Rapport de Stage

Résolution numérique d'équations différentielles stochastiques rétrogrades à l'aide de la quantification optimale

Adrien Richou

Sous la direction d'Étienne Tanré

20 septembre 2006

Supaero
Option IMS-IF

Université Paul Sabatier
Laboratoire de Statistique
et Probabilités

Inria Sophia-Antipolis
Projet Omega

Table des matières

Résumé

Ce rapport regroupe l'ensemble du travail effectué lors d'un projet de fin d'études de cinq mois réalisé au sein de l'équipe *OMEGA* à l'INRIA de Sophia-Antipolis. Nous y traitons de méthodes pour résoudre numériquement des équations différentielles stochastiques rétrogrades (EDSR) réfléchies ou non réfléchies. La première partie introduit ces concepts, établie des liens avec les équations aux dérivées partielles et présente des applications dans le domaine des mathématiques financières. La deuxième partie traite des méthodes de quantification optimale, methodes qui seront utiles pour réaliser une discrétisation spatiale des EDSR dont on cherche à approcher numériquement leur solution. Un algorithme de recherche de quantification optimale pour des vecteurs aléatoires dont l'ensemble image est de cardinal fini, appelé *k-mean*, est introduit dans une troisième partie. Nous y traitons des problèmes théoriques de convergence et des questions pratiques d'implantation de cet algorithme. Tous ces outils présentés nous ont permis de mettre au point un schéma de discrétisation dont la convergence théorique est étudiée dans la partie quatre. Enfin, il est question dans la dernière partie de quelques applications numériques de ce schéma. Nous y abordons des problémes liés à la fixation de ses paramètres ainsi qu'un certain nombre de développements réalisés ou seulement envisagés.

Mots clés : équation différentielle stochastique rétrograde, équation différentielle stochastique rétrograde réfléchie, équation aux dérivées partielles, mathématiques financières, évaluation d'options, quantification optimale, algorithme du k-mean.

Remerciements

Je tiens à remercier chaleureusement l'ensemble du projet Omega qui m'a très bien accueilli durant mon séjour à Sophia-Antipolis. L'ambiance qui y règne ainsi que les compétences de ses membres furent pour moi un très bon stimulant. Je remercie plus particulièrement M. Denis Talay qui a accepté de me recevoir au sein de son équipe, ainsi que M. Etienne Tanré qui m'a encadré. Je lui suis reconnaissant pour sa disponibilité et l'autonomie dont il m'a laissé disposer. Enfin, je remercie M. Manuel Samuelides pour m'avoir aidé durant la recherche de mon stage à Supaero.

Chapitre 1

Les équations différentielles stochastiques rétrogrades (EDSR)

1.1 Présentation

1.1.1 Contexte général

Dans toute la suite nous considérerons un espace de probabilité $(\Omega, \mathcal{F}, \mathbb{P})$ et $(B_t)_{t\in[0,T]}$ un mouvement brownien de dimension d défini sur cet espace de probabilité avec T un réel strictement positif. On note $(\mathcal{F}_t; 0 \leq t \leq T)$ sa filtration naturelle augmentée, c'est-à-dire la $\sigma-$algèbre générée par le mouvement brownien B et les ensembles de probabilité nulle. De plus, on note $M^2([0,T], \mathbb{R}^d)$ l'espace des processus progressivement mesurables de carré intégrable sur $[0,T]\times\Omega$. Enfin, $|.|$ représentera indifféremment la valeur absolue sur $\mathbb{R}$, la norme canonique de $\mathbb{R}^d$ et la norme d'opérateur de $\mathbb{R}^{d\times d}$.

1.1.2 Solutions d'EDSR

Les équations différentielles stochastiques rétrogrades (EDSR) ont été introduites par Bismut en 1973 dans le cas linéaire et par Pardoux et Peng en 1990 dans le cas général ([19]). Elles tirent notamment leur origine des travaux portant sur le contrôle stochastique optimal. La solution d'une équation différentielle stochastique rétrograde est un triplé (X, Y, Z) $\mathcal{F}_t$ adapté, de carré intégrable, qui vérifie

$$(E) \quad \begin{cases} X_t = x_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s \\ Y_t = g(X_T) + \int_t^T f(s, X_s, Y_s, Z_s)ds - \int_t^T Z_s dB_s \end{cases}$$

avec $x_0 \in \mathbb{R}^d$, $b : \mathbb{R}^d \to \mathbb{R}^d$, $f : [0,T]\times\mathbb{R}^d\times\mathbb{R} \to \mathbb{R}$, $\sigma : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ et $g : \mathbb{R}^d \to \mathbb{R}$.

Remarques :

- X est un simple processus de diffusion partant de x_0 en $t = 0$.
- Y ne possède pas de condition initiale mais une condition finale. Il n'est donc pas évident a priori d'avoir un processus adapté. En effet, si l'on supprime Z de (E), on obtient

$$Y_t = g(X_T) + \int_t^T f(s, X_s, Y_s)ds.$$

  Dans ce cas là, Y n'a aucune chance de pouvoir être un processus adapté car il dépend explicitement du futur. Ainsi le processus Z peut être vu comme un contrôle permettant au processus Y d'être adapté.

1.1.3 Relations avec les équations aux dérivées partielles (EDP)

On considère l'équation aux dérivées partielles quasi-linéaire suivante

$$(\mathcal{E}) \quad \begin{cases} \partial_t u(t,x) + \langle b(x), \nabla_x u(t,x)\rangle + \frac{1}{2} tr((\sigma\sigma^*)(x)\nabla^2_{x,x} u(t,x)) \\ +f(t,x,u(t,x),(\nabla u\sigma)(t,x)) = 0 \\ u(T,x) = g(x) \end{cases}$$

Cette équation est fortement reliée à (E). En effet, si $u \in C^{1,2}([0,T] \times \mathbb{R}^d, \mathbb{R})$ est une solution de $(\mathcal{E})$ et X un processus vérifiant

$$X_t = x_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s$$

alors $(X_t, u(t,X_t), (\nabla u\sigma)(t,X_t))_{t\in[0,T]}$ est une solution de (E). Pour s'en convaincre, il suffit d'appliquer la formule d'Itô à $u_t = u(t,X_t)$:

$$du_t = \mathcal{L}_t u(t,X_t)dt + (\nabla u\sigma)(t,X_t)dB_t$$

avec

$$\mathcal{L}_t = \frac{\partial}{\partial t} + \frac{1}{2}\sum_{i,j=1}^d (\sigma\sigma^*)_{i,j}(x)\frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^d b_i(x)\frac{\partial}{\partial x_i}.$$

Or u vérifie $(\mathcal{E})$, donc

$$du(t,X_t) = -f(t,x,u(t,x),(\nabla u\sigma)(t,X_t))dt + (\nabla u\sigma)(t,X_t)dB_t$$

ce qui est le résultat escompté. Bien entendu, il convient maintenant d'étudier de plus près l'existence et l'unicité de solutions pour (E) et $(\mathcal{E})$: ce sera l'objet de la partie 1.2.

1.2 Résultats d'existence et d'unicité

1.2.1 Résultats généraux sur les EDSR

Rappelons tout d'abord le résultat classique suivant

Théorème 1.2.1 *On suppose que b et σ sont uniformément lipschitziens. Pour tout $x \in \mathbb{R}$ il existe une unique solution dans $M^2([0,T],\mathbb{R}^d)$ de l'équation différentielle stochastique*

$$X_t^{0,x} = x + \int_0^t b(X_u^{0,x})du + \int_0^t \sigma(X_u^{0,x})dB_u, \quad 0 \leq t \leq T$$

Démonstration :

- Unicité : supposons que $(X_{1,t}^{0,x})$ et $(X_{2,t}^{0,x})$ sont deux processus solutions. Alors

$$\begin{aligned}
\mathbb{E}\left|X_{1,t}^{0,x} - X_{2,t}^{0,x}\right|^2 &\leq C_d\mathbb{E}\left|\int_0^t b(X_{1,u}^{0,x}) - b(X_{2,u}^{0,x})du\right|^2 \\
&\quad + C_d\mathbb{E}\left|\int_0^t \sigma(X_{1,u}^{0,x}) - \sigma(X_{2,u}^{0,x})dB_u\right|^2 \\
\mathbb{E}\left|X_{1,t}^{0,x} - X_{2,t}^{0,x}\right|^2 &\leq tC_d\mathbb{E}\left[\int_0^t \left|b(X_{1,u}^{0,x}) - b(X_{2,u}^{0,x})\right|^2 du\right] \\
&\quad + C_d\mathbb{E}\left[\int_0^t \left|\sigma(X_{1,u}^{0,x}) - \sigma(X_{2,u}^{0,x})\right|^2 du\right]
\end{aligned}$$

  avec C_d une constante ne dépendant que de d. En utilisant le fait que b et σ sont uniformément lipschitziennes[1], nous obtenons

$$\mathbb{E}\left|X_{1,t}^{0,x} - X_{2,t}^{0,x}\right|^2 \leq C_dK^2(1+t)\mathbb{E}\left[\int_0^t \left|X_{1,u}^{0,x} - X_{2,u}^{0,x}\right|^2 du\right].$$

  Il suffit alors d'appliquer le lemme de Gronwall pour en déduire que

$$\mathbb{E}\left|X_{1,t}^{0,x} - X_{2,t}^{0,x}\right|^2 = 0$$

- Existence : comme $M^2([0,T],\mathbb{R}^d)$ est un espace de Hilbert et que nous ne connaissons pas le processus limite, nous allons construire une suite de Cauchy qui va converger vers le processus souhaité. Ainsi, on considère la suite de processus suivante définie par récurrence

$$X_t^{n+1} = x + \int_0^t b(X_u^n)du + \int_0^t \sigma(X_u^n)dB_u$$

[1] b et σ sont K lipschitziens.

avec $X_t^0 = x$. Remarquons tout de suite que, si cette suite converge, le processus limite est solution de l'équation différentielle stochastique. En reprenant les mêmes calculs que pour l'unicité, on montre que

$$\mathbb{E}\left|X_t^{n+1} - X_t^n\right|^2 \leq \underbrace{C_d K^2 (1+T)}_{A} \mathbb{E}\left[\int_0^t \left|X_u^{n+1} - X_u^n\right|^2 du\right].$$

En posant

$$u_n = \int_0^T \mathbb{E}\left|(X_t^{n+1} - X_t^n\right|^2 dt,$$

on obtient

$$u_{n+1} \leq u_0 \frac{A^n T^{n+1}}{(n+1)!}.$$

□

Il est démontré dans [19] le résultat suivant

Théorème 1.2.2 *On suppose que*

- *b et σ sont uniformément lipschitziennes,*
- *g est une fonction mesurable à croissance au plus polynômiale,*
- *f est une fonction mesurable, bornée vis à vis des deux premières variables et uniformément lipschitzienne vis à vis des deux dernières variables.*

Alors (E) possède une unique solution (X, Y, Z) dans $M^2([0,T]), \mathbb{R}^d) \times M^2([0,T], \mathbb{R}) \times M^2([0,T], \mathbb{R}^d)$.

Démonstration : La démonstration complète se trouve dans [19]. Nous allons juste développer quelque peu la méthode de construction de la solution en détaillant les étapes importantes. Notons dans un premier temps qu'une partie du résultat, à savoir l'existence et l'unicité de X, est l'objet du théorème 1.2.1. En ce qui concerne les processus Y et Z, nous allons nous appuyer sur des lemmes successifs.

Lemme 1.2.3 *Soient $\bar{Y} \in M^2([0,T], \mathbb{R})$ et $\bar{Z} \in M^2([0,T], \mathbb{R})$, alors l'équation*

$$Y_t = g(X_T) + \int_t^T f(s, X_s, \bar{Y}_s, \bar{Z}_s) ds - \int_t^T Z_s dB_s$$

possède une unique solution (Y, Z) dans $M^2([0,T], \mathbb{R}) \times M^2([0,T], \mathbb{R}^d)$.[2]

Démonstration : On pose

$$Y_t = \mathbb{E}\left[g(X_T) + \int_t^T f(s, X_s, \bar{Y}_s, \bar{Z}_s) ds | \mathcal{F}_t\right]$$

[2] Nous supposons toujours les hypothèses du théorème 1.2.2 vérifiées.

D'après le théorème de représentation martingale (cf [11] partie 3.4) il existe un unique processus $Z \in M^2([0,T],\mathbb{R}^d)$ tel que[3]

$$\mathbb{E}\left[g(X_T) + \int_0^T f(s, X_s, \bar{Y}_s, \bar{Z}_s)ds|\mathcal{F}_t\right] = Y_0 + \int_0^t Z_s dBs.$$

Or

$$\begin{aligned} g(X_T) &+ \int_t^T f(s, X_s, \bar{Y}_s, \bar{Z}_s)ds - \int_t^T Z_s dB_s \\ &= g(X_T) + \int_t^T f(s, X_s, \bar{Y}_s, \bar{Z}_s)ds - \int_0^T Z_s dB_s + \int_0^t Z_s dB_s \\ &= -\int_0^t f(s, X_s, \bar{Y}_s, \bar{Z}_s)ds + \mathbb{E}\left[g(X_T) + \int_0^T f(s, X_s, \bar{Y}_s, \bar{Z}_s)ds|\mathcal{F}_t\right] \\ &= Y_t. \end{aligned}$$

□

Lemme 1.2.4 *Soit $\bar{Y} \in M^2([0,T],\mathbb{R})$, alors l'équation*

$$Y_t = g(X_T) + \int_t^T f(s, X_s, \bar{Y}_s, Z_s)ds - \int_t^T Z_s dB_s$$

possède une unique solution (Y,Z) dans $M^2([0,T],\mathbb{R}) \times M^2([0,T],\mathbb{R}^d)$.[4]

Démonstration : Pour démontrer l'existence, les auteurs de [19] utilise la même idée que pour le théorème 1.2.1, à savoir construire une suite de Cauchy de $M^2([0,T],\mathbb{R}) \times M^2([0,T],\mathbb{R}^d)$. En utilisant le lemme précédent, nous pouvons considérer la suite $(Y^n, Z^n)_{n\geq 0}$ suivante

$$\begin{cases} Y_t^0 = 0, \quad Z_t^0 = 0, \\ Y_t^n = g(X_T) + \int_t^T f(s, X_s, \bar{Y}_s, Z_s^{n-1})ds - \int_t^T Z_s^n dB_s. \end{cases}$$

La convergence de cette suite et l'unicité de la solution sont démontrées dans [19]. □

Revenons à la démonstration du théorème. Pour s'assurer de l'existence nous allons, une nouvelle fois, construire une suite de Cauchy de $M^2([0,T],\mathbb{R}) \times M^2([0,T],\mathbb{R}^d)$ en utilisant le lemme précédent :

$$\begin{cases} Y_t^0 = 0, \quad Z_t^0 = 0, \\ Y_t^n = g(X_T) + \int_t^T f(s, X_s, Y_s^{n-1}, Z_s^n)ds - \int_t^T Z_s^n dB_s. \end{cases}$$

La convergence de cette suite et l'unicité de la solution sont démontrées dans [19]. □

[3] Pour pouvoir appliquer le théorème de représentation martingale il convient d'avoir une martingale progressivement mesurable de carré intégrable. Nous avons donc besoin des hypothèses sur g et du résultat classique $\mathbb{E}\,|X|_T^{2p} < +\infty$.

[4] Nous supposons toujours les hypothèses du théorème 1.2.2 vérifiées.

1.2.2 Résultats sur les EDP associées aux EDSR

Pour s'assurer de l'existence et de l'unicité de solutions pour les EDP associées aux EDSR il convient d'avoir des hypothèses plus fortes sur f, g, b et σ.

Hypothèses $(\mathcal{H}_1)$ **:**

1. b, f, g et σ sont bornées en espace et ont une croissance au plus linéaire vis à vis des autres variables.
2. b, f, g et σ sont uniformément lipschitziennes vis à vis de toutes les variables.
3. $\sigma\sigma^*$ est uniformément elliptique. ie $\exists \varepsilon > 0$, $\forall x \in \mathbb{R}^d$, $\sigma(x)\sigma^*(x) - \varepsilon Id$ est positive.
4. g est bornée dans $C^2(\mathbb{R}^d)$.

Comme nous souhaitions assouplir l'hypothèse de bornitude en espace de b, f, g et σ, nous avons considéré un second jeu d'hypothèses.

Hypothèses $(\mathcal{H}_2)$ **:**

1. $b \in C^3(\mathbb{R}^d, \mathbb{R}^d)$, $\sigma \in C^3(\mathbb{R}^d, \mathbb{R}^{d\times d})$, $g \in C^3(\mathbb{R}^d)$.
2. Les dérivées partielles d'ordre inférieur ou égal à 3 de b et σ sont bornées.
3. Les dérivées d'ordre inférieur ou égal à 3 de g sont à croissance au plus polynômiale.
4. $(x, y, z) \to f(s, x, y, z)$ est de classe C^3 quelque soit $s \in [0, T]$ et de plus
   - Pour tout $s \in [0, T]$, les dérivées partielles d'ordre inférieur ou égal à 3 de $f(s, ., 0, 0)$ sont à croissance au plus polynômiale.
   - $\partial f/\partial y$, $\partial f/\partial z$ ainsi que leurs dérivées partielles d'ordre 1 et 2 en x, y et z sont bornées sur $[0, T] \times \mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d$.

