 Résultats

Dans le tableau 4 Nous regroupons le prix des Calls européens pour le modèle de merton pour $\delta = 0.2$.

3.3 Réduction de variance

3.3.1 Modèle de Kou

$$\begin{aligned} \text{On a } X_1 &= X_0 e^{b + \sqrt{c}Z + \sum_{k=1}^N \left(-H \frac{\log(U_k)}{\lambda_+} + (1-H) \frac{\log(U_k)}{\lambda_-} \right)} \\ \text{Avec } b &= r - c/2 - \lambda \left[p \frac{\lambda_+}{\lambda_+ - 1} + (1-p) \frac{\lambda_-}{\lambda_- + 1} - 1 \right] \\ Z &\stackrel{\text{loi}}{=} \mathcal{N}(0,1); N \stackrel{\text{loi}}{=} \mathcal{P}(\lambda); H_k \stackrel{\text{loi}}{=} \mathcal{B}(1,p); U_k, V_k \text{ i.i.d } \stackrel{\text{loi}}{=} \mathcal{U}([0,1]) \end{aligned}$$

En raison de sa forme particulière et de l'indépendance des variables, la méthode de préconditionnement suivante peut être appliquée au payoff du call :

$$\begin{aligned}
g(N, H, U, V) &\triangleq \sum_{k=1}^N \left(-H_k \frac{\log(U_k)}{\lambda_+} + (1 - H_k) \frac{\log(V_k)}{\lambda_-} \right) \\
X_0 e^{b+c/2+g(N, H, U, V)} &= \widetilde{X}_0(N, H, U, V) \\
E \left[e^{-r} (X_1 - K)_+ \right] &= E \left[e^{-r} \left(\widetilde{X}_0(N, H, U, V) e^{\sqrt{c}Z - c/2} - K \right)_+ \right] \\
&= E \left[E \left[e^{-r} \left(\widetilde{X}_0(n, h, u, v) e^{\sqrt{c}Z - c/2} - K \right)_+ \right]_{n=N, u=U, v=V, h=H} \right] \\
&= E \left[\text{Call_BS}(\widetilde{X}_0(N, H, U, V), K, c, r, T) \right]
\end{aligned}$$

Considérons maintenant les paramètres suivants pour le modèle de Kou : $p = 0, 5$, $\lambda_+ = 3$, $\lambda_- = 6$, $\lambda = 4$. En prenant un spot égal à 100, $c = 0.04$ et $M=10000$, nous obtenons les résultats suivants :

	Strike	Sans réduction de variance				Réduction de variance			
		mean	Écart type	Borne inférieure	Borne supérieure	mean	Écart type	Borne inférieure	Borne supérieure
Doléans-Dade méthode 2	80	41.169	164.588	37.943	44.395	42.022	147.196	39.137	44.907
	100	37.123	178.756	33.619	40.627	33.577	148.173	35.710	36.481
	120	31.617	192.773	27.838	35.395	32.699	161.973	27.133	35.874

TABLE 3 – comparaisons des prix et des intervalles de confiance avec et sans reduction de variance

Le tableau 3 présente les prix et les intervalles de confiance obtenus pour le modèle de Kou à l'aide du deuxième algorithme décrit précédemment. Nous sommes parvenus à réduire la variance dans ce modèle de Kou, cependant il reste difficile à exploiter car il nécessite de bien choisir trois paramètres λ , λ_+ et λ_- .

Il convient de noter que la réduction de variance n'a été appliquée que pour ce deuxième algorithme, car la même méthode peut être utilisée pour les autres algorithmes.

3.3.2 Modèle de Merton

Lorsqu'il s'agit de calculer le payoff d'un call, exprimé par la formule : $\text{payoff}(X_1) = \text{payoff}(X_0 e^{(b+\sqrt{c}Z+\delta\sqrt{N}\tilde{Z})})$, où Z et \tilde{Z} suivent une loi normale $\mathcal{N}(0, 1)$ et N suit une loi de Poisson de paramètre λ , avec $b = r - \frac{c}{2} + \lambda - \lambda e^{\delta^2/2}$, il est possible de proposer la méthode suivante pour réduire la variance :

$$E[\text{payoff}(X_1)] = E \left[\frac{\frac{f(N, Z, \tilde{Z}) + f(N, -Z, \tilde{Z})}{2} + \frac{f(N, Z, -\tilde{Z}) + f(N, -Z, -\tilde{Z})}{2}}{2} \right]$$