Notons que les hypothèses $(\mathcal{H}_2)$ sont plus contraignantes que $(\mathcal{H}_1)$ sur la régularité des coefficients.

D'après [13][5] et [14] on a le résultat suivant :

Théorème 1.2.5
- *Sous les hypothèses* $(\mathcal{H}_1)$, $(\mathcal{E})$ *admet une solution* $u \in C^{1,2}([0, T] \times \mathbb{R}^d, \mathbb{R})$. *De plus, il existe une constante* C *telle que pour tout* $(t, x) \in [0, T] \times \mathbb{R}^d$ *on a*

$$|u(t, x)| + |\nabla_x u(t, x)| + \left|\nabla^2_{x,x} u(t, x)\right| + |\partial_t u(t, x)| \leq C$$

  Enfin, u est unique dans la classe des fonctions $\tilde{u} \in C([0, T] \times \mathbb{R}^d, \mathbb{R}) \cap C^{1,2}([0, T[\times\mathbb{R}^d, \mathbb{R})$ *vérifiant* $\sup_{(t,x)\in[0,T[\times\mathbb{R}^d}(|\tilde{u}(t, x)| + |\nabla_x \tilde{u}(t, x)|) < +\infty$
- *Sous ces mêmes hypothèses, il existe un unique triplé* (X, Y, Z) *progressivement mesurable, de carré intégrable, à valeur dans* $\mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d$ *tel que* $\mathbb{E}\sup_{t\in[0,T]}(|X_t|^2 + |Y_t|^2) < +\infty$ *et satisfaisant* $\mathbb{P}p.s.$ (E).

[5] chapitre VI, théorème 4.1

D'après [18] on a le résultat suivant :

Théorème 1.2.6 • *Sous les hypothèses* $(\mathcal{H}_2)$, $(\mathcal{E})$ *admet une unique solution* $u \in C^{1,2}([0,T] \times \mathbb{R}^d, \mathbb{R})$. *De plus, pour tout* $p \geq 2$, *il existe* c_p *et un entier* q *tels que* $\forall p \geq 2$, $\forall (x,x') \in (\mathbb{R}^d)^2$, $\forall (t,t') \in [0,T]^2$,

$$\begin{aligned} |u(t,x) - u(t',x')|^p &\leq c_p(1+|x|^q)(|x-x'|^p + |t-t'|^{p/2}) \\ |\nabla_x u(t,x) - \nabla_{x'} u(t',x')|^p &\leq c_p(1+|x|^q+|x'|^q)(|x-x'|^p + |t-t'|^{p/2}) \end{aligned}$$

- *Sous ces mêmes hypothèses, il existe un unique triplé* (X,Y,Z) *progressivement mesurable, de carré intégrable, à valeur dans* $\mathbb{R}^d \times \mathbb{R} \times \mathbb{R}^d$ *satisfaisant* $\mathbb{P}p.s.$ (E).

Le théorème suivant permet de faire le lien entre Les EDSR et les EDP associées.

Théorème 1.2.7 *Plaçons nous sous les hypothèses* $(\mathcal{H}_1)$ *ou* $(\mathcal{H}_2)$. *On note* u *l'unique solution de* $(\mathcal{E})$ *et* (X,Y,Z) *l'unique solution adaptée de* (E). *Alors,*

$$u(t,x) = \mathbb{E}[Y_t | X_t = x] \quad (\nabla u \sigma)(t,x) = \mathbb{E}[Z_t | X_t = x]$$

Ce résultat découle directement des théorèmes d'existence et d'unicité précédents et de la remarque de la partie 1.1.3.

Remarques :

- Dans la majeure partie des travaux sur les EDSR, Y est un processus vectoriel. Ainsi, on ne considère plus l'EDP associée à l'EDSR, mais le système d'EDP associé.
- Il est possible d'assouplir certaines hypothèses de dérivabilité pour f, g, b et σ. Dans ce cas, $u(t,x) := \mathbb{E}[Y_t | X_t = x]$ n'est pas une solution de $(\mathcal{E})$, car pas assez régulière, mais seulement une solution de viscosité de $(\mathcal{E})$. De tels résultats sont disponibles dans [18], [20] ou [21] par exemple.

1.2.3 Où l'on retrouve la formule de Feynman-Kac

Supposons que f puisse s'écrire sous la forme

$$f(t,x,y,z) = c(t,x)y + h(t,x).$$

Dans ce cas, le processus Y de l'EDSR associée possède une solution explicite que l'on peut déterminer grâce à la méthode de variation de la constante :

$$\begin{aligned} Y_s^{t,x} &= g(X_T^{t,x}) e^{\int_s^T c(r,X_r^{t,x})dr} + \int_s^T h(r, X_r^{t,x}) e^{\int_s^r c(u,X_u^{t,x})du} dr \\ &\quad - \int_s^T e^{\int_s^r c(u,X_u^{t,x})du} Z_r^{t,x} dB_r \end{aligned}$$

Nous avons vu que si u est une solution de $(\mathcal{E})$, et s'il existe une unique solution (X, Y, Z) dans $M^2([0,T]), \mathbb{R}^d) \times M^2([0,T], \mathbb{R}) \times M^2([0,T], \mathbb{R}^d)$, alors

$$u(t,x) = Y_t^{t,x} = \mathbb{E}\left[g(X_T^{t,x}) e^{\int_t^T c(r, X_r^{t,x}) dr} + \int_t^T h(r, X_r^{t,x}) e^{\int_t^r c(u, X_u^{t,x}) du} dr \right], \tag{1.1}$$

ce qui est la formule de Feynman-Kac.

1.3 Les EDSR réfléchies

1.3.1 Présentation

Dans certaines applications des EDSR il peut apparaître une contrainte, dite de « frontière libre », de la forme $Y_t \geq h(t, X_t)$. Il convient alors de modifier la notion d'EDSR pour intégrer cette contrainte. La solution d'une équation différentielle stochastique rétrograde réfléchie est un quadruplé (X, Y, Z, K) $\mathcal{F}_t$ adapté, de carré intégrable, qui vérifie

$$(ER) \quad \begin{cases} X_t = x_0 + \int_0^t b(X_s) ds + \int_0^t \sigma(X_s) dB_s \\ Y_t = g(X_T) + \int_t^T f(s, X_s, Y_s, Z_s) ds + K_T - K_t - \int_t^T Z_s dB_s, \quad Y_t \geq h(t, X_t) \\ \int_0^T (Y_t - h(t, X_t)) dK_t = 0 \end{cases}$$

avec K un processus continu et croissant, $K_0 = 0$, $x_0 \in \mathbb{R}^d$, $b : \mathbb{R}^d \to \mathbb{R}^d$, $f : [0,T] \times \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$, $\sigma : \mathbb{R}^d \to \mathbb{R}^{d \times d}$, $g : \mathbb{R}^d \to \mathbb{R}$ et $h : [0,T] \times \mathbb{R}^d \to \mathbb{R}$.

L'introduction du processus croissant K permet de « contenir » Y au dessus de h en venant s'ajouter à celui-ci. L'équation $\int_0^T (Y_t - h(t, X_t)) dK_t = 0$ autorise K à « contenir » Y de façon optimale, c'est-à-dire seulement lorsque $Y_t = h(t, X_t)$. Il est montré dans [6] le résultat suivant

Théorème 1.3.1 *Supposons que*

- *b et σ sont uniformément lipschitziennes,*
- *g est continue et à croissance au plus polynômiale,*
- *f et h sont continues,*
- *il existe $K \in \mathbb{R}^+$ et $p \in \mathbb{N}$ tels que pour tous $t \in [0,T]$, $x, z, z' \in \mathbb{R}^d$, $y, y' \in \mathbb{R}$,*
  - *$|f(t,x,0,0)| \leq K(1 + |x|^p)$,*
  - *$|f(t,x,y,z) - f(t,x,y',z')| \leq K(|y - y'| + |z - z'|)$,*
  - *$h(t,x) \leq K(1 + |x|^p)$,*
- *$h(T,x) \leq g(x)$.*

Alors, (ER) possède une unique solution (X, Y, Z, K) dans $M^2([0,T]), \mathbb{R}^d) \times M^2([0,T], \mathbb{R}) \times M^2([0,T], \mathbb{R}^d) \times M^2([0,T], \mathbb{R})$.

1.3.2 EDP avec obstacle associée

Il est montré dans [6] que Y peut être vu comme l'interprétation probabiliste de la solution de l'EDP avec obstacle suivante :

$$(\mathcal{ER}) \quad \begin{cases} \max\left(\mathcal{L}_t u(t,x) + f(t,x,u(t,x),(\nabla u\sigma)(t,x)), h(t,x) - u(t,x)\right) = 0 \\ u(T,x) = g(x) \end{cases}$$

Comme précédemment, si u est une solution de $(\mathcal{ER})$, alors $u(t,X_t) = Y_t$. Les auteurs de [6] traitent du cas plus général où l'on a une solution de viscosité.

1.3.3 Problème de temps d'arrêt optimal associé

Il est également possible de considérer Y comme la solution d'un problème de temps d'arrêt optimal.

Proposition 1.3.2 *Si (X,Y,Z,K) est une solution de (ER), alors*

$$Y_t = ess\sup_{\tau\in\mathcal{T}_t} \mathbb{E}\left[\int_t^\tau f(s,X_s,Y_s,Z_s)ds + h(\tau,X_\tau)\mathbb{1}_{\tau<T} + g(X_T)\mathbb{1}_{\tau=T}|\mathcal{F}_t\right],$$

avec $\mathcal{T}_t$ l'ensemble des temps d'arrêt à valeur dans $(t,T]$.

Démonstration : Soit $\tau \in \mathcal{T}_t$. On a alors

$$\begin{aligned} Y_t &= \mathbb{E}\left[\int_t^\tau f(s,X_s,Y_s,Z_s)ds + Y_\tau + K_\tau - K_t|\mathcal{F}_t\right] \\ &\geq \mathbb{E}\left[\int_t^\tau f(s,X_s,Y_s,Z_s)ds + h(\tau,X_\tau)\mathbb{1}_{\tau<T} + g(X_T)\mathbb{1}_{\tau=T}|\mathcal{F}_t\right] \end{aligned}$$

car K est croissante et $Y \geq h(.,X)$. Pour obtenir l'inégalité réciproque, posons maintenant le temps d'arrêt suivant

$$D_t = \inf\left\{u \in [t,T]; Y_u = h(u,X_u)\mathbb{1}_{u<T} + g(X_T)\mathbb{1}_{u=T}\right\}.$$

Comme $\int_0^T (Y_u - h(u,X_u))dK_u = 0$ et que K est continue, alors $K_{D_T} - K_t = 0$. Ainsi

$$Y_t = \mathbb{E}\left[\int_t^{D_t} f(s,X_s,Y_s,Z_s)ds + h(D_t,X_{D_t})\mathbb{1}_{D_t<T} + g(X_T)\mathbb{1}_{D_t=T}|\mathcal{F}_t\right].$$

□

1.4 Quelques applications à la finance

Rappelons quelques notions sur les produits dérivés en finance. Une option financière est un produit dérivé qui donne le droit, et non l'obligation, d'acheter (option d'achat, appelée aussi « call ») ou de vendre (option de vente, appelée

aussi « put ») une quantité donnée d'actifs financiers, appelés actifs sous-jacents, à un prix précisé à l'avance (prix d'exercice) et à une date d'échéance donnée (option européenne) ou avant une date donnée (option américaine). Ce droit lui même s'achète ou se vend sur un marché d'options. De façon plus générale, on peut considérer des contrats qui permettent le versement, à la date d'exercice, d'une prime fonction des actifs financiers. Cette prime est appelée Payoff. Le but pour la banque est alors de pouvoir fixer un prix pour ces contrats en utilisant une modélisation mathématique du marché. Le fait que le prix du contrat soit connu de manière certaine à la date d'exercice laisse envisager que la théorie des EDSR soit applicable dans ce domaine : X serait le sous-jacent et Y le prix de l'option. En pratique, les auteurs de [8] ont mis en lumière de nombreuses applications des EDSR dans le domaine financier et notamment l'évaluation d'options.

1.4.1 L'évaluation d'options européennes

L'exemple le plus simple concerne les options européennes sur actifs boursiers. On note X le sous-jacent : il représente d actifs boursiers. Nous le modélisons comme un processus de diffusion classique :

$$X_t = x_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s.$$

On note Y le prix de l'option. Enfin, g représente le Payoff qui est fonction des actifs boursiers. L'évaluation de l'option peut se faire à l'aide de la méthode de réplication : on crée un portefeuille, constitué d'un actif sans risque A rémunéré au taux r et des actifs risqués X, et qui sera égal à la valeur de l'option à la date d'exercice. On note θ^0 la quantité d'actif sans risque et θ^1 les quantités d'actifs risqués. Alors

$$Y_t = \theta_t^0 A_t + \theta_t^{\mathbb{1}}.X_t$$

Nous supposerons que le portefeuille est autofinancé, donc

$$dY_t = r\theta_t^0 A_t dt + \theta_t^{\mathbb{1}}.dX_t = r(Y_t - \theta_t^{\mathbb{1}}.X_t)dt + \theta_t^{\mathbb{1}}.dX_t.$$

Ainsi, nous obtenons l'EDSR suivante :

$$Y_t = g(X_T) - \int_t^T r(Y_s - \theta_s^{\mathbb{1}}.X_s) + \theta_s^{\mathbb{1}}.b(X_s)ds - \int_t^T \theta_s^{\mathbb{1}}.\sigma(X_s)dB_s.$$

Remarques :

- Le contrôle est ici assuré par θ^1, les quantités d'actifs risqués.
- Notons que cet exemple est très simple, nous n'avons pas forcément besoin de la théorie des EDSR pour résoudre ce problème. En effet, en considérant le prix actualisé $e^{-rt}Y_t$ au lieu de Y_t, nous pouvons faire disparaître le terme $\int_t^T rY_s ds$. De plus, par l'intermédiaire du théorème de Girsanov, il est possible de faire un changement de probabilité afin de transformer le prix actualisé en une martingale[6].

[6] Cette transformation nécessite tout de même certaines hypothèses sur b et σ.

1.4.2 L'évaluation d'options américaines

Nous nous plaçons sous les mêmes hypothèses que précédemment mais cette fois l'acheteur peut exercer son droit à tout moment jusqu'à la date d'échéance. Supposons que l'acheteur divulgue sa stratégie à l'avance, c'est-à-dire le moment d'exercice de son option. Ce moment est un temps d'arrêt τ. Dans ces conditions, il suffit de reprendre les calculs précédents pour évaluer Y :

$$Y_t^\tau = g(X_\tau) - \int_t^\tau r(Y_s^\tau - \theta_s^{\mathbb{1}}.X_s) + \theta_s^{\mathbb{1}}.b(X_s)ds - \int_t^\tau \theta_s^{\mathbb{1}}.\sigma(X_s)dB_s.$$

Intuitivement, nous aurons donc

$$Y_t = ess \sup_{\tau \in \mathcal{T}_t} Y_t^\tau.$$

Y devrait alors vérifier l'EDSR réfléchie suivante

$$\begin{cases} X_t = x_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s \\ Y_t = g(X_T) - \int_t^T r(Y_s^\tau - \theta_s^{\mathbb{1}}.X_s) + \theta_s^{\mathbb{1}}.b(X_s)ds + K_T - K_t - \int_t^T \theta_s^{\mathbb{1}}.\sigma(X_s)dB_s, \quad Y_t \geq g(X_t) \\ \int_0^T (Y_t - g(X_t))dK_t = 0 \end{cases}$$

Ce résultat est démontré proprement dans [7].

Remarque : Il apparaît dans l'EDSR réfléchie la contrainte $Y_t \geq g(X_t)$. Celle ci se justifie par l'hypothèse d'absence d'opportunité d'arbitrage. En effet, supposons que $Y_t < g(X_t)$: Il suffit alors d'exercer son droit au moment de l'achat du contrat pour obtenir sûrement un gain strictement positif.

1.4.3 Exemples de portefeuilles sous contraintes

Supposons maintenant que la banque ne puisse pas prêter et emprunter au même taux[7]. Notons r le taux d'intérêt du placement sans risque et R le taux d'intérêt d'emprunt avec $r < R$. Alors on a

$$dY_t = r(Y_t - \theta_t^{\mathbb{1}}.X_t)^+ dt + R(Y_t - \theta_t^{\mathbb{1}}.X_t)^- dt + \theta_t^{\mathbb{1}}.dX_t$$

$$Y_t = g(X_T) - \int_t^T r(Y_s - \theta_s^{\mathbb{1}}.X_s)^+ + R(Y_s - \theta_s^{\mathbb{1}}.X_s)^- + \theta_s^{\mathbb{1}}.b(X_s)ds + \theta_s^{\mathbb{1}}.\sigma(X_s)dB_s.$$

Contrairement aux exemples précédents, nous obtenons cette fois une fonction f non-linéaire.