avec $\text{payoff}(X_1) = f(N, Z, \tilde{Z})$

Pour le payoff du modèle de Merton on peut utiliser la méthode de pré-conditionnement :

$$\begin{aligned}
E \left[e^{-r} \left(X_0 e^{b+\sqrt{c}Z+\delta\sqrt{N}\tilde{Z}} - K \right)_+ \right] &= E \left[E \left[e^{-r} \left(X_0 e^{b+c/2+\sqrt{n}\tilde{z}} e^{\sqrt{c}Z - c/2} - K \right)_+ \right]_{n=N, \tilde{z}=\tilde{Z}} \right] \\
&= E \left[\text{Call_BS}(\widetilde{X}_0(N, \tilde{Z}), K, c, r, T) \right]
\end{aligned}$$

Ainsi en utilisant la méthode de la variable antithétique nous obtenons :

$$\begin{aligned}
E \left[e^{-r} \left(X_0 e^{b+\sqrt{c}Z+\sqrt{N}\tilde{Z}} - K \right)_+ \right] &= E \left[\frac{\Xi + \tilde{\Xi}}{2} \right] \\
\text{Avec } \Xi &= e^{-r} \left(X_0 e^{b+c/2+b\sqrt{N}\tilde{Z}} \mathcal{N}(d_1(N, \tilde{Z})) - K \mathcal{N}(d_0(N, \tilde{Z})) \right) \\
\tilde{\Xi} &= e^{-r} \left(X_0 e^{b+c/2-\delta\sqrt{N}\tilde{Z}} \mathcal{N}(d_1(N, -\tilde{Z})) - K \mathcal{N}(d_0(N, -\tilde{Z})) \right)
\end{aligned}$$

Donc il suffit de simuler plusieurs $\frac{\Xi+\tilde{\Xi}}{2}(\omega)$ et prendre leur moyenne.

	Strike	Sans réduction de variance				Réduction de variance			
		mean	Écart type	Borne inférieure	Borne supérieur	mean	Écart type	Borne inférieure	Borne supérieur
Modèle de Merton	80	29.159	42.783	28.321	29.998	28.742	19.892	28.352	29.132
	100	18.835	39.008	18.071	19.600	17.449	20.577	17.046	17.852
	120	11.630	32.834	10.987	12.274	10.619	19.612	10.234	11.003

TABLE 4 – comparaisons des prix et des intervalles de confiance avec et sans reduction de variance

Dans le tableau 4, nous avons regroupé les résultats des simulations des prix avec et sans réduction de variance. Nous avons réussi à réduire considérablement la variance pour le modèle de Merton. L'implémentation du modèle de Merton semble plus simple, car contrairement au modèle de Kou, on a un seul paramètre δ à déterminer.

4 Etude de la convergence :

L'article montre qu'un bon choix pour la convergence en terme de précision et de nombre d'itérations est donné par k_n telle que :

$$k_n = \begin{cases} 1 & \text{si } \frac{\theta_n}{b_n^2} \wedge \frac{\sqrt{\theta_n}}{c} \geq w_n \\ \sup(b_n^2 w_n, c^2 w_n^2) & \text{sinon} \end{cases} \quad w_n = \left(\int_{|x| < 1/u_n} x^2 F(dx) \right)^{-\frac{1}{2}}$$

Dans cette configuration, on aura que le schéma \bar{X}^n de l'EDS auxiliaire converge vers la solution X de l'EDS avec une vitesse w_n . En effet, $w_n(X - \bar{X}^n)$ converge en loi par rapport à la topologie de Skorohod vers un processus non nul.

En second lieu, selon le papier de S.RUBENTHALER, on sait que la convergence en loi du processus dans $[0,1]$ est équivalente à la convergence en loi de $w_n(X - \bar{X}^n)$.

Pour l'étude de la convergence nous avons travaillé sur 3 méthodes qui se basent principalement sur la même idée qui de faire le calcul de convergence, en premier lieu, et visualiser l'histogramme de la loi limite et voir qu'il présente une loi non nulle ce qui justifie les résultat de convergence selon S.RUBENTHALER.