Les auteurs de [8] traitent également des problèmes d'évaluation dans des marchés incomplets. Dans ce cas la banque ne peut pas se couvrir sur la totalité des actifs ce qui pose des problèmes d'existence de statégie de couverture. Nous renvoyons le lecteur à [8] pour plus de détails.

[7] Cet exemple est tiré de [8].

1.5 Résolution numérique des EDSR

Dans le cas général, (X, Y, Z) ne peut être déterminé théoriquement. Le but de notre travail va donc consister à calculer numériquement Y_0. Pour cela, il convient de faire des approximations. On a

$$X_{t_{k+1}}^{t_k,X_{t_k}} = X_{t_k} + \int_{t_k}^{t_{k+1}} b(X_s^{t_k,X_{t_k}})ds + \int_{t_k}^{t_{k+1}} \sigma(X_s^{t_k,X_{t_k}})dB_s$$

$$Y_{t_k}^{t_k,X_{t_k}} = \mathbb{E}\left[Y_{t_{k+1}}^{t_k,X_{t_k}} + \int_{t_k}^{t_{k+1}} f(s, X_s^{t_k,X_{t_k}}, Y_s^{t_k,X_{t_k}}, Z_s^{t_k,X_{t_k}})ds\right].$$

Posons $t_k = hk$ et simplifions les intégrales[8] :

$$\bar{X}_{t_{k+1}}^{t_k,\bar{X}_{t_k}} = \bar{X}_{t_k} + hb(\bar{X}_{t_k}) + \sigma(\bar{X}_{t_k})(B_{t_{k+1}} - B_{t_k})$$

$$\bar{Y}_{t_k}^{t_k,\bar{X}_{t_k}} = \mathbb{E}\left[\bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}} + hf(t_k, \bar{X}_{t_k}, \bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}}, \bar{Z}_{t_{k+1}}^{t_k,\bar{X}_{t_k}})\right].$$

Continuons à simplifier le problème en supposant que f ne dépend pas de Z[9]

$$\bar{Y}_{t_k}^{t_k,\bar{X}_{t_k}} = \mathbb{E}\left[\bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}} + hf(t_k, \bar{X}_{t_k}, \bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}})\right].$$

Pour les EDSR réfléchies nous simplifions le problème de temps d'arrêt optimal en un problème de temps d'arrêt optimal à valeur dans un espace discrétisé : on considère l'ensemble des temps d'arrêt à valeur dans $\{t_k, 0 \leq k \leq n\}$. On a alors l'approximation suivante

$$\bar{Y}_{t_k}^{t_k,\bar{X}_{t_k}} = \sup\left(h(t_k, \bar{X}_{t_k}), \mathbb{E}\left[\bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}} + hf(t_k, \bar{X}_{t_k}, \bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}})\right]\right).$$

Malheureusement, dans les deux cas il reste à calculer des espérances conditionnelles ce qui est extrêmement long numériquement. Nous allons donc passer par une discrétisation spatiale en utilisant des méthodes de quantification optimale afin de simplifier notablement la résolution numérique.

[8] Cela revient à prendre un schéma d'Euler pour X.
[9] Le problème général sera réintroduit dans une section du dernier chapitre.

Chapitre 2

La quantification optimale

Ce chapitre s'appuie sur les articles [1], [2], [17] et [16].

2.1 La quantification de vecteurs aléatoires

2.1.1 Présentation

Les méthodes de quantification optimale consistent à approcher de manière discrète des vecteurs aléatoires sous la contrainte d'un critère d'optimalité. Ainsi, l'idée de base est de remplacer un vecteur aléatoire $X \in L^2(\Omega, \mathbb{R}^d)$ par un vecteur aléatoire Y^* prenant au plus N valeurs en minimisant l'erreur induite :

$$Y^* = \arg\min \left\{ \|X - Y\|_2, Y : \Omega \to \mathbb{R}^d, \mathit{mesurable}, \mathrm{Card}(Y(\Omega)) \leq N \right\}$$

Notons $\Gamma = Y(\Omega) = \{x_1, \ldots, x_N\}$ le support de Y et $\mathrm{Proj}_\Gamma : \mathbb{R}^d \to \Gamma$ la projection sur le plus proche voisin. Nous pouvons remarquer que

$$\|X - \mathrm{Proj}_\Gamma(X)\|_2 \leq \|X - Y\|_2 .$$

Ainsi, le problème revient à déterminer uniquement le support Γ optimal car l'on peut facilement en déduire un vecteur associé :

$$\hat{X}^\Gamma := \mathrm{Proj}_\Gamma(X) = \sum_{i=1}^{N} x^i \mathbb{1}_{C_i(\Gamma)}(X), \quad X \in \mathbb{R}^d,$$

avec $C_1(\Gamma), \ldots, C_N(\Gamma)$ une partition borélienne de $\mathbb{R}^d$ telle que

$$C_i(\Gamma) \subset \left\{ \xi \in \mathbb{R}^d : \left|\xi - x^i\right| = \min_{x^j \in \Gamma} \left|\xi - x^j\right| \right\}.$$

Cette partition est appelée pavage de Voronoï.

Notre problème d'optimisation revient donc à minimiser la fonction

$$Q_N^2(x^1, ..., x^N) = \left\| X - \hat{X}^\Gamma \right\|_2^2 = \mathbb{E}\left[\min_{1 \leq i \leq N} |X - x^i|^2 \right].$$

Celle-ci est appelée distorsion ou erreur de quantification L^2. Comme c'est une fonction continue, on montre facilement qu'elle atteint un minimum.

Remarque : Par la suite, nous appellerons « quantification de X » $\hat{X}^\Gamma$ et « quantificateurs de X » $x^1, \ldots, x^N$.

2.1.2 Obtention de grilles de quantification

Il convient maintenant de se demander comment on obtient numériquement une grille de quantification, c'est-à-dire un Γ optimal. L'idée est de dériver formellement Q_N^2 pour ensuite construire un algorithme de gradient stochastique. Réécrivons tout d'abord la fonction distorsion

$$Q_N^2(x^1, ..., x^N) := \int q_N^2(x, \xi)\mathbb{P}_X(d\xi)$$

avec

$$q_N^2(x, \xi) := \min_{1 \le i \le N} |x^i - \xi|^2, \quad x = (x^1, ..., x^N) \in (\mathbb{R}^d)^N, \xi \in \mathbb{R}^d.$$

Alors on a

$$\begin{aligned} \nabla Q_N^2(x^1, ..., x^N) &= \mathbb{E}\left[\nabla_x q_N^2(x, X)\right] \\ \nabla_x q_N^2(x, \xi) &= 2\left((x^i - \xi)\mathbb{1}_{C_i(x)}(\xi)\right)_{1 \le i \le N} \end{aligned}$$

On définit donc l'algorithme de cette façon

$$\Gamma^{n+1} = \Gamma^n - \frac{\delta_{n+1}}{2}\nabla_x q_N^2(\Gamma^n, \xi^{n+1})$$

avec $(\xi^n)_{n>0}$ une suite de vecteurs aléatoires i.i.d. de même densité que X et $(\delta_n)_{n>0}$ une suite décroissante vérifiant

$$\sum_{n>0} \delta_n = +\infty \quad et \quad \sum_{n>0} \delta_n^2 < +\infty.$$

De plus, les auteurs de [1] montrent que l'erreur de quantification Q_N^2 et les poids $\pi_i = \mathbb{P}_X(C_i(\Gamma^*))$ peuvent être estimés en même temps. En pratique, si l'on note $\Gamma^n = \{x^{1,n}, ..., x^{N,n}\}$, l'algorithme se déroule de la façon suivante :

1. sélection : $j_{n+1} \in \arg\min_i |x^{i,n} - \xi^{n+1}|$
2. apprentissage de la grille :

$$\begin{cases} x^{j_{n+1},n+1} &= x^{j_{n+1},n} - \delta_{n+1}(x^{j_{n+1},n} - \xi^{n+1}) \\ x^{i,n+1} &= x^{i,n}, \quad i \neq j_{n+1} \end{cases}$$

3. apprentissage de l'erreur de quantification :

$$Q_N^{2,n+1} = Q_N^{2,n} - \frac{1}{n+1}(Q_N^{2,n} - |x^{j_{n+1},n} - \xi^{n+1}|^2), \quad Q_N^{2,0} = 0,$$

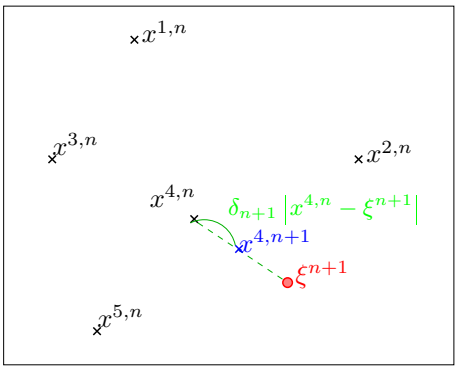

FIG. 2.1 – Exemple d'itération à l'étape n pour l'algorithme de Kohonen.

4. apprentissage des poids de quantification :

$$\pi_i^{n+1} = \pi_i^n - \frac{1}{n+1}(\pi_i^n - \mathbb{1}_{i=j_{n+1}}), \quad \pi_i^0 = 1/N, \quad 1 \leq i \leq N.$$

Cet algorithme est connu sous le nom *Competitive Learning Vector Quantization* (CLVQ) ou algorithme de Kohonen. Malheureusement, la convergence de celui-ci n'a été démontrée qu'en dimension 1 pour le cas général et en dimension supérieure lorsque $\mathbb{P}_X$ est à support compact. Les preuves se trouvent dans l'annexe de [1] et ses références. Notons que les résultats généraux sur la convergence des algorithmes stochastiques ne peuvent pas être appliqués dans ce cas à cause du manque de régularité du gradient. Un exemple d'itération est donné sur la figure 2.1. La figure 2.2 représente les quantificateurs obtenus à l'aide de l'algorithme de Kohonen pour une gaussienne centrée réduite en dimension 2.[1]

Pour terminer, nous allons traiter quelques considérations pratiques sur l'implantation de l'algorithme.

- Le premier problème concerne le choix de la suite (δ_n). Dans ce type d'algorithme, on prend souvent un pas du type $\delta_n = 1/n$. Ici, il est préférable que le pas ne décroisse pas trop vite pour les premières itérations. Les auteurs de [17] ont déterminé théoriquement et numériquement que, pour une distribution uniforme, le pas suivant est bien adapté :

$$\delta_n = \delta_0 \frac{a}{a + \delta_0 bn}$$

avec $a = 4N^{1/d}$, $b = \pi^2 N^{-2/d}$ et $\delta_0 = 1$. Cela permet d'avoir un pas à peu près constant lorsque n est petit et d'avoir un comportement asymptotique en $O(1/n)$ lorsque n est grand. Selon ces mêmes auteurs, ce pas

[1] Le pavage de Voronoï a été calculé grâce à un algorithme de Steven Fortune disponible à l'adresse `http://cm.bell-labs.com/who/sjf/`.

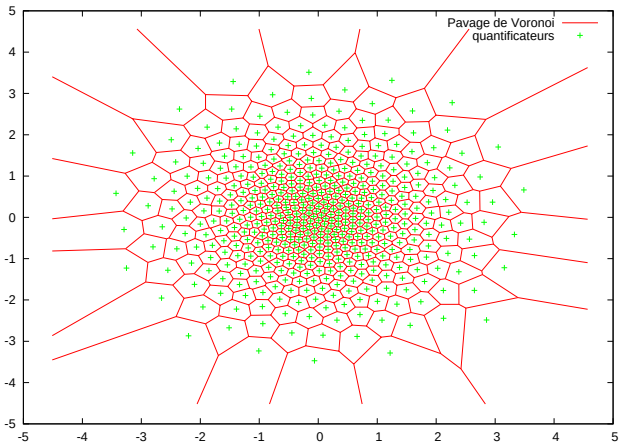

FIG. 2.2 – 500 quantificateurs obtenus pour une gaussienne centrée réduite en dimension 2 à l'aide de 10^8 itérations de l'algorithme de Kohonen. Le pavage de Voronoï est également représenté.

reste adapté pour d'autres distributions, notamment les distributions gaussiennes. Nos investigations numériques viennent confirmer ce résultat : le tableau suivant regroupe les distorsions obtenues en appliquant l'algorithme de Kohonen à une distribution gaussienne centrée, réduite, en dimension 1, avec 10^6 itérations, $N = 100$, pour différents pas δ_n.

$\frac{a}{a+bn}$	$\frac{a}{a+bn^{3/4}}$	$\frac{a}{a+bn^{1/2}}$	$\frac{a}{a+bn\log n}$	$\frac{1}{n}$	$\frac{1}{n^{1/2}}$	$\frac{a}{a+b^{1/2}n}$
$3,1.10^{-4}$	$7.0.10^{-4}$	$5,3.10^{-4}$	$3,2.10^{-4}$	$8,4.10^{-2}$	$3,7.10^{-3}$	$8,4.10^{-4}$

- Le deuxième point concerne l'initialisation de la grille. La solution la plus simple consiste à choisir N vecteurs tirés aléatoirement suivant la loi de X. Selon [17], il semblerait que cette méthode ne soit pas optimale lorsque N devient trop grand. Dans ce cas, il peut être intéressant de procéder par étapes successives, surtout lorsque la dimension d est importante : l'algorithme est appliqué une première fois pour un nombre restreint N_0 de quantificateurs puis N_1 quantificateurs sont ajoutés[2] au « centre » de la distribution, ou aléatoirement, et l'algorithme est relancé une nouvelle fois. Ces ajouts successifs sont réalisés jusqu'à obtenir $\sum_i N_i = N$. Notons qu'il convient de modifier le pas initial δ_0 à chaque fois que l'algorithme est relancé afin que le pas continue à diminuer. Nos essais numériques n'ont pas permis de mettre en lumière ce phénomène, car trop limités en dimension ($d \leq 2$) et en nombre de quantificateurs ($N \leq 500$). Enfin, Sylvain Maire nous a soumis l'idée d'initialiser la grille à l'aide de tirages quasi-aléatoires[3], et non plus pseudo-aléatoires. Nous n'avons malheureusement

[2] $N_1 \ll N_0$
[3] cf les suites d'Halton, de Warnock ou de Von der Corput par exemple.

pas eu le temps de tester numériquement cette idée.

- Enfin, remarquons que les poids de quantification π_i et la distorsion peuvent être également estimés par des méthodes de Monte-Carlo une fois que l'algorithme de Kohonen a été exécuté. Il s'avère que nos tests numériques ont montrés que l'estimation « en ligne » des poids de quantification et de la distorsion converge très lentement et qu'il est préférable de lancer une estimation par Monte-Carlo une fois que l'algorithme de Kohonen est terminé. Les figures 2.3 et 2.4 illustrent ce phénomène.

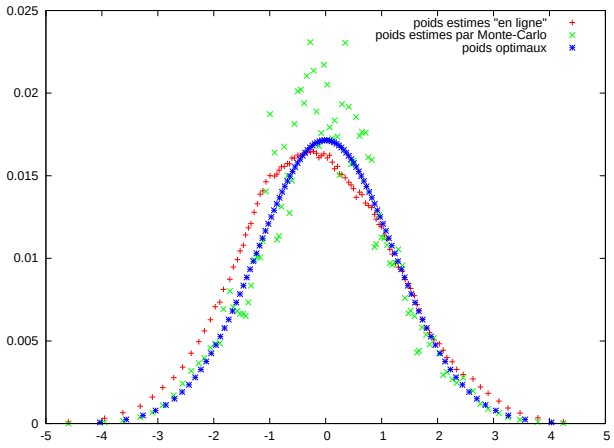

FIG. 2.3 – poids estimés « en ligne » et par Monte-Carlo pour une gaussienne centrée réduite en dimension 1, 100 quantificateurs et 10^6 itérations de l'algorithme de Kohonen.

2.1.3 Quelques résultats théoriques utiles pour la suite

Tout d'abord, on peut montrer[4] qu'une quantification optimal de X vérifie la propriété

$$\hat{X} = \mathbb{E}[X|\hat{X}]. \tag{2.1}$$

Concrètement, cela signifie que les points de $\hat{X}(\Omega)$ sont les barycentres de leurs cellules de Voronoï.