4.1 1ère étude de la convergence :

4.1.1 Algorithme :

La première méthode se base sur un résultat asymptotique : $w_N(X - \bar{X}^N) \approx w_N(X^N - \bar{X}^N)$ avec N le nombre d'itération d'un schéma d'Euler qui est assez grand et donc on a pour X^N :

$$X_t^N = X_0 e^{Z_t - \frac{1}{2}(Z^c)_t} \Pi_{s \leq t} (1 + \Delta Z_s) e^{-\Delta Z_s} 1_{\Delta Z_s \in D_N} = X_0 e^{\bar{b} - \frac{\epsilon}{2} + \sqrt{c}W_1} \Pi_{s \leq t} (1 + \Delta Z_s) 1_{\Delta Z_s \in D_N}$$

Ici on va étudier la convergence par évènement et donc on va étudier la limite $w_N(X^N(\omega) - \bar{X}^N(\omega))$:

Construction de la fonction "difference" qui retourne un vecteur de M valeurs $w_N(X_1^N(\omega) - \bar{X}_1^N(\omega))$, sa moyenne son écart type.

- - pour $j = 0$ à M :
 - réinitialisation $x = x_0$, produit = x_0 , saut = 0, $h = 0$, $w_1 = 0$, epsilon = 0, réinitialisation de la grille
 - pour $i = 1$ à taille de la grille
 - \star epsilon = grid $[i][1]$, vaut 1 si saut, 0 sinon ; $h = \text{grid}[i][0] - \text{grid}[i-1][0]$
 - $\star W = W + \text{increment}W$
 - $\star x = x * (1 + \text{bn} * h + \text{increment}W)$
 - si eps=1 et donc on a un saut, $x = x * (1 + \text{saut})$ et $\bar{x} = \bar{x} * (1 + \text{saut})$
 - enregistrer la valeur $w_N(\bar{x} * e^{\bar{b} - \frac{\epsilon}{2} + W} - x)$
- retourner toutes les valeurs et tracer l'histogramme

4.1.2 Résultats :

On voit donc l'histogramme suivant de la distribution limite qui présente une distribution non nulle ce qui est conforme avec le papier de [Rub03] ,

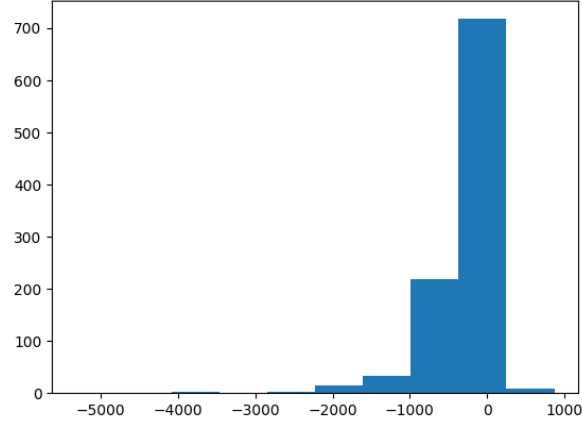


FIGURE 2 – histogramme de la 1ère méthode

4.2 2ème étude de la convergence :

4.2.1 Algorithme :

$$w_N \left(X_1^N - \tilde{X}_1^N \right) = w_N \left(X_1^{c,N} - \tilde{X}_1^{c,N} \right) \Pi_{s \leq 1} (1 + \Delta Z_s) 1_{\Delta Z_s \in D_N} \text{ Avec}$$

$$dX_t^{c,N} = X_t^{c,N} d[b_N t + \sqrt{c} dW_t] \text{ et } \tilde{X}^{c,N} \text{ son schéma d'Euler}$$

$$w_N \left(X_1^N - \tilde{X}_1^N \right) = \frac{w_N}{\sqrt{\max(t_{i+1}^N - t_i^N)}} \sqrt{\max(t_{i+1}^N - t_i^N)} \left(X_1^{c,N} - \tilde{X}_1^{c,N} \right) \Pi_{s \leq 1} (1 + \Delta Z_s) 1_{\Delta Z_s \in D_N}$$

Or dans les schéma d'Euler l'erreur forte de $X_1^{c,N} - \tilde{X}_1^{c,N}$ est d'ordre $\frac{1}{\sqrt{\max(t_{i+1}^N - t_i^N)}}$ donc il suffit de montrer que la variable aléatoire $w_N \sqrt{\max(t_{i+1}^N - t_i^N)} \Pi_{s \leq 1} (1 + \Delta Z_s) 1_{\Delta Z_s \in D_N}$ converge en loi vers une variable finie non nulle, donc on peut étudier la limite en loi de $w_n \sqrt{\frac{1}{\theta_n} \wedge \frac{1}{k_n}} \Pi_{i=1}^{\text{Poisson}(\theta_n)} (1 + Y_i)$ avec Y_i i.i.d de loi $\frac{1_{D_n}(x) \nu(dx)}{\theta_n}$

- Construction de la fonction h^{-1} qui est l'inverse de la fonction de répartition de la loi des sauts ΔZ^n . Ainsi $h^{-1}(\mathcal{U}[0, 1]) \stackrel{\text{loi}}{=} \frac{1_{D_n}(x) \nu(dx)}{\theta_n}$.
- Construction de la fonction "methode2" qui retourne un vecteur de M valeurs $w_n \sqrt{\frac{1}{\theta_n} \wedge \frac{1}{k_n}} \Pi_{i=1}^{\text{Poisson}(\theta_n)} (1 + Y_i)$, sa moyenne et son écart-type empirique
- pour $j = 0$ à M :
 - ré-initialisation $\text{saut} = 0, x = 1, \text{nbsaut} = \text{réalisation d'une variable de poisson de paramètre } \theta_n$
 - pour $i = 1$ à nbsaut : $U = \mathcal{U}[0, 1], x = x (1 + h^{-1}(U))$
 - enregistrer la valeur $w_n \sqrt{\frac{1}{\theta_n} \wedge \frac{1}{k_n}} x$
 - actualiser la moyenne empirique ainsi que l'écart type empirique
- retourner les valeurs avec l'histogramme

4.2.2 Résultat :

On voit aussi que l'histogramme montre une distribution d'une loi non nulle ce qui est conforme avec le papier de [Rub03]

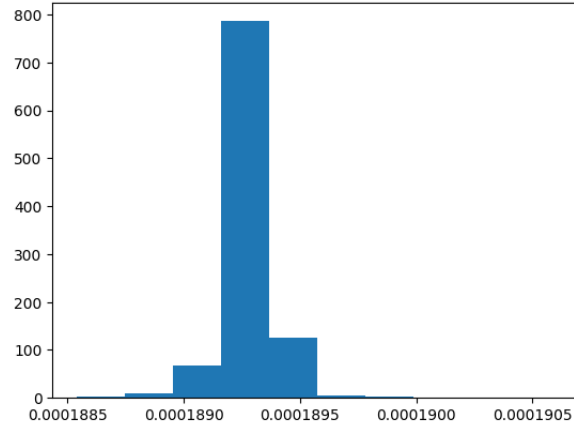


FIGURE 3 – histogramme de la 2ème méthode

4.3 Etude de la convergence en moyenne :

Les deux méthodes précédentes nous obligent à avoir le même ω et le même N pour les deux calculs qu'on fait. Une autre méthode pour évaluer la convergence est d'étudier pour $N \neq n$ $w_n(\mathbb{E}[X^N] - \mathbb{E}[\tilde{X}^n])$ et donc on ne se limite pas à un seul évènement mais on s'intéresse à une moyennisation sur tout les cas possibles. Et donc on fixe N grand fixé et fait varier n jusqu'à N .

4.3.1 Algorithme :

Construction de la fonction cv-esperance dépendant entre autre de n et N qui retourne une approximation de $w_n(\mathbb{E}[X^N] - \mathbb{E}[\tilde{X}^n])$

- pour $j = 0$ à M :
 - ré-initialisation $x = x_0$, $nbsaut =$ réalisation d'une variable de poisson de paramètre θ_N
 - pour $i = 1$ à $nbsaut$: $U = \mathcal{U}[0, 1]$, $x = x(1 + h^{-1}(U))$
 - enregistrer la valeur $xe^{\tilde{b}-c/2+\sqrt{c}\mathcal{N}(0,1)}$ dans un tableau. (Il s'agit de $X_1^N(\omega)$)
 - calculer la moyenne des valeurs enregistrées et obtention de $\mathbb{E}[X_1^N]$.
- pour $j = 0$ à M :
 - ré-initialisation $x = x_0$, $saut = 0$, $h = 0$, ré-initialisation de la grille
 - pour $i = 1$ à taille de la grille :
 - $h = \text{grid}[i][0] - \text{grid}[i-1][0]; \mathcal{UN}(0, 1)$
 - $x = x * (1 + bn * h + \mathcal{N}(0, 1) * \text{sqrt}(c * h))$
 - si t_i^n est un temps de saut : simuler un saut avec $saut = h^{-1}(U)$ puis $x = x * (1 + h^{-1}(U))$
 - enregistrer la valeur x
 - calculer la moyenne et avoir $\mathbb{E}[\tilde{X}^n]$
 - retourner la valeur $w_n(\mathbb{E}[X_1^N] - \mathbb{E}[\tilde{X}_1^n])$