En pratique, nous allons utiliser ces méthodes de quantifcation pour calculer numériquement certaines intégrales. Ainsi, si l'on prend une fonction $\phi : \mathbb{R}^d \to \mathbb{R}$ mesurable, alors nous approcherons

$$\mathbb{E}[\phi(X)] = \int_{\mathbb{R}^d} \phi(\xi)\mathbb{P}_X(d\xi)$$

[4]Il suffit d'utiliser les méthodes employée dans la partie 3.2.3.

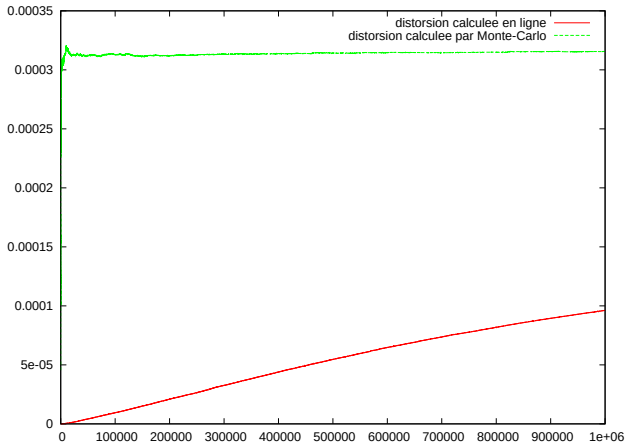

FIG. 2.4 – Estimation « en ligne » de la distorsion en dimension 1 pour 100 quantificateurs en fonction du nombre d'itérations de l'algorithme de Kohonen et estimation par Monte-Carlo, après 10^6 itérations de l'algorithme de Kohonen, en fonction du nombre de tirages.

par

$$\mathbb{E}[\phi(\hat{X})] = \int_{\mathbb{R}^d} \phi(\xi)\mathbb{P}_{\hat{X}}(d\xi) = \sum_{i=1}^{N} \hat{\pi}_i \phi(x_i)$$

Moyennant certaines hypothèses sur ϕ, nous pouvons contrôler l'erreur ainsi commise.

Proposition 2.1.1 *On suppose ϕ K-lipschitzienne. Alors*

$$\left|\mathbb{E}[\phi(X)] - \mathbb{E}[\phi(\hat{X})]\right| \leq K \left\|X - \hat{X}\right\|_2 .$$

La démonstration est immédiate.

Proposition 2.1.2 *On suppose ϕ différentiable et sa différentielle $D\phi$ K-lipschitzienne. On suppose également que $\hat{X}$ est une quantification optimale de X. Alors*

$$\left|\mathbb{E}[\phi(X)] - \mathbb{E}[\phi(\hat{X})]\right| \leq K \left\|X - \hat{X}\right\|_2^2 .$$

Démonstration : En appliquant une formule de Taylor nous obtenons

$$\left|\phi(X) - (\phi(\hat{X}) + D\phi(\hat{X}).(X - \hat{X}))\right| \leq K \left|X - \hat{X}\right|^2 .$$

Ainsi

$$\left|\mathbb{E}[\phi(X)] - \mathbb{E}[\phi(\hat{X})] - \mathbb{E}[D\phi(\hat{X}).(X - \hat{X})]\right| \leq K \left\|X - \hat{X}\right\|_2^2 .$$

Or, en utilisant (2.1) on a

$$\mathbb{E}[D\phi(\hat{X}).(X-\hat{X})] = \mathbb{E}\left[D\phi(\hat{X}).\mathbb{E}[X-\hat{X}|\hat{X}]\right] = 0.$$

□

Enfin, le théorème de Zador nous fourni le comportement asymptotique de la distorsion optimale lorsque N tend vers l'infini. Ce théorème est du à Zador et a été complété par Bucklew et Wise puis Graf et Luschgy.

Théorème 2.1.3 (Zador) *Soit $X \in L^2(\Omega, \mathbb{R}^d)$ telle que $\mathbb{P}_X(d\xi) = g(\xi)\lambda_d(d\xi) + \nu(d\xi)$ avec λ_d la mesure de Lebesgue sur $\mathbb{R}^d$ et $\nu \perp \lambda_d$. Alors*

$$\lim_N \left(N^{2/d} \min_{|\Gamma| \leq N} \left\| X - \hat{X}^\Gamma \right\|_2^2 \right) = C_d \left(\int_{\mathbb{R}^d} g^{\frac{d}{d+2}}(\xi) d\xi \right)^{\frac{d+2}{d}}$$

Les références pour la démonstration sont disponibles dans [1], [2], [17] et [16].

2.2 Quantification des chaînes de Markov

2.2.1 Présentation du problème

Nous avons déjà discrétisé en temps le processus forward X en considérant le schéma d'Euler

$$\bar{X}_{t_{k+1}}^{t_k, \bar{X}_{t_k}} = \bar{X}_{t_k} + hb(\bar{X}_{t_k}) + \sigma(\bar{X}_{t_k})(B_{t_{k+1}} - B_{t_k}).$$

Nous voulons maintenant approcher ce processus à temps discret par un processus $(\hat{X}_k)$ discrétisé en espace. On note $\Gamma_k = \left\{x_k^1, \ldots, x_k^{q_k}\right\}$ les quantificateurs au temps t_k et $\hat{p}_k^{i,j} = \mathbb{P}[\hat{X}_k = x_k^j | \hat{X}_{k-1} = x_{k-1}^i]$ les probabilités de transition. Alors, nous allons approximer

$$\bar{Y}_{t_k}^{t_k, x_k^i} = \mathbb{E}\left[\bar{Y}_{t_{k+1}}^{t_k, x_k^i} + hf(t_k, x_k^i, \bar{Y}_{t_{k+1}}^{t_k, x_k^i})\right]$$

par[5]

$$\hat{Y}_{t_k}^{t_k, x_k^i} = \sum_{j=1}^{q_{k+1}} \hat{p}_{k+1}^{i,j} \left(\hat{Y}_{t_{k+1}}^{t_{k+1}, x_{k+1}^j} + hf(t_k, x_k^i, \hat{Y}_{t_{k+1}}^{t_{k+1}, x_{k+1}^j}) \right).$$

Cette fois, il est possible de calculer numériquement le processus discrétisé en temps et en espace $(\hat{Y}_{t_k})$. Il reste alors à savoir comment nous allons discrétiser $\bar{X}$.

[5] Pour les EDSR réfléchies, il suffit de considérer le sup avec la fonction h.

2.2.2 Quantification marginale

La première méthode consiste à chercher en même temps les quantifications optimales des variables aléatoires $\bar{X}_{t_0}, \ldots, \bar{X}_{t_n}$. Ainsi, à chaque temps hk on considère la grille de quantification $\Gamma_k = \{x^1, ..., x^{q_k}\}$ composée de q_k points de $\mathbb{R}^d$, et on défini

$$\hat{X}_k = \mathrm{Proj}_{\Gamma_k}(\bar{X}_k), \quad k = 0, ..., n.$$

En pratique, l'algorithme de Kohonen est appliqué n fois en parallèle. Les probabilités de transitions peuvent être estimées directement ou calculées par Monte-Carlo. Pour une présentation complète de l'algorithme, il convient de se référer à [1]. Malheureusement, ce processus $\hat{X}$ ainsi construit n'est plus une chaîne de Markov. On sait néanmoins qu'il existe une chaîne de Markov $(\hat{X}_k^c)$ possédant les mêmes probabilités de transition $(\hat{p}_k^{ij})_{i,j}$ aux temps $k = 1, ..., n$.

2.2.3 Quantification markovienne

Le but de la seconde méthode est de conserver le caractère markovien du processus. Cette fois, nous cherchons les quantifications optimales pas à pas. Ainsi, si l'on écrit la chaîne de Markov $\bar{X}$ sous la forme

$$X_k = F_k(X_{k-1}, B_{t_k} - B_{t_{k-1}}),\ k = 1, ..., n,$$

alors on définit le nouveau processus $(\hat{X}_k)_k$ par

$$\hat{X}_k = \mathrm{Proj}_{\Gamma_k}(F_k(\hat{X}_{k-1}, \varepsilon_k)),\ k = 1, ..., n,$$

et

$$\hat{X}_0 = \mathrm{Proj}_{\Gamma_0}(X_0).$$

Le nouveau processus ainsi construit est encore une chaîne de Markov associée à la matrice de transition $[\hat{p}_k^{ij}]$ définie précédemment. Contrairement à la quantification marginale, l'algorithme de Kohonen n'est plus appliqué n fois en parallèle mais en série.

2.2.4 Application et limites

Le principal inconvénient de la première méthode est la perte du caractère markovien du processus. La seconde méthode a tendance, quant à elle, à propager les erreurs. Néanmoins, selon [16] il n'existe pas de différence majeure entre celles-ci. En pratique, elles nécessitent souvent des temps de calcul non négligeables. Toutefois, elles s'avèrent extrêmement bien adaptées lorsqu'il est possible de se ramener à des processus de référence que l'on peut quantifier puis stocker une fois pour toute. C'est le cas par exemple des processus de Black-Scholes du type

$$dX_t = \mu X_t dt + \sigma X_t dB_t.$$

Dans ce cas, il est possible d'écrire explicitement X_t en fonction de B_t. Ainsi, il suffit de quantifier de façon optimale le mouvement brownien standard une

fois pour toutes. En pratique on cherche juste la quantification optimale d'une gaussienne centrée réduite, la quantification optimale des B_{t_k} se déduisant alors par dilatation. De façon plus générale, cette idée est applicable pour tout processus s'écrivant explicitement en fonction du brownien. Les auteurs de [2] et [16] l'utilisent pour évaluer le prix d'options américaines en grande dimension[6]. Notons que des quantifications optimales de gaussiennes centrées réduites sont disponibles pour différentes dimensions et plusieurs nombres de quantificateurs à l'adresse `http://www.univ-paris12.fr/www/labos/cmup/homepages/printems/n01/`.

2.2.5 Utilisation de la quantification de la gaussienne

Une autre idée consiste à remplacer les gaussiennes $\Delta B^k = B_{t_{k+1}} - B_{t_k}$ par des quantifications optimales $\Delta\hat{B}^k$ dans le schéma d'Euler :

$$\hat{X}_{k+1} = \hat{X}_k + b(\hat{X}_k)h + \sigma(\hat{X}_k)\Delta\hat{B}^k.$$

Malheureusement, ce type de schéma est explosif : au temps t_k on obtient N^k points de grille, avec N le nombre de quantificateurs pour les gaussiennes. Afin d'éviter ce phénomène, il est envisageable de réduire au fur et à mesure le nombre de points de grille. Nous avons envisagé deux possibilités.

1. La première consiste à projeter les points de grilles obtenus à l'instant t_k sur une grille fixe du type $\delta\mathbb{Z}^d$ avec δ un pas constant. Si l'on note Π la projection sur cette grille fixe, on a alors

$$\left\{\begin{array}{rcl} \hat{X}_{t_0} & = & x_0 \\ \hat{X}_{t_{k+1}} & = & \Pi(\hat{X}_{t_k} + \mathcal{T}(t_k, \hat{X}_{t_k})) \\ avec & & \mathcal{T}(t_k, x) = b(x)h + \sigma(x)\Delta\hat{B}^k \end{array}\right. \tag{2.2}$$

   Cette méthode est intéressante algorithmiquement car la projection sur une grille fixe a une complexité indépendante du nombre de points de grille. Par contre elle ne semble pas topologiquement optimale car elle ne tient pas compte de la répartition non uniforme des points de grilles avant projection.

2. La seconde consiste à faire de la quantification markovienne sur le processus

$$\hat{X}_{k+1} = \hat{X}_k + b(\hat{X}_k)h + \sigma(\hat{X}_k)\Delta\hat{B}^k.$$

   Contrairement à ce que nous avons vu précédemment, on doit maintenant quantifier de façon optimale des variables aléatoires dont le support est déjà discret. Cette particularité fait que nous pouvons utiliser une nouvelle méthode pour arriver à nos fins : celle-ci est l'objet du chapitre suivant.

[6] Les exemples numériques vont jusqu'à $d = 10$.

Chapitre 3

Le problème du k-mean

3.1 Motivations

En mettant au point un algorithme de résolution d'équations différentielles stochastiques avec une quantification brownienne, nous avons été confronté au problème suivant : nous souhaitions réduire fortement un ensemble contenant un très grand nombre N de points pondérés de $\mathbb{R}^d$ en minimisant la perte d'information. L'idée la plus simple consiste à projeter les N points sur une grille régulière du type $\delta\mathbb{Z}^d$ avec δ un pas constant. Néanmoins, on se rend compte que cette solution n'est pas optimale car elle n'est pas flexible, c'est à dire que l'on ne peut pas raffiner le maillage là où il y a le plus de points avec de « grosses pondérations » et au contraire relacher le maillage là où les pondérations des points sont faibles. Une autre solution consiste à fixer un entier k puis déterminer k points qui minimisent l'erreur L^2 entre les N points initiaux et leur plus proche voisin parmi ces k points. Dans le cas particulier de poids équirépartis, on tombe sur la problématique du *k-mean*.

3.2 Problématique initiale

3.2.1 Le problème du *k-mean*

Soient $x_1, \ldots, x_N$ N vecteurs de $\mathbb{R}^d$ que l'on cherche à approcher par k vecteurs $y_1, \ldots, y_k$. Le but est alors de minimiser la fonction distorsion

$$D(y_1, \ldots, y_k) = \sum_{i=1}^{N} \|x_i - y(x_i)\|_2^2 \tag{3.1}$$

avec

$$y(x) = argmin_{y \in \{y_1, \ldots, y_k\}} \|x - y\|_2 . \tag{3.2}$$

Ce problème se retrouve notamment dans la problématique plus générale de l'apprentissage non supervisé où l'on cherche à classer N vecteurs donnés en k

groupes distincts. Un exemple très simple consiste à prendre N points générés à partir de k gaussiennes de centres $y_1, \ldots, y_k$ et de chercher à estimer ces centres par $\tilde{y}_1, ..., \tilde{y}_k$ en minimisant la distorsion D. De façon générale, ce problème est NP-complet, il est donc vain de tenter de le résoudre algorithmiquement de façon exacte. Dans toute la suite on suppose $N > k$[1] et les x_i distincts deux à deux.

3.2.2 Nécessité de définitions supplémentaires

Avant toute chose, il convient de revenir sur l'équation (3.2). En effet, lorsqu'il existe un point x qui est à égale distance de plusieurs points y_i, il peut apparaître un problème de choix dans la définition de la fonction $argmin$. Pour régler ce problème, définissons tout d'abord la notion de voisinages ou pavages de Voronoï.

Définition 3.2.1 (Pavages de Voronoï) *Soient $x_1, \ldots, x_N$ N vecteurs de $\mathbb{R}^d$ et $y_1, \ldots, y_k$ k vecteurs de $\mathbb{R}^d$. On dit que $\{\Gamma(y_1), \ldots, \Gamma(y_k)\}$ est un pavage de Voronoï des points $\{y_1, \ldots, y_k\}$ si c'est une partition de $\{x_1, \ldots, x_N\}$ vérifiant*

$$\forall i \in \{1, \ldots, k\}, \quad \forall x \in \Gamma(y_i), \quad \|x - y_i\|_2 = \min_{y \in \{y_1, \ldots, y_k\}} \|x - y\|_2$$

Pour la suite, nous allons privilégier un pavage de Voronoï particulier.

Définition 3.2.2 (Pavage de Voronoï) *Soient $x_1, \ldots, x_N$ N vecteurs de $\mathbb{R}^d$ et $y_1, \ldots, y_k$ k vecteurs de $\mathbb{R}^d$. On dit que $\left\{\hat{\Gamma}(y_1), \ldots, \hat{\Gamma}(y_k)\right\}$ est le pavage de Voronoï des points $\{y_1, \ldots, y_k\}$ si c'est une partition de $\{x_1, \ldots, x_N\}$ vérifiant*

$$\forall i \in \{1, \ldots, k\}, \quad \forall x \in \hat{\Gamma}(y_i), \quad \begin{cases} \|x - y_i\|_2 < \|x - y_j\|_2 & si\ j < i \\ \|x - y_i\|_2 \leq \|x - y_j\|_2 & sinon \end{cases}$$

Bien entendu, d'autres définitions eurent été acceptables. Nous pouvons maintenant définir proprement la fonction y :

$$\forall x \in \{x_1, \ldots, x_N\}, \quad y(x) = y_i \text{ si } x \in \hat{\Gamma}(y_i). \tag{3.3}$$

Remarque : Notons que $(y_1, \ldots, y_k)$ possède un unique pavage de Voronoï si, et seulement si, il n'existe aucun point de $\{x_1, \ldots, x_N\}$ qui est à égale distance de plusieurs points y_i.

3.2.3 Quelques résultats théoriques

Certaines preuves présentes dans cette partie sont adaptées de [5]. Les résultats principaux concernent l'existence d'un minimum global pour D^2 et la caractérisation des minima locaux de D^3.