4.3.2 Résultat :

On voit que la valeur de l'espérance n'est pas nulle et donc on peut dire que notre variable limite n'est donc pas nulle ce qui garantie une convergence au sens de Skorohod

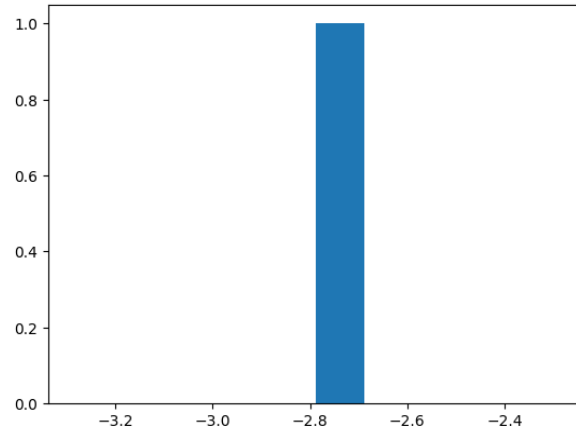


FIGURE 4 – exécution pour une valeur de l'espérance

4.4 Moyennisation sur plusieurs n :

Ici on va fixer N et on fait varier n jusqu'à N et donc on va faire la fonction précédente pour plusieurs valeurs de n et voir le comportement de la convergence. On a donc les résultats suivants sur la valeur de l'espérance en fonction de n ce qui ne présente pas une forme particulière mais garantie une convergence d'après le papier de [\[Rub03\]](#)

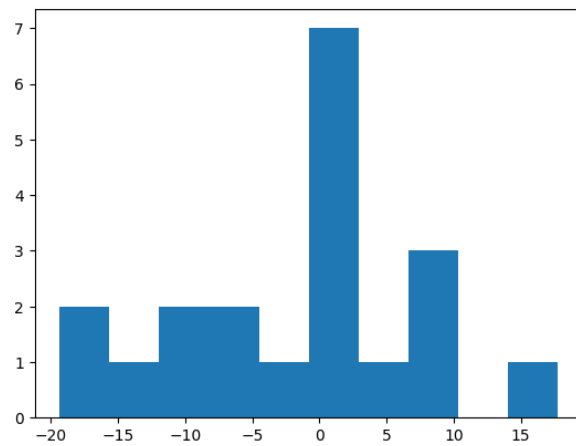


FIGURE 5 – histogramme pour plusieurs valeurs de n

5 Pricing Fourier

Pour des mesures finies, il existe dans l'article [TV09] une méthode rapide et efficace pour calculer les prix des Calls pour différents strikes, grâce à l'algorithme de la transformée de Fourier.

5.1 Cadre général

$$e^{-rt}X_t = X_0 e^{\left(-\frac{c}{2} - \lambda \int_{\mathbb{R}} (e^x - 1)G(dx)\right)t + \sqrt{c}W_t + \sum_{k=1}^{N_t} L_k} \quad L_k \text{ i.i.d } \stackrel{loi}{=} G(dx); N_t \stackrel{loi}{=} \mathcal{P}(t\lambda)$$

On a bien $e^{-rt}X_t = e^{M_t}$ est une martingale et il suffit donc de vérifier qu'il existe $\alpha > 0$ tel que $\int_{x>1} (e^{(1+\alpha)x})G(dx) < \infty$ pour qu'on soit dans le cadre de la proposition 1 page 9 de [TV09].

Maintenant, si on considère un call européen de la forme :

$$\begin{aligned} C(k) &= \mathbb{E} \left[e^{-r} (X_1 - e^k)_+ \right] = e^{-r} E \left[(X_0 e^{r+M_1} - e^k)_+ \right] \\ &= X_0 e^{-rt} E \left[\left(e^{r+M_1} - e^{k-\ln(X_0)} \right)_+ \right] = X_0 C(\tilde{k}) \\ \tilde{k} &= k - \ln(X_0) \end{aligned}$$

Grâce à l'article de P.Tankov, on comprend qu'en calculant la transformée de Fourier du call on ne peut pas aboutir à des résultats car le call n'est pas intégrable et donc il ne satisfait pas les critères d'analyse de Fourier. Cependant, l'idée est de calculer la transformée de Fourier de la différence :

$$z(k) = C(k) - C_{BS}^{\Sigma}(k)$$

Ainsi la transformée de Fourier de z est $\zeta(v)$ qui a la formule suivante :

$$\zeta(v) = e^{ivr} \left(\frac{\phi(v-i) - \exp\left(-\frac{\Sigma^2}{2}((v-i)^2 + i(v-i))\right)}{iv(1+iv)} \right)$$

Sachant que le second terme est la transformée de Fourier du modèle de Black-Scholes :

$$\Phi_{BS}(v) = \exp\left(-\frac{\Sigma^2}{2}(v^2 + iv)\right)$$

Comme dans l'article de P.Tankov, on a pris $\Sigma = 0.2$ pour satisfaire la condition de nullité de ζ à l'origine. On a donc les résultats suivants :

$$\begin{aligned} C(\tilde{k}) &= C_{BS}^{\Sigma}(\tilde{k}) + \frac{1}{2\pi} \int_{-\infty}^{\infty} e^{-iv\tilde{k}} \zeta(v) dv \\ \zeta(v) &= e^{ivr} \left(\frac{\phi(v-i) - \exp\left(-\frac{\Sigma^2}{2}((v-i)^2 + i(v-i))\right)}{iv(1+iv)} \right) \\ \phi(u) &= E \left[e^{iuM_1} \right] = e^{iu\left(-\frac{c}{2} - \lambda \int_{\mathbb{R}} (e^x - 1)G(dx)\right) - \frac{cu^2}{2} + \lambda \int_{\mathbb{R}} (e^{iux} - 1)G(dx)} \\ C(\tilde{k}) &\approx C_{BS}^{\Sigma}(\tilde{k}) + \frac{L}{2\pi(N-1)} \sum_{m=0}^{N-1} w_m \zeta(-L/2 + m\Delta) e^{-i\tilde{k}(-L/2 + m\Delta)} \\ w_m &= \begin{cases} 0 & \text{si } m = 0 \\ 4/3 & \text{si } m \text{ impair} \\ 2/3 & \text{sinon} \end{cases} \end{aligned}$$

Plusieurs log-strikes : $\tilde{k}_n = \tilde{k}_0 + \frac{2\pi n}{N\Delta}$; $\Delta = \frac{L}{N-1}$; N et L grands et N une puissance de 2

$$R_n = \sum_{m=0}^{N-1} w_m \zeta(-L/2 + m\Delta) e^{-i\tilde{k}_0 m \Delta} e^{-\frac{2i\pi n m}{N}}$$

$$C(\tilde{k}_n) \approx C_{BS}^{\Sigma}(\tilde{k}_n) + \frac{L}{2\pi(N-1)} e^{i\tilde{k}_n L/2} R_n$$

$$\text{Enfin } C(k_n) = X_0 C(\tilde{k}_n)$$

5.2 Modèle de Kou :

On a pour le modèle de Kou, $f(dx) = (p\lambda_+ e^{-\lambda_+ x} \mathbf{1}_{x>0} + (1-p)\lambda_- e^{-\lambda_- |x|} \mathbf{1}_{x<0}) dx$ Ainsi l'hypothèse de base : Il existe $\alpha > 0$ tel que $\int_{x>1} (e^{(1+\alpha)x}) f(dx) < \infty$ est vrai ssi $\lambda_+ > 1$.