[1] Sinon le problème est trivial.
[2] théorème 3.2.6
[3] théorème 3.2.8

Dans la suite nous considérerons la fonction $\mathcal{D}$ suivante

$$\begin{array}{rccc} \mathcal{D}: & \mathbb{R}^{d\times k}\times\mathcal{P}_k & \to & \mathbb{R} \\ & (y_1,\dots,y_k,V_1,\dots,V_k) & \mapsto & \sum_{i=1}^{k}\sum_{x\in V_i}\|x-y_i\|_2 \end{array} \tag{3.4}$$

avec $\mathcal{P}_k$ l'ensemble des partitions de $\{x_1,\dots,x_n\}$ en k régions.

Lemme 3.2.3 *Supposons que $\mathcal{D}$ est minimale en $(y_1^*,\dots,y_k^*,V_1^*,\dots,V_k^*)$, alors les V_i^* sont non vides.*

Démonstration : Le lemme est trivial si $k=1$. Sinon, supposons, sans perte de généralité, que V_1^* est vide. Comme $k<N$, il existe un V_i^* de cardinal strictement supérieur à un : V_2^* par exemple[4]. Soit $x\in V_2^*$ tel que $x\neq y_2^*$. Alors

$$\mathcal{D}(x,y_2^*,\dots,y_k^*,\{x\},V_2^*\setminus\{x\},\dots,V_k^*)<\mathcal{D}(y_1^*,\dots,y_k^*,V_1^*,\dots,V_k^*)$$

ce qui est absurde. □

Corollaire 3.2.4 *Supposons que $\mathcal{D}$ est minimale en $(y_1^*,\dots,y_k^*,V_1^*,\dots,V_k^*)$, alors les y_i^* sont distincts deux à deux.*

Démonstration : Si les y_i ne sont pas distincts deux à deux, nous pouvons supposer, sans perte de généralité, que $y_1^*=y_2^*$. Alors

$$\mathcal{D}(y_1^*,\dots,y_k^*,V_1^*,\dots,V_k^*)=\mathcal{D}(y_1^*,\dots,y_k^*,V_1^*\cup V_2^*,\{\varnothing\},\dots,V_k^*)$$

Le lemme 3.2.3 achève la preuve. □

Proposition 3.2.5 *$\mathcal{D}$ possède un minimum global. De plus, une condition nécessaire pour que $\mathcal{D}$ soit minimale est que $\{V_1,\dots,V_k\}$ soit un pavage de Voronoï de $\{y_1,\dots,y_k\}$ et que, simultanément, les y_i soient les barycentres des V_i.*

Démonstration : Comme $\mathcal{P}_k$ est un ensemble fini, il suffit de montrer qu'à $V_1,\dots,V_k$ fixés, $\mathcal{D}$ possède un minimum qu'elle atteint. Notons $\mathcal{D}_{\{V_i\}}$ la fonction $\mathcal{D}$ ou les V_i sont fixés. Il est évident que $\mathcal{D}_{\{V_i\}}$ est une fonction continue sur $\mathbb{R}^{d\times k}$. De plus $\mathcal{D}_{\{V_i\}}(y_1,\dots,y_k)\to+\infty$ lorsque $\sum_{i=1}^{k}\|y_i\|_2\to+\infty$. Ainsi, on peut se restreindre à un compact sur lequel $\mathcal{D}_{\{V_i\}}$ est bornée et atteint ses bornes.
On suppose maintenant que $\mathcal{D}$ atteint son minimum en $(y_1^*,\dots,y_k^*,V_1^*,\dots,V_k^*)$. Alors, la dérivée directionnelle de $\mathcal{D}$ par rapport à y_i^* suivant n'importe quel vecteur v de $\mathbb{R}^d$ est nulle. En simplifiant l'écriture[5], on a

$$\frac{1}{\varepsilon}(\mathcal{D}(y_i^*+\varepsilon v)-\mathcal{D}(y_i^*))=\sum_{x\in V_i^*}-2\langle x-y_i^*,v\rangle+\varepsilon\|v\|_2^2$$

[4] toujours sans perte de généralité
[5] On ne fait pas apparaître dans l'écriture de $\mathcal{D}$ les variables qui ne varient pas.

Lorsque $\varepsilon \to 0$, on trouve facilement, à l'aide du lemme 3.2.3, que

$$y_i^* = \frac{\sum_{x \in V_i^*} x}{\sum_{x \in V_i^*} 1}.$$

Ainsi les y_i^* sont les barycentres des V_i^*. Supposons maintenant que $\{V_1^*, \ldots, V_k^*\}$ ne soit pas un pavage de Voronoï de $\{y_1^*, \ldots, y_k^*\}$. Pour tout $x \in \{x_1, \ldots, x_N\}$, il existe deux entiers i et j tels que $x \in V_i^*$, $x \in \hat{\Gamma}(y_j^*)$ et $\|x - y_j\| \leq \|x - y_i\|$. De plus, il existe au moins un x pour lequel l'inégalité est stricte car sinon $\{V_1^*, \ldots, V_k^*\}$ serait un pavage de Voronoï. Ainsi,

$$\mathcal{D}(y_1^*, \ldots, y_k^*, \hat{\Gamma}(y_1^*), \ldots, \hat{\Gamma}(y_k^*)) < \mathcal{D}(y_1^*, \ldots, y_k^*, V_1^*, \ldots, V_k^*)$$

ce qui est absurde. Donc $\{V_1^*, \ldots, V_k^*\}$ est un pavage de Voronoï de $\{y_1^*, \ldots, y_k^*\}$. □

Théorème 3.2.6 (Minimum global) *La fonction D définie par (3.1) possède un minimum global qu'elle atteint en au moins un « point ». Si l'on note $(y_1^*, \ldots, y_k^*)$ un de ces « points », alors les y_i^* sont les barycentres des pavés de Voronoï $\hat{\Gamma}(y_i^*)$ associés.*

Démonstration : Remarquons que

$$D(y_1, \ldots, y_k) = \mathcal{D}(y_1, \ldots, y_k, \Gamma(y_1), \ldots, \Gamma(y_k))$$

pour n'importe quel pavage de Voronoï $\{\Gamma(y_1), \ldots, \Gamma(y_k)\}$ de $\{y_1, \ldots, y_k\}$. Or, d'après la proposition 3.2.5, $\mathcal{D}$ atteint son minimum global en des « points » de la forme $(y_1, \ldots, y_k, \Gamma(y_1), \ldots, \Gamma(y_k))$. Ainsi D et $\mathcal{D}$ ont le même minimum qu'elles atteignent aux mêmes $(y_1, \ldots, y_k)$. □

On peut exprimer cette condition nécessaire différemment : Les points pour lesquels D est minimale sont à chercher parmi les points fixes de $\Phi_{\hat{\Gamma}}$ avec

$$\begin{array}{rccc} \Phi_{\hat{\Gamma}} : & \mathbb{R}^{d \times k} & \to & \mathbb{R}^{d \times k} \\ & (y_1, \ldots, y_k) & \mapsto & (\phi_{\hat{\Gamma}}^1(y_1, \ldots, y_k), \ldots, \phi_{\hat{\Gamma}}^k(y_1, \ldots, y_k)) \end{array} \tag{3.5}$$

$$\phi_{\hat{\Gamma}}^i(y_1, \ldots, y_k) = \begin{cases} y_i & \text{si } \left|\hat{\Gamma}(y_i)\right| = 0 \\ \frac{1}{\left|\hat{\Gamma}(y_i)\right|} \sum_{x \in \hat{\Gamma}(y_i)} x & \text{sinon} \end{cases}$$

Proposition 3.2.7 *Pour tout $(y_1, \ldots, y_k) \in \mathbb{R}^{d \times k}$,*

- *$D(\Phi_{\hat{\Gamma}}(y_1, \ldots, y_k)) = D(y_1, \ldots, y_k)$ si $(y_1, \ldots, y_k)$ est un point fixe de $\Phi_{\hat{\Gamma}}$.*
- *$D(\Phi_{\hat{\Gamma}}(y_1, \ldots, y_k)) < D(y_1, \ldots, y_k)$ sinon.*

Cette proposition sera très utile par la suite pour construire des algorithmes de « résolution » du *k-mean*.

Démonstration : Si $(y_1, \ldots, y_k)$ est un point fixe de Φ, le résultat est trivial. Sinon, il suffit de voir que

$$\mathcal{D}(\phi^1_{\hat{\Gamma}}(y_1, \ldots, y_k), \ldots, \phi^k_{\hat{\Gamma}}(y_1, \ldots, y_k), \hat{\Gamma}(y_1), \ldots, \hat{\Gamma}(y_k)) < D(y_1, \ldots, y_k).$$

Or

$$D(\Phi_{\hat{\Gamma}}(y_1, \ldots, y_k)) \leq \mathcal{D}(\phi^1_{\hat{\Gamma}}(y_1, \ldots, y_k), \ldots, \phi^k_{\hat{\Gamma}}(y_1, \ldots, y_k), \hat{\Gamma}(y_1), \ldots, \hat{\Gamma}(y_k)).$$

□

Nous allons maintenant caractériser les minima locaux de D.

Théorème 3.2.8 (Minima locaux) *Considérons k vecteurs $(y_1, \ldots, y_k)$ deux à deux distincts. Alors $(y_1, \ldots, y_k)$ est un minimum local de D si, et seulement si, c'est un point fixe de $\Phi_{\hat{\Gamma}}$ possédant un unique pavage de Voronoï.*

Démonstration : Si $k = 1$ le résultat est trivial. Sinon, prouvons tout d'abord le lemme suivant

Lemme 3.2.9 *Soient $y_1, \ldots, y_k$ k vecteurs de $\mathbb{R}^d$ ayant un unique pavage de Voronoï. Alors, $(y_1, \ldots, y_k)$ est un minimum local de D ssi c'est un point fixe de Φ.*

On suppose que $\{y_1, \ldots, y_k\}$ possède un unique pavage de Voronoï. Alors,

$$\exists \varepsilon > 0 \quad \forall i \in \{1, \ldots, k\} \quad \forall x \in \hat{\Gamma}(y_i), \quad \|x - y_i\|_2 \leq \|x - y_j\|_2 - \varepsilon \quad j \neq i.$$

Pour tout $(y'_1, \ldots, y'_k)$ tels que $\sup_{1 \leq i \leq k} \|y_i - y'_i\|_2 < \varepsilon/2$, on a

$$\|x - y'_i\|_2 \leq \|x - y_i\| + \|y_i - y'_i\| \leq \|x - y_j\| - \varepsilon/2 \quad j \neq i.$$

Donc, $\{y'_1, \ldots, y'_k\}$ a le même pavage de Voronoï que $\{y_1, \ldots, y_k\}$. Ainsi, D est différentiable et convexe dans un voisinage de $\{y_1, \ldots, y_k\}$. Donc, $(y_1, \ldots, y_k)$ est un minimum local de D ssi toutes les dérivées directionnelles de $\mathcal{D}$ suivant n'importe quel vecteur v de $\mathbb{R}^{d \times k}$ sont nulles. En reprenant les calculs de la démonstration de la proposition 3.2.5, on obtient que $(y_1, \ldots, y_k)$ est un minimum local de D ssi les y_i sont les barycentres des V_i. □

Lemme 3.2.10 *Si $(y_1, \ldots, y_k)$ est un minimum local de D, alors c'est un point fixe de $\Phi_{\hat{\Gamma}}$.*

Supposons que $(y_1, \ldots, y_k)$ n'est pas un point fixe de $\Phi_{\hat{\Gamma}}$ et notons $g_1, \ldots, g_k$ les barycentres de $\hat{\Gamma}(y_1), \ldots, \hat{\Gamma}(y_k)$. Alors, les y_i sont différents des g_i, donc la dérivée directionnelle de $\mathcal{D}$ dans la direction $(g_1 - y_1, \ldots, g_k - y_k)$ est strictement négative. Ainsi, il existe $M > 0$ tel que pout tout $0 < \varepsilon < M$ on a

$$\mathcal{D}(y_1 + \varepsilon(g_1 - y_1), \ldots, y_k + \varepsilon(g_k - y_k), \hat{\Gamma}(y_1), \ldots, \hat{\Gamma}(y_k)) < D(y_1, \ldots, y_k)$$

Or, on a déjà vu dans la démonstration de la proposition 3.2.5 que

$$\begin{aligned} &D(y_1 + \varepsilon(g_1 - y_1), \dots, y_k + \varepsilon(g_k - y_k)) \\ &\leq \mathcal{D}(y_1 + \varepsilon(g_1 - y_1), \dots, y_k + \varepsilon(g_k - y_k), \hat{\Gamma}(y_1), \dots, \hat{\Gamma}(y_k)). \end{aligned}$$

Ainsi,

$$D(y_1 + \varepsilon(g_1 - y_1), \dots, y_k + \varepsilon(g_k - y_k)) < D(y_1, \dots, y_k)$$

ce qui signifie que $(y_1, \dots, y_k)$ n'est pas un minimum local de D. □

Il reste maintenant à montrer que si $(y_1, \dots, y_k)$ est un minimum local de $\mathcal{D}$ tel que les y_i sont distincts deux à deux, alors il possède un unique pavage de Voronoï. Pour cela nous démontrerons la contraposée. Supposons donc que $\{y_1, \dots, y_k\}$ ne possède pas un unique pavage de Voronoï. Soit x_i un point équidistant de y_j et y_k. Sans perte de généralité, on peut supposer que x_1 est équidistant de y_1 et y_2. Notons que x_1 est dans le pavé de Voronoï de y_1. On considère le pavage de Voronoï suivant

$$\{\Gamma(y_1), \dots, \Gamma(y_k)\} = \left\{\hat{\Gamma}(y_1) \setminus \{x_1\}, \hat{\Gamma}(y_2) \cup \{x_1\}, \dots, \hat{\Gamma}(y_k)\right\}.$$

L'idée est alors de « bouger » y_2 vers son nouveau barycentre. Pour cela, il faut d'abord s'assurer que y_2 n'est pas confondu avec le barycentre de $\hat{\Gamma}(y_2) \cup \{x_1\}$ noté g_2. D'après le lemme 3.2.10, y_2 est le barycentre de $\hat{\Gamma}(y_2)$. Il suffit donc de montrer que $\hat{\Gamma}(y_2) \cup \{x_1\}$ et $\hat{\Gamma}(y_2)$ n'ont pas le même barycentre. On montre facilement que ces deux ensembles ont le même barycentre si, et seulement si, x_1 est confondu avec le barycentre de $\hat{\Gamma}(y_2)$, à savoir y_2. Or x_1 étant équidistant de y_1 et y_2, cela est vrai si, et seulement si, y_1 et y_2 sont confondus, ce qui est faux par hypothèse. Donc $g_2 - y_2 \neq 0$. On peut alors reprendre l'idée de la démonstration du lemme 3.2.10 : Il existe $M > 0$ tel que pout tout $0 < \varepsilon < M$ on a

$$\mathcal{D}(y_1, y_2 + \varepsilon(g_2 - y_2), \dots, y_k), \Gamma(y_1), \dots, \Gamma(y_k)) < D(y_1, \dots, y_k).$$

Or,

$$D(y_1, y_2 + \varepsilon(g_2 - y_2), \dots, y_k) \leq \mathcal{D}(y_1, y_2 + \varepsilon(g_2 - y_2), \dots, y_k), \Gamma(y_1), \dots, \Gamma(y_k)).$$

Ainsi,

$$D(y_1, y_2 + \varepsilon(g_2 - y_2), \dots, y_k) < D(y_1, \dots, y_k).$$

Donc $(y_1, \dots, y_k)$ n'est pas un minimum local de $\mathcal{D}$. □

Remarque : Le fait que les y_i soient distincts est nécessaire dans la proposition 3.2.8. En effet, il est possible de trouver des contre-exemples lorsque cette hypothèse est assouplie. Pour $k = 3$, on peut par exemple prendre y_1 comme le barycentre des x_i et $y_2 = y_3$ « suffisamment » éloignés des x_i.

3.2.4 L'algorithme du k-mean ou algorithme de Lloyd généralisé

Les résultats obtenus dans la partie 3.2.3 nous permettent de donner une méthode de résolution exacte du problème du *k-mean* :

- On parcourt l'ensemble des partitions de $\{x_1, \dots, x_N\}$ en k régions.
- Pour chaque partition, on calcule les barycentres $y_1, \dots, y_k$ associés.
- Pour chacun de ces k-uplets, on calcule la distorsion $D(y_1, \dots, y_k)$.
- On conserve le k-uplé qui a la distorsion la plus faible.

Néanmoins, cet algorithme de résolution n'est pas applicable en pratique car le nombre de partitions de N éléments en k régions est explosif. En effet, si l'on note $N_{N,k}$ ce nombre, alors ce dernier vérifie la récurrence suivante :

$$N_{N,k} = N_{N-1,k-1} + kN_{N-1,k}.$$

Pour se donner une idée des ordres de grandeur, on trouve, par exemple, $N_{n,2} = 2^{n-1} - 1$ et $N_{12,5} > 10^6$.

Une idée pour approcher un minimum de la distorsion consiste à vouloir profiter de la décroissance de $\Phi_{\hat{\Gamma}}$ (proposition 3.2.7). C'est exactement ce qui est utilisé dans l'algorithme du k-mean, également appelé algorithme de Lloyd généralisé[6] :

1. On initialise l'algorithme en tirant les y_i^0 deux à deux distincts parmi les $\{x_1, \dots, x_N\}$.
2. On itère $(y_1^{n+1}, \dots, y_k^{n+1}) = \Phi_{\hat{\Gamma}}(y_1^n, \dots, y_k^n)$ tant que l'on a pas atteint un point fixe.

Un exemple d'itération pour cet algorithme est illustré sur la figure 3.1.

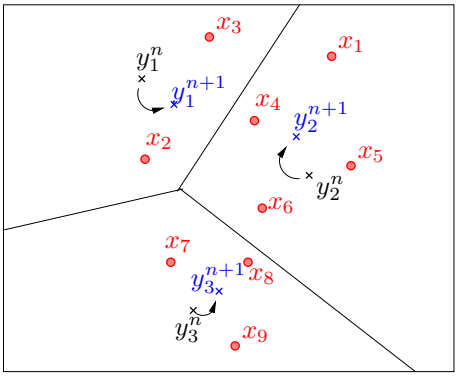

FIG. 3.1 – Exemple d'itération de l'algorithme du k-mean.

Théorème 3.2.11 *L'algorithme du k-mean se termine.*

[6] À l'origine, l'algorithme de Lloyd ne concerne que la dimension un.

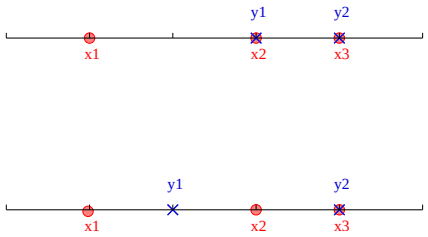

FIG. 3.2 – Cas où l'algorithme du k-mean classique ne se termine pas avec un minimum local. La figure du haut représente l'initialisation et la figure du bas, la première itération.

Démonstration : Posons $d_n = D(y_1^n, \dots, y_k^n)$. D'après la proposition 3.2.7, cette suite est strictement décroissante tant qu'un point fixe de $\Phi_{\hat{\Gamma}}$ n'est pas atteint. Ainsi, tant qu'un point fixe n'est pas atteint, les pavages de Voronoï $\mathcal{P}_i = \left\{\hat{\Gamma}(y_1^i), \dots, \hat{\Gamma}(y_k^i)\right\}$ sont deux à deux distincts. En effet, si $\mathcal{P}_i = \mathcal{P}_j$, alors $d_{i+1} = d_{j+1}$ ce qui contredit la décroissance stricte de la suite $(d_i)_i$. Pour conclure sur la terminaison de l'algorithme, il suffit de rappeller qu'il existe un nombre fini de partitions d'un ensemble de N éléments en k régions : l'algorithme a au plus $N_{N,k}$ étapes. □

Malheureusement, le point fixe atteint par l'algorithme présenté précédemment n'est pas forcément un minimum local car on n'est pas certain que celui-ci possède un unique pavage de Voronoï. La figure 3.2 montre justement un cas où le point fixe atteint n'est pas un minimum local. En effet, on a

$$D(y_1^1 - \varepsilon, y_2^1) = (1-\varepsilon)^2 + 1 = 2 - 2\varepsilon + \varepsilon^2.$$

Donc pour $\varepsilon > 0$ suffisamment petit,

$$D(y_1^1 - \varepsilon, y_2^1) < 2 = D(y_1^1, y_2^1).$$

On peut néanmoins le modifier légèrement pour arriver à nos fins :

1. On initialise l'algorithme en tirant les y_i^0 deux à deux distincts parmi les $\{x_1, \dots, x_N\}$.
2. On itère $(y_1^{n+1}, \dots, y_k^{n+1}) = \Phi_{\hat{\Gamma}}(y_1^n, \dots, y_k^n)$ tant que l'on n'a pas atteint un point fixe.
3. Si ce point fixe possède un unique pavage de Voronoï, l'algorithme est fini. Sinon, on détermine deux points y_i^n et y_{i+j}^n tels qu'il existe un $x \in \{x_1, \dots, x_N\}$ qui soit équidistant de ces deux points. On considère alors le pavage de Voronoï suivant

   $$\Gamma(y_i^n) = \hat{\Gamma}(y_i^n) \backslash \{x\}\,, \quad \Gamma(y_{i+j}^n) = \hat{\Gamma}(y_{i+j}^n) \cup \{x\}\,, \quad \Gamma(y_k^n) = \hat{\Gamma}(y_k^n)\, k \neq i, i+j$$

   et on pose $(y_1^{n+1}, \dots, y_k^{n+1}) = \Phi_\Gamma(y_1^n, \dots, y_k^n)$. On retourne ensuite à l'étape 2.

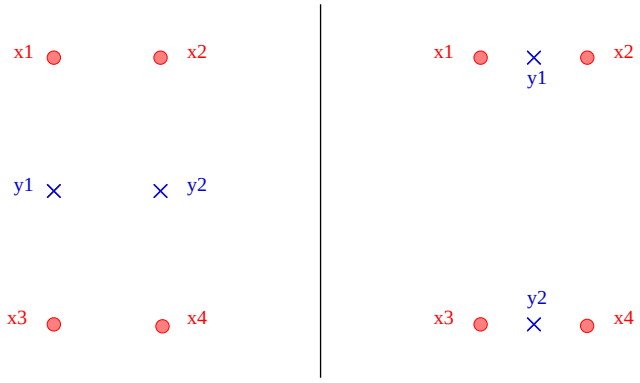

FIG. 3.3 – Un minimum local (gauche) et un minimum global (droite) pour les mêmes $\{x_1, \ldots, x_N\}$.

Théorème 3.2.12 *L'algorithme de k-mean modifié se termine et le point fixe renvoyé est un minimum local de D.*

Démonstration : Montrons tout d'abord qu'à chaque étape n, les y_i^n sont deux à deux distincts. Pour cela, considérons, sans perte de généralité, les deux points y_1^n et y_2^n. On note E_1^n et E_2^n les deux demi-espaces ouverts séparés par l'hyperplan médiateur de y_1^n et y_2^n. Alors, $y_1^{n+1} \in \bar{E}_1^n$ et $y_2^{n+1} \in E_2^n$. En effet, tout élément de $\hat{\Gamma}(y_1^n) \cup \hat{\Gamma}(y_2^n)$ équidistant de y_1^n et y_2^n appartient à $\hat{\Gamma}(y_1^n)$. Ainsi, y_1^{n+1} et y_2^{n+1} sont distincts. Montrons maintenant que les itérations éventuelles de l'étape 3. font décroître la distorsion. Supposons que $(y_1^n, \ldots, y_k^n)$ est un point fixe ne possédant pas un unique pavage de Voronoï. Alors, comme les y_i^n sont deux à deux distincts, nous avons vu dans la démonstration de la proposition 3.2.8 que $(y_1^n, \ldots, y_k^n)$ n'est pas un point fixe de Φ_Γ. Ainsi, la proposition 3.2.7[7] nous assure que $d_{n+1} < d_n$. Comme (d_i) est toujours une suite strictement décroissante, on peut réappliquer la démarche de la proposition 3.2.11 pour démontrer la terminaison de l'algorithme du k-mean modifié. Enfin, le point fixe renvoyé est un minimum local de D d'après la proposition 3.2.8. □

Malheureusement, en pratique nous ne sommes pas du tout assuré que la distorsion du point fixe retourné soit proche de la distorsion minimale. Il est même possible de trouver des exemples où la distorsion du point fixe retourné est très éloignée de la distorsion minimale (figure 3.3). Nous ne pourrons donc pas nous contenter de cet algorithme.

Remarque : Les auteurs de [22] proposent l'algorithme suivant :

[7] En théorie cette proposition n'est démontrée que pour $\Phi_{\hat{\Gamma}}$, néanmoins on démontre très facilement qu'elle reste valable pour Φ_Γ

1. On initialise l'algorithme en tirant les y_i^0 deux à deux distincts parmi les $\{x_1, \ldots, x_N\}$.
2. On itère $(y_1^{n+1}, \ldots, y_k^{n+1}) = \Phi_{\hat{\Gamma}}(y_1^n, \ldots, y_k^n)$ jusqu'à ce que
   - $D(y_1^n, \ldots, y_k^n) = \mathcal{D}(y_1^{n+1}, \ldots, y_k^{n+1}, \hat{\Gamma}(y_1^n), \ldots, \hat{\Gamma}(y_k^n))$
   - ou $\mathcal{D}(y_1^{n+1}, \ldots, y_k^{n+1}, \hat{\Gamma}(y_1^n), \ldots, \hat{\Gamma}(y_k^n)) = D(y_1^{n+1}, \ldots, y_k^{n+1})$

Ces mêmes auteurs « démontrent » que cet algorithme se termine et que l'on obtient un minimum local[8]. Malheureusement, ce résultat est faux : il est possible de trouver des exemples comme celui de la figure 3.2 qui le contredisent.

3.3 l'algorithme de k-mean utilisé

L'algorithme de k-mean initial est le point de départ de nombreuses variantes basées, le plus souvent, sur des heuristiques améliorant la vitesse de convergence. Nous nous sommes intéressés à l'algorithme de [10] pour deux raisons.

- Cet algorithme est pensé pour optimiser les problèmes de recherche du plus proche voisin, c'est-à-dire la phase qui demande le plus de temps.
- Une version codée de cet algorithme est disponible en C++ sous licence GPL.

Avant de le présenter, il convient de justifier son utilisation en mettant en lumière les problèmes rencontrés dans le codage du k-mean. Indéniablement, la phase la plus délicate à coder est la recherche pour chaque x_i d'un plus proche voisin parmi les points $y_1^j, \ldots, y_k^j$. Reformulée de façon plus générale, la problématique consiste à déterminer le plus proche voisin d'un point x parmi k points $p_1, \ldots, p_k$. En dimension 1, ce problème est facilement optimisable en classant ces points. On peut alors utiliser une recherche dichotomique pour trouver le plus proche voisin et ainsi obtenir une complexité en $O(\log k)$. Malheureusement, ce procédé n'est pas facilement généralisable en dimension supérieure. Néanmoins, une structure de données appelée *kd-tree*[9] permet de garder une complexité en $O(\log k)$. L'idée est de construire un arbre binaire qui partitionne l'espace par des hyperplans. Évidemment, il est important que l'arbre soit bien équilibré, c'est-à-dire que les hyperplans partagent l'espace restant en deux sous-espaces contenant approximativement le même nombre de points. Concrètement, un *kd-tree* est défini récursivement comme :

- soit un ensemble vide $\emptyset$,
- soit un couple $\{pt, \Theta\}$ composé d'un point de $\mathbb{R}^d$ et d'un domaine Θ de $\mathbb{R}^d$ dont les frontières sont alignées avec les axes des coordonnées[10],
- soit un quadruplé $\{pt, \Theta, hp, fils\_gauche, fils\_droit\}$ composé d'un point pt de $\mathbb{R}^d$, d'un domaine Θ de $\mathbb{R}^d$ dont les frontières sont alignées avec les axes des coordonnées, d'un hyperplan hp de $\mathbb{R}^d$ et de deux *kd-tree* $fils\_gauche$ et $fils\_droit$.

[8] théorème 5 page 83 et théorème 11 page 85.

[9] Cette structure de données n'est qu'un exemple parmi d'autres. Elle fait partie de la classe plus générale des *BSP trees* (Binary Space Partitioning) où l'on trouve également le *Principal Axis tree*. On peut également citer d'autres classes comme les *Quadtrees* et les *Octrees*.

[10] Ce domaine peut-être vu comme un hyper-rectangle dont les bornes peuvent-être infinies.

Algorithme 1 : Construction

Entrées : i un entier compris entre 1 et d, $\{pt_1, \ldots, pt_j\}$ un ensemble de points de $\mathbb{R}^d$ et Θ un domaine de $\mathbb{R}^d$ dont les frontières sont alignées avec les axes des coordonées.

Sorties : Un *kd-tree*.

si $j \leq 1$ **alors**
  si $j = 0$ **alors**
    retourner $\emptyset$
  sinon
    retourner $\{pt_1, D\}$
sinon
  $H \leftarrow$ l'hyperplan perpendiculaire à l'axe de la i$^{\text{e}}$ coordonnée, passant par un des points $x \in \{pt_1, \ldots, pt_j\}$ et séparant les points restants en deux ensembles E_1 et E_2 de même cardinal, à 1 près;
  L'hyperplan partitionne Θ en deux sous-domaines Θ_1 et Θ_2 tels que $E_1 \subset \Theta_1$ et $E_2 \subset \Theta_2$;
  $fils\_gauche \leftarrow$ Construction$(i[d] + 1, E_1, \Theta_1)$;
  $fils\_droit \leftarrow$ Construction$(i[d] + 1, E_2, \Theta_2)$;
  retourner $\{x, D, H, fils\_gauche, fils\_droit\}$

Pour construire un *kd-tree* contenant les points $p_1, \ldots, p_k$, il suffit alors d'utiliser l'algorithme 1 avec les bons arguments : $Construction(1, \{p_1, \ldots, p_k\}, \mathbb{R}^d)$. Notons que le coût de création de l'arbre est en $O(k \log k)$. Un exemple de construction est représenté sur la figure 3.4.

Une telle construction nous permet d'obtenir une partition de l'espace qui va nous servir pour le problème du plus proche voisin. Supposons qu'un *kd-tree* a été construit pour les points $p_1, \ldots, p_k$ et que l'on cherche le plus proche voisin de x parmi ces points. La recherche va se dérouler en deux phases :

1. le *kd-tree* étant une partition de $\mathbb{R}^d$, l'arbre est descendu jusqu'à la feuille représentant le domaine où se situe x. Cette feuille est associée au point p_i.[11]
2. On parcourt une nouvelle fois l'arbre en s'arrêtant uniquement sur les nœuds représentant les domaines de $\mathbb{R}^d$ qui intersectent la boule de centre x et de rayon $|x - p_i|$.[12]

Un exemple de recherche du plus proche voisin basé sur l'exemple de la figure 3.4 est illustré sur la figure 3.5. Pour une présentation beaucoup plus consistante de la structure *kd-tree* et de la recherche du plus proche voisin, le lecteur

[11] Si, l'on tombe sur une feuille $\emptyset$, on considère alors le point du nœud père.

[12] En pratique, l'arbre est « remonté » depuis le nœud contenant p_i. De plus, lorsque l'algorithme trouve un meilleur candidat $p_{i'}$, la suite de la recherche est simplifiée en ne considérant plus que les domaines qui intersectent la boule de centre x et de rayon $|x - p_{i'}|$ parmi les domaines restant à explorer.

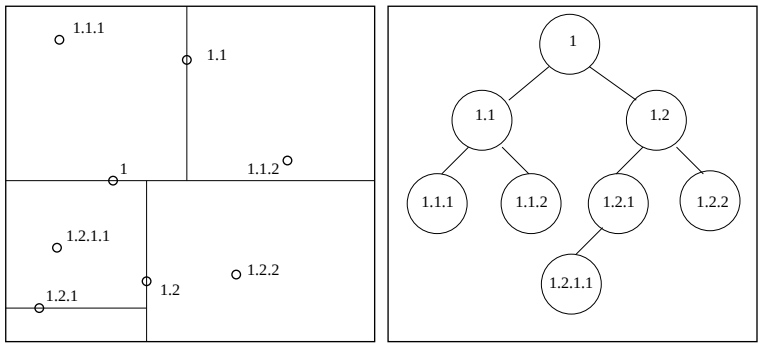

FIG. 3.4 – Exemple de construction de *kd-tree* en deux dimensions. Les hyperplans sont représentés à gauche et l'arbre correspondant à droite.

peut se reporter au rapport [15]. Il peut également consulter l'adresse `http://fr.wikipedia.org/wiki/Principal_Axis_Tree` pour une présentation du *Principal Axis tree*, une variante du *kd-tree*.

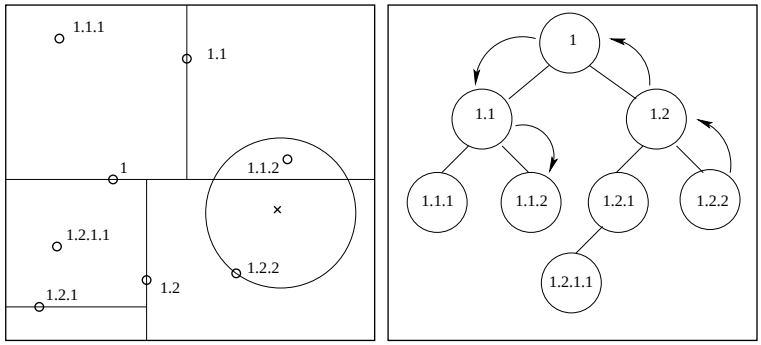

FIG. 3.5 – Exemple de recherche de plus proche voisin à l'aide de la structure *kd-tree*. La croix représente le point pour lequel on cherche le plus proche voisin. La recherche dans l'arbre est représentée à droite.

Revenons maintenant au problème d'origine. Si l'on applique directement la structure de *kd-tree* à l'algorithme du k-mean, on doit alors construire un nouvel arbre à chaque itération puisque tous les y_i sont modifiés. L'algorithme proposé dans [10] propose une utilisation beaucoup plus intéressante de la structure de *kd-tree* puisque l'arbre est, cette fois, construit sur les données x_i qui, par définition, ne sont pas modifiées. Vu que le problème n'est pas symétrique, il faut alors retravailler sur la structure pour pouvoir résoudre le problème de plus proche voisin. Cette fois, chaque nœud de l'arbre contient les éléments suivants :

- La somme des vecteurs x_i associés au nœud,
- le nombre de vecteurs x_i associés au nœud,

- le plus petit hyper-rectangle de $\mathbb{R}^d$ aligné avec les axes des coordonnées et contenant tous les points associés au nœud.

Dans un souci de clarté, nous appellerons cette structure *kd-tree* modifié. On définit également un centre z comme :

- un vecteur $z.y$ représentant un élément de $\left\{y_1^{n-1}, ..., y_k^{n-1}\right\}$, si l'on est à la nième itération,
- une somme $z.somme$ de vecteurs de $\{x_1, ..., x_N\}$ tels que $z.y$ soit leur plus proche voisin, c'est-à-dire $z.somme = \sum_{x \in \Gamma(z.y)} x$,
- le nombre $z.nb$ de vecteurs sommés dans $z.somme$, $z.nb = |\Gamma(z.y)|$.

Z_n représentera l'ensemble des centres à la nième itération, c'est-à-dire les k centres représentant les k vecteurs $y_1^{n-1}, ..., y_k^{n-1}$. Notons que pour passer à l'itération suivante il suffit de remplacer, pour chaque $z \in Z_n$, $z.y$ par $z.somme/z.nb$ (si $z.nb > 0$). Il convient donc de calculer, à chaque itération et pour chaque centre z, les quantités $z.nb$ et $z.somme$: c'est le rôle de l'algorithme 2.

Algorithme 2 : Traitement

Entrées : Z un sous-ensemble de $\left\{y_1^{n-1}, ..., y_k^{n-1}\right\}$ (lorsque l'on est à la nième itération), T un *kd-tree* modifié.

si T *est un nœud terminal* **alors**
  pour chaque x_i *associé au nœud* **faire**
    $z \leftarrow$ le plus proche voisin de x_i dans Z;
    $z.somme \leftarrow z.somme + x_i$;
    $z.nb = z.nb + 1$;
sinon
  $z^* \leftarrow$ le point de Z le plus proche du centre de l'hyper-rectangle du nœud courant;
  pour chaque $z \in Z \setminus \{z^*\}$ **faire**
    On calcule l'hyperplan bissecteur de $z - z^*$;
    si *cet hyperplan n'intersecte pas l'hyper-rectangle du nœud courant* **alors**
      $Z \leftarrow Z \setminus \{z\}$;
      (cf l'exemple de la figure 3.6.)
  si $Z = \{z^*\}$ **alors**
    pour chaque x_i *associé au nœud* **faire**
      $z^*.somme \leftarrow z^*.somme + x_i$;
      $z^*.nb = z^*.nb + 1$;
  sinon
    Traitement(Z,$fils\_gauche$ de T);
    Traitement(Z,$fils\_droit$ de T);

Notons que l'initialisation de l'algorithme du k-mean proposé dans [10] est faite en tirant au hasard k vecteurs dans les données. De plus, les critères d'arrêt

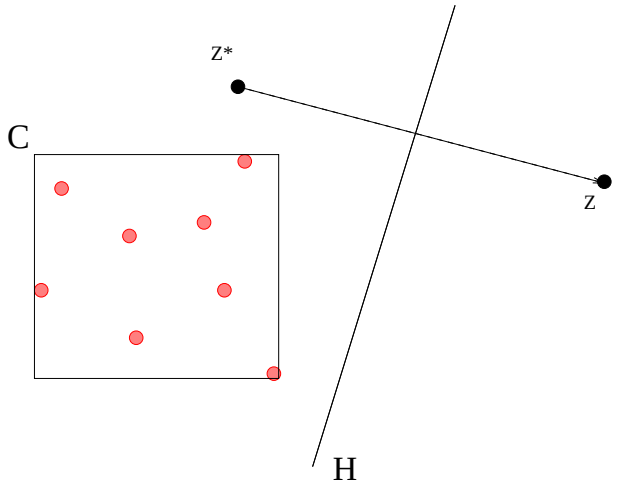

FIG. 3.6 – Le candidat z est retiré de Z car l'hyperplan H n'intersecte pas l'hyper-rectangle C.

sont différents des critères théoriques vus précédemment. En pratique, le programme demande en paramètre le nombre d'itérations total désiré. Puis l'algorithme est fragmenté en « phases », elles même fragmentées en « sous-phases ». Chaque « phase » commence par une tirage aléatoire des $\{y_1, \ldots, y_k\}$ parmi les $\{x_1, \ldots, x_N\}$, puis la première « sous-phase » débute. Une « sous-phase » peut se terminer seulement sous deux conditions : soit la variation relative de distorsion est inférieure à un paramétre donné, soit le nombre d'itérations de l'algorithme dans la « sous-phase » est supérieur à un autre paramètre donné. Une « phase » se termine dès que la seconde condition est vérifiée. L'algorithme se termine dès que le nombre d'itérations total désiré est atteint. Il retourne alors les candidats qui avaient la distorsion la plus faible. Notons que les auteurs de [10] ont également implanté une version couplant l'algorithme du k-mean classique avec une heuristique du type « recuit simulé ». Néanmoins, nous n'en parlerons pas car, dans notre cas, elle n'apporte pas d'amélioration notable. Enfin, en ce qui concerne la complexité, les auteurs de [10] ont démontré un résultat[13] que nous n'allons pas détailler, car trop compliqué. Néanmoins, comme le but d'une structure *kd-tree* est d'effectuer une recherche de plus proche voisin en $O(\log N)$, nous supposerons que la complexité de cet algorithme est en

$$O(N \log N + n_{iter} k \log N)$$

avec n_{iter} le nombre d'itérations. Le premier terme concerne la création de l'arbre et le second terme, les parcours de cet arbre.

[13] théorème 1, page 884

3.4 Le *k-mean pondéré*

3.4.1 Nouvelle problématique

Le problème est maintenant de quantifier de façon optimale une loi discrète $X : \Omega \to \mathbb{R}^d$. On note $x_1, ..., x_N$ le support de cette loi et $\delta_1, ..., \delta_N$ les poids associés, i.e. $\mathbb{P}(X = x_i) = \delta_i$. On cherche à approcher cette loi par une nouvelle loi discrète $Y : \Omega \to \mathbb{R}^d$ de support $y_1, ..., y_k$ et dont les poids associés sont $p_1, ..., p_k$. Le but est alors de minimiser la nouvelle fonction distorsion

$$D(y_1, ...y_k) = \sum_{i=1}^{N} \delta_i \left\| x_i - y(x_i) \right\|_2^2 \tag{3.6}$$

avec $y(x)$ toujours défini par (3.3), les p_i étant donnés par

$$p_i = \sum_{x_j \in \hat{\Gamma}(y_i)} \delta_j. \tag{3.7}$$

On suppose toujours $k < N$ et que les x_i sont disjoints deux à deux. Cette nouvelle problématique étant très proche de celle du k-mean, il semble naturel de modifier l'algorithme du k-mean pour la résoudre. En fait, celui-ci est quasiment identique, à la différence près qu'au lieu de calculer les barycentres simples des pavés de Voronoï, on calcule les barycentres pondérés. Ainsi, on pose

$$\begin{array}{rccc} \tilde{\Phi}_{\hat{\Gamma}} : & \mathbb{R}^{d\times k} & \to & \mathbb{R}^{d\times k} \\ & (y_1, \dots, y_k) & \mapsto & (\tilde{\phi}^1_{\hat{\Gamma}}(y_1, \dots, y_k), \dots, \tilde{\phi}^k_{\hat{\Gamma}}(y_1, \dots, y_k)) \end{array} \tag{3.8}$$

$$\tilde{\phi}^i_{\hat{\Gamma}}(y_1, \dots, y_k) = \begin{cases} y_i & \text{si } \left|\hat{\Gamma}(y_i)\right| = 0 \\ \frac{\sum_{x_j \in \hat{\Gamma}(y_i)} \delta_j x_j}{\sum_{x_j \in \hat{\Gamma}(y_i)} \delta_j} & \text{sinon} \end{cases}$$

On utilise alors le même algorithme que précédemment mais en utilisant la fonction $\tilde{\Phi}$:

1. On initialise l'algorithme en tirant les y_i^0 deux à deux distincts parmi les $\{x_1, \dots, x_N\}$.
2. On itère $(y_1^{n+1}, \dots, y_k^{n+1}) = \tilde{\Phi}_{\hat{\Gamma}}(y_1^n, \dots, y_k^n)$ tant que l'on n'a pas atteint un point fixe.
3. Si ce point fixe possède un unique pavage de Voronoï, l'algorithme est fini. Sinon, on détermine deux points y_i^n et y_{i+j}^n tels qu'il existe un $x \in \{x_1, \dots, x_N\}$ qui soit équidistant de ces deux points. On considère alors le pavage de Voronoï suivant

   $$\Gamma(y_i^n) = \hat{\Gamma}(y_i^n) \backslash \{x\}, \quad \Gamma(y_{i+j}^n) = \hat{\Gamma}(y_{i+j}^n) \cup \{x\}, \quad \Gamma(y_k^n) = \hat{\Gamma}(y_k^n)\, k \neq i, i+j$$

   et on pose $(y_1^{n+1}, \dots, y_k^{n+1}) = \tilde{\Phi}_{\Gamma}(y_1^n, \dots, y_k^n)$. On retourne ensuite à l'étape 2.

De plus, les poids des y_i sont définis ainsi :

$$p_i^n = \sum_{x_j \in \hat{\Gamma}(y_i^n)} \delta_j. \tag{3.9}$$

3.4.2 Convergence du nouvel algorithme

Tous les résultats démontrés dans la partie 3.2.3 restent valables, à la différence près qu'il convient de considérer des barycentres pondérés et non plus des barycentres « simples ». Ces résultats nous permettent d'avoir un équivalent du théorème 3.2.12 pour le k-mean pondéré.

Théorème 3.4.1 *L'algorithme du k-mean pondéré se termine et le point fixe renvoyé est un minimum local de D.*

3.5 Quelques résultats numériques

Pour tester l'algorithme en pratique, nous avons tout d'abord testé une réduction du nombre de quantificateurs de quantifications optimales gaussiennes en dimension 1 et 2 (figures 3.7 et 3.9). Les figures 3.8 et 3.10 permettent d'apprécier l'évolution des distorsions en fonction du nombre d'itérations de l'algorithme. Ensuite, afin de se retrouver dans un cas plus proche de la quantification de chaîne de Markov, nous avons testé l'algorithme sur la quantification d'une gaussienne $\mathcal{N}(0, \sqrt{2})$ obtenue par un schéma d'Euler à deux pas $(X_i)_{0 \leq i \leq 2}$ avec $X_{i+1} = X_i + \mathcal{N}(0, 1)$. Pour le premier pas, 500 quantificateurs optimaux de $\mathcal{N}(0, 1)$ sont utilisés et 10 pour le second pas. Les 5000 quantificateurs ainsi obtenus sont représentés sur la figure 3.11. Les figures 3.12 et 3.13 montrent les résultats obtenus lorsque le nombre de quantificateurs passe de 5000 à 500 pour 100 et 1000 itérations. De plus, la figure 3.14 représente l'évolution des distorsions en fonction du nombre d'itérations de l'algorithme. Enfin, le tableau suivant compare les distorsions obtenues à l'itération 991 avec les distorsions des quantifications optimales.

Nombre de quantificateurs	Distorsion optimale	Distorsion expérimentale	Distorsion du k-mean
40	$1,6187.10^{-3}$	$1,6961.10^{-3}$	$1,6574.10^{-3}$
60	$5.4231.10^{-2}$	$6,6824.10^{-2}$	$5,9480.10^{-2}$
500	$7.44105.10^{-3}$	$9,8351.10^{-3}$	$6,2571.10^{-3}$

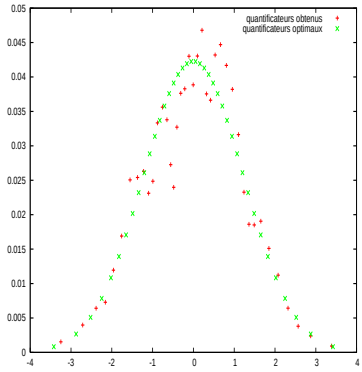

FIG. 3.7 – Passage de 400 à 40 quantificateurs en dimension 1, comparaison avec les 40 quantificateurs optimaux.

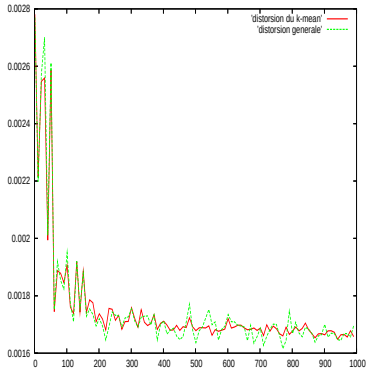

FIG. 3.8 – Distorsion du k-mean (trait plein) et distorsion générale (trait pointillé) en fonction du nombre d'itérations de l'algorithme de k-mean pour une réduction de 400 à 40 quantificateurs en dimension 1.

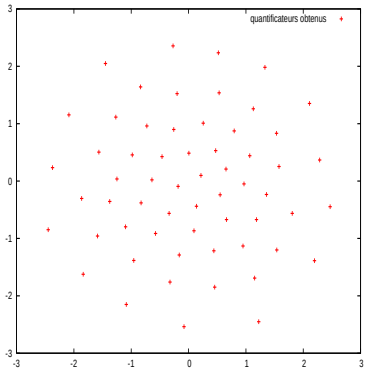

FIG. 3.9 – Passage de 600 à 60 quantificateurs en dimension 2.

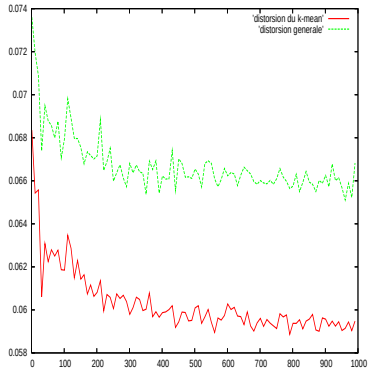

FIG. 3.10 – Distorsion du k-mean (trait plein) et distorsion générale (trait pointillé) en fonction du nombre d'itérations de l'algorithme de k-mean pour une réduction de 600 à 60 quantificateurs en dimension 2.

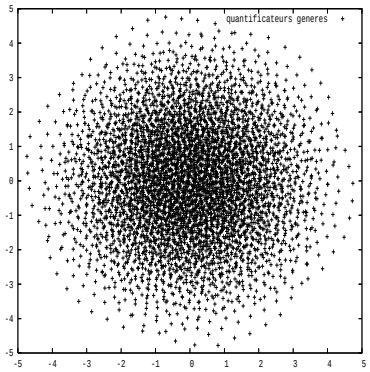

FIG. 3.11 – 5000 quantificateurs browniens générés par 500 quantificateurs puis 10 quantificateurs.

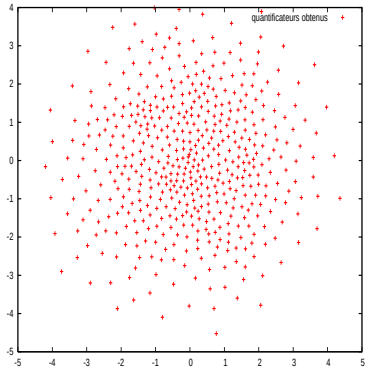

FIG. 3.12 – Passage de 5000 à 500 quantificateurs en dimension 2 avec 100 itérations.

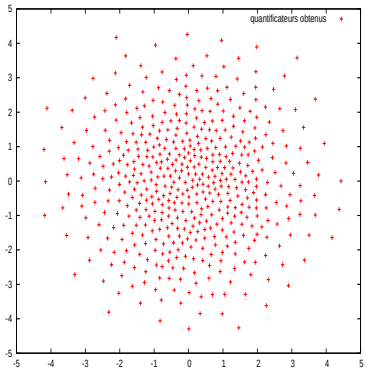

FIG. 3.13 – Passage de 5000 à 500 quantificateurs en dimension 2 avec 1000 itérations.

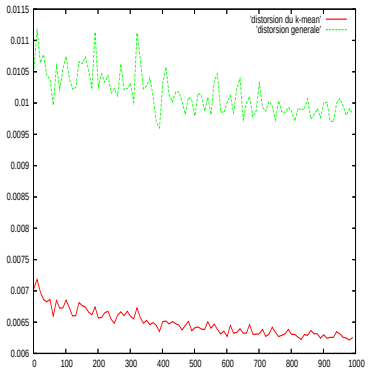

FIG. 3.14 – Distorsion du k-mean (trait plein) et distorsion générale (trait continu) en fonction du nombre d'itérations de l'algorithme de k-mean pour une réduction de 5000 à 500 quantificateurs en dimension 2.

Chapitre 4

Convergence théorique du schéma

Nous allons, dans ce chapitre, détailler le schéma de discrétisation d'EDSR envisagé dans la partie 2.2.5 afin d'étudier théoriquement sa vitesse de convergence (théorèmes 4.4.2 et 4.4.3). Rappelons que les équations (E) et $(\mathcal{E})$ ainsi que les hypothèses $(\mathcal{H}_1)$ et $(\mathcal{H}_2)$ sont définies dans le premier chapitre.

4.1 relation entre les EDSR et les EDP

Toutes les notations et les résultats utiles ont été établis dans le premier chapitre. Rappelons tout de même les équations (E) et $(\mathcal{E})$ ainsi que les hypothèses $(\mathcal{H}_1)$ et $(\mathcal{H}_2)$ en ajoutant l'hypothèse que f est indépendante de z.

$$(E) \quad \begin{cases} X_t = x_0 + \int_0^t b(X_s)ds + \int_0^t \sigma(X_s)dB_s \\ Y_t = g(X_T) + \int_t^T f(s, X_s, Y_s)ds - \int_t^T Z_s dB_s \end{cases}$$

$$(\mathcal{E}) \quad \begin{cases} \partial_t u(t,x) + \langle b(x), \nabla_x u(t,x)\rangle + \frac{1}{2} tr((\sigma\sigma^*)(x)\nabla^2_{x,x} u(t,x)) + f(t,x,u(t,x)) = 0 \\ u(T,x) = g(x) \end{cases}$$

Hypothèses $(\mathcal{H}_1)$:

1. b, f, g et σ sont bornées en espace et ont une croissance au plus linéaire vis à vis des autres variables.
2. b, f, g et σ sont uniformément lipschitziennes vis à vis de toutes les variables.
3. $\sigma\sigma^*$ est uniformément elliptique.
4. g est bornée dans $C^2(\mathbb{R}^d)$.

Hypothèses $(\mathcal{H}_2)$ **:**

1. $b \in C^3(\mathbb{R}^d, \mathbb{R}^d)$, $\sigma \in C^3(\mathbb{R}^d, \mathbb{R}^{d\times d})$, $g \in C^3(\mathbb{R}^d)$.
2. Les dérivées partielles d'ordre inférieur ou égal à 3 de b et σ sont bornées.
3. Les dérivées partielles d'ordre inférieur ou égal à 3 de g sont à croissance au plus polynômiale.
4. $(x, y) \to f(s, x, y)$ est de classe C^3 quelque soit $s \in [0, T]$ et de plus
   - Pour tout $s \in [0, T]$, les dérivées partielles d'ordre inférieur ou égal à 3 de $f(s, ., 0)$ sont à croissance au plus polynômiale.
   - $\partial f/\partial y$ ainsi que ses dérivées partielles d'ordre 1 et 2 en x et y sont bornées sur $[0, T] \times \mathbb{R}^d \times \mathbb{R}$.

4.2 Discrétisation des processus

Dans toute la suite on suppose que les hypothèses $(\mathcal{H}_1)$ ou $(\mathcal{H}_2)$ sont vérifiées. On note $h = T/n$ avec n le nombre de pas de temps et Π_k la projection sur la k-ième grille notée $\mathcal{C}_k$. Nous considérerons deux cas, celui de la projection sur une grille fixe et celui de la projection à l'aide du k-mean. On note également $\Delta B^k = B_{t_{k+1}} - B_{t_k}$ et $\Delta\hat{B}^k$ une quantification optimale de ΔB^k ainsi que N le nombre de quantificateurs[1]. En pratique, les $\Delta\hat{B}^k$ sont générées à l'aide d'une quantification optimale de loi normale centrée réduite que l'on dilate de $\sqrt{h}$. Ainsi les $\Delta\hat{B}^k$ sont indépendantes et de même loi. On approche X par le processus discrétisé suivant :

$$\left\{\begin{array}{rcl} \hat{X}_{t_0} & = & x_0 \\ \hat{X}_{t_{k+1}} & = & \Pi_{k+1}(\hat{X}_{t_k} + \mathcal{T}(t_k, \hat{X}_{t_k})) \\ avec & & \mathcal{T}(t_k, x) = b(x)h + \sigma(x)\Delta\hat{B}^k \end{array}\right. \tag{4.1}$$

Le processus discret $(\hat{X}_{t_k})$ peut-être prolongé sur $[0, T]$ de cette façon :

$$\hat{X}_t = \hat{X}_{t_k} + b(\hat{X}_{t_k})(t - t_k) + \sigma(\hat{X}_{t_k})(B_t - B_{t_k}) \tag{4.2}$$

Enfin, on approche (Y) par le processus discrétisé suivant :

$$\hat{Y}_{t_k} = \bar{u}(t_k, \hat{X}_{t_k}) \tag{4.3}$$

avec

$$\left\{\begin{array}{rcl} \bar{u}(T, x) & = & g(x) \\ \bar{u}(t_k, x) & = & \mathbb{E}\left[\bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k, x))) \right. \\ & & \left. + f(t_k, x, \bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k, x))))h\right] \quad \forall x \in \mathcal{C}_k \end{array}\right. \tag{4.4}$$

Pour prouver la convergence du processus discrétisé vers le processus (Y), nous ferons appel dans la démonstration à un processus auxiliaire $(\bar{Y}_t)$.

$$\bar{Y}_t = u(t, \hat{X}_t) \tag{4.5}$$

[1] N est une constante, elle ne dépend pas de k.

avec u la solution de $(\mathcal{E})$.

On note $(\bar{Z}_t)$ la parties martingale de $(\bar{Y}_t)$. De plus, d'après le théorème de représentation martingale ([11] partie 3.4) il existe un processus progressivement mesurable et de carré intégrable $(\hat{Z}_t)$ tel que $\hat{Y}_{t_{k+1}} + hf(t_k, \hat{X}_{t_k}, \hat{Y}_{t_{k+1}}) = \hat{Y}_{t_k} + \int_{t_k}^{t_{k+1}} \hat{Z}_s dB_s$ pour tout k.

4.3 résultats préliminaires

Dans toute la suite on suppose les hypothèses suivantes vérifiées.

Hypothèses $(\mathcal{H}_3)$:

- Il existe une constante C_0 telle que $\delta \leq C_0 h$ lorsque les projections se font sur une grille de pas fixe δ.
- $\mathbb{E}[\hat{X}_{t_{k-1}} + \mathcal{T}(t_k, \hat{X}_{t_{k-1}}) - \hat{X}_{t_k} | \hat{X}_{t_k}] = 0$[2] pour tout $k \in \{1, \ldots, n\}$ lorsque les projections se font à l'aide du k-mean.
- Il existe une constante positive c et un entier q tels que $\forall (x, x') \in (\mathbb{R}^d)^2$, $\forall (t, t') \in [0, T]^2$, $\forall y \in \mathbb{R}$,[3]

$$|f(t, x, y) - f(t', x', y)| \leq c(1 + |x|^q + |x'|^q + |y|^q)(|t - t'| + |x - x'|).$$

De plus, nous noterons C les constantes réelles strictement positives ne dépendant que des paramètres découlant des hypothèses $(\mathcal{H}_1)$ ou $(\mathcal{H}_2)$. Lorsque celles-ci dépendent d'un paramètre supplémentaire, ce dernier est précisé en indice. Insistons sur le fait que cette notation est générique.

Lemme 4.3.1 *Soient S un vecteur aléatoire de $\mathbb{R}^d$, $\hat{S}$ une quantification de S telle que $\mathbb{E}[S - \hat{S}|\hat{S}] = 0$ et ϕ une fonction convexe de $\mathbb{R}^d$ dans $\mathbb{R}$. Alors on a*

$$\mathbb{E}\left[\phi(\hat{S})\right] \leq \mathbb{E}\left[\phi(S)\right]$$

Démonstration : Il suffit d'appliquer l'inégalité de Jensen :

$$\mathbb{E}\left[\phi(\hat{S})\right] = \mathbb{E}\left[\phi\left(\mathbb{E}[S|\hat{S}]\right)\right] \leq \mathbb{E}\left[\phi(S)\right]$$

□

Lemme 4.3.2 *Pour tout $p \in \mathbb{N}^*$ il existe une constante C_p telle que*

$$\sup_{0 \leq k \leq n} \mathbb{E}\left|\hat{X}_{t_k}\right|^{2p} \leq C_p$$

[2] Cette propriété signifie que la projection est réalisée sur une distribution discrète qui est un point fixe pour l'algorithme du k-mean.

[3] Cette hypothèse est utile uniquement lorsque l'on suppose les hypothèses $(\mathcal{H}_2)$ vérifiées.

Démonstration Majorons dans un premier temps la quantité $\mathbb{E}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p}$:

$$\begin{aligned}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^2 &= \left|\hat{X}_{t_k}+b(\hat{X}_{t_k})h\right|^2+\left|\sigma(\hat{X}_{t_k})\Delta\hat{B}^k\right|^2\\ &\quad +2\sqrt{h}\left\langle \hat{X}_{t_k}+b(\hat{X}_{t_k})h,\sigma(\hat{X}_{t_k})\frac{\Delta\hat{B}^k}{\sqrt{h}}\right\rangle\\ &= A_1+\sqrt{h}A_2\end{aligned}$$

Or, $\mathbb{E}[A_1^{p-1}A_2|\hat{X}_{t_k}]=0$. En effet, $\Delta\hat{B}^k$ est une quantification optimale indépendante de $\hat{X}_{t_k}$ ce qui implique que $\mathbb{E}[\Delta\hat{B}^k|\hat{X}_{t_k}]=\mathbb{E}[\Delta\hat{B}^k]=\mathbb{E}[\Delta B^k]=0$. Ainsi,

$$\mathbb{E}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p} = \mathbb{E}\left[A_1^p\right]+h\mathbb{E}\left[\sum_{i=2}^{p}\binom{p}{i}h^{i/2-1}A_1^{p-i}A_2^i\right]$$

b et σ sont uniformément lipschitziennes donc $|b(x)|\leq|b(0)|+L_b|x|$ et $|\sigma(x)|\leq|\sigma(0)|+L_\sigma|x|$. De plus, d'après le lemme 4.3.1 on a

$$\mathbb{E}\left[\left|\frac{\Delta\hat{B}^k}{\sqrt{h}}\right|^i\right]\leq\mathbb{E}\left[\left|\frac{\Delta B^k}{\sqrt{h}}\right|^i\right]=\alpha_i.$$

Vu que $h\leq T$, on trouve

$$\mathbb{E}\left[\sum_{i=2}^{p}\binom{p}{i}h^{i/2-1}A_1^{p-i}A_2^i\right]\leq C_p\left(1+\mathbb{E}\left[\left|\hat{X}_{t_k}\right|^{2p}\right]\right) \tag{4.6}$$

De plus,

$$A_1=\left|\hat{X}_{t_k}\right|^2+\underbrace{h^2\left|b(\hat{X}_{t_k})\right|^2+2h\langle\hat{X}_{t_k},b(\hat{X}_{t_k})\rangle+h\left|\sigma(\hat{X}_{t_k})\frac{\Delta\hat{B}^k}{\sqrt{h}}\right|^2}_{hB_1}$$

$$A_1^p=\left|\hat{X}_{t_k}\right|^{2p}+h\sum_{i=1}^{p}\binom{p}{i}h^{i-1}\left|\hat{X}_{t_k}\right|^{2p-2i}B_1^i$$

En utilisant les mêmes arguments que précédemment on a

$$\sum_{i=1}^{p}\binom{p}{i}h^{i-1}\left|\hat{X}_{t_k}\right|^{2p-2i}B_1^i\leq C_p\left(1+\mathbb{E}\left[\left|\hat{X}_{t_k}\right|^{2p}\right]\right)$$

Donc

$$\mathbb{E}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p}\leq C_ph+\mathbb{E}(1+C_ph)\left|\hat{X}_{t_k}\right|^{2p}$$

Avant de conclure la démonstration, il convient de considérer séparément les deux types de projection :

- Lorsque les projections se font à l'aide du k-mean, les hypothèses $(\mathcal{H}_3)$ ainsi que le lemme 4.3.1 nous assurent que

$$\begin{aligned}\mathbb{E}\left|\hat{X}_{t_{k+1}}\right|^{2p} &\leq \mathbb{E}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p}\\ &\leq C_p h+(1+C_p h)\mathbb{E}\left|\hat{X}_{t_k}\right|^{2p}\end{aligned}$$

- Lorsque les projections se font sur une grille fixe de pas δ, on a

$$\begin{aligned}\left|\hat{X}_{t_{k+1}}\right| &\leq \left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p}\\ &\quad+\sum_{i=1}^{2p}\binom{2p}{i}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{i}\left|\hat{X}_{t_{k+1}}-\hat{X}_{t_k}-\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p-i}\\ &\leq \left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{2p}+\sum_{i=1}^{2p}\binom{2p}{i}\left|\hat{X}_{t_k}+\mathcal{T}(t_k,\hat{X}_{t_k})\right|^{i}\left|\frac{\sqrt{d}}{2}\delta\right|^{2p-i}\end{aligned}$$

  Les hypothèses $(\mathcal{H}_3)$ nous assurent que $\left|\frac{\sqrt{d}}{2}\delta\right|^{2p-i}\leq C_p h$. En appliquant les mêmes majorations que précédemment il vient

$$\mathbb{E}\left|\hat{X}_{t_{k+1}}\right|^{2p}\leq C_p h+(1+C_p h)\mathbb{E}\left|\hat{X}_{t_k}\right|^{2p}$$

Pour conclure, on montre par récurrence sur k que

$$\begin{aligned}\mathbb{E}\left|\hat{X}_{t_k}\right|^{2p} &\leq \sum_{i=0}^{k-1}C_p h(1+C_p h)^i+|x_0|^{2p}(1+C_p)^k\\ &\leq C_p(1+|x_0|^{2p})\end{aligned}$$

□

Lemme 4.3.3 *Pour tout $p\in\mathbb{N}^*$ il existe une constante C_p telle que*

$$\mathbb{E}\left[\left|\hat{X}_t-\hat{X}_{t_k}\right|^{2p}\right]\leq C_p h^p\quad\forall t\in[t_k,t_{k+1}[$$

Démonstration : On a

$$\begin{aligned}\left|\hat{X}_t-\hat{X}_{t_k}\right|^{2p} &\leq \left|2(t-t_k)^2\left|b(\hat{X}_{t_k})\right|^2+2\left|\sigma(\hat{X}_{t_k})(B_t-B_{t_k})\right|^2\right|^p\\ &\leq 2^{2p-1}\left((t-t_k)^{2p}\left|b(\hat{X}_{t_k})\right|^{2p}+\left|\sigma(\hat{X}_{t_k})(B_t-B_{t_k})\right|^{2p}\right)\end{aligned}$$

En utilisant la lipschitzité de b et σ, et en appliquant le lemme 4.3.2, on obtient facilement le résultat. □

Lemme 4.3.4 *Pour tout $p\in\mathbb{N}^*$ il existe une constante C_p telle que*

$$\sup_{0\leq t\leq T}\mathbb{E}\left|\hat{X}_t\right|^{2p}\leq C_p$$

Démonstration : Il suffit d'appliquer les lemmes 4.3.2 et 4.3.3 en remarquant que

$$\hat{X}_t = (\hat{X}_{t_k}) + (\hat{X}_t - \hat{X}_{t_k})$$

et en utilisant les mêmes majorations que précédemment. □

Lemme 4.3.5 *Pour tout $p \in \mathbb{N}^*$ il existe une constante C_p telle que*

$$\mathbb{E}\left[\left|\hat{X}_{t_k} - \hat{X}_{t_{k-1}}\right|^{2p}\right] \leq C_p(h^p + \varepsilon_{2p,k}) \quad \forall 1 \leq k \leq n$$

avec

- $\varepsilon_{2p,k} = \delta^{2p}$ *lorsque la projection se fait sur une grille fixe de pas δ.*
- $\varepsilon_{2p,k} = \mathbb{E}\left[\left|\hat{X}_{t_k} - (\hat{X}_{t_{k-1}} + \mathcal