Par Lévy-Khintchine on a :

$$\phi(u) = \mathbb{E}(e^{iuM_1}) = e^{T\left(-\frac{\sigma^2}{2} + ibu + iu\lambda\left\{\frac{p}{\lambda_+ - iu} - \frac{1-p}{\lambda_- + iu}\right\}\right)}$$

5.3 Modèle de Merton :

Pour le modèle de Merton on a : $f(dx) = \frac{\exp(-(x-\mu)^2)}{\sqrt{2\pi\delta^2}} dx$ donc il existe $\alpha > 0$ tel que $\int_{x>1} (e^{(1+\alpha)x}) G(dx) < \infty$ est vrai ssi $\lambda_+ > 1$. Et par Lévy-Khintchine on a :

$$\phi(u) = \mathbb{E}(e^{iuM_1}) = e^{T\left(-\frac{\sigma^2}{2} + ibu + \lambda\left\{e^{-\delta^2 u^2/2 + i\mu u} - 1\right\}\right)}$$

5.4 Résultats

Pour le calcul de FFT on a essayé de calculer le résultat sur plusieurs strikes grâce à la formule $\tilde{k}_n = \tilde{k}_0 + \frac{2\pi n}{N\Delta}$. Ainsi on peut visualiser la variation du prix du call avec l'augmentation du strike dans les figures suivantes avec comme paramètres : $X_0 = 100$, $N = 2^{11}$, $\delta = 1$, $\Sigma = 0.2$, $c = 0.2$, $\lambda_+ = 3$, $\lambda_- = 6$, $\lambda = 4$

5.4.1 Modèle de Kou :

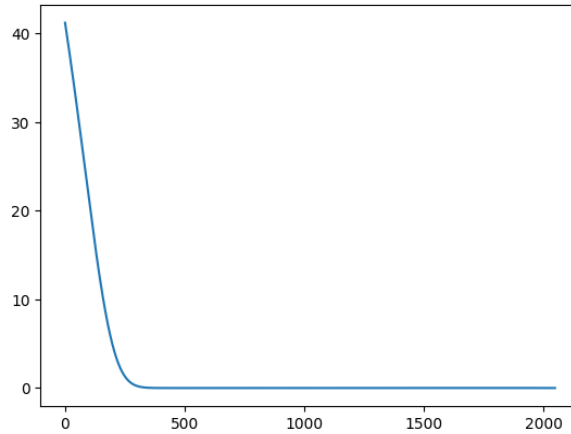


FIGURE 6 – variation du call pour le modèle de Kou suivant la valeur de n

On observe donc une fonction décroissante ce qui est cohérent financièrement et régulière en strike. De plus cette méthode est plus rapide que celle en utilisant un schéma de discrétisation grâce à la FFT qui nous permet de réduire le calcul.

5.4.2 Modèle de Merton :

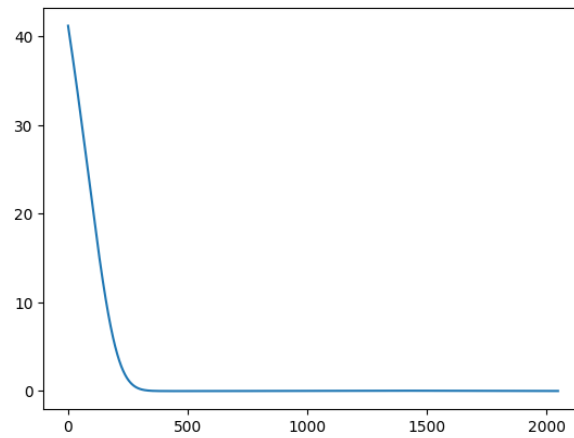


FIGURE 7 – variation du call pour le modèle de Merton suivant la valeur de n

D'une manière similaire, on peut visualiser les prix des calls dans le modèle de Merton qui sont semblables à ceux du modèle de Kou et qui présentent une meilleure complexité en terme de calcul par rapport aux méthodes vues précédemment.

Références

- [Duq] Thomas Duquesne. Introduction au processus de sauts. *UPMC, Polycopié du “Master 2 Probabilités et Finance”*.
- [NEK] Emmanuel Gobet Nicole El Karoui. Les outils stochastiques des marchés financiers.
- [Pag16] Gilles Pagès. Introduction to numerical probability for finance,. *UPMC, Polycopié du “Master 2 Probabilités et Finance”*, 2016.
- [POP] Alexandre POPIER. Processus de lévy, et applications en finance. *ENSTA, Polycopié parcours finance quantitative*.
- [Rub03] Sylvain Rubenthaler. Numerical simulation of the solution of a stochastic differential equation driven by a lévy process. *Stochastic Processes and their Applications*, 103(2) :311–349, 2003.
- [TV09] Peter Tankov and Ekaterina Voltchkova. Jump-diffusion models : A practitioner’s guide. *Banque et Marchés*, 99 :1–24, 2009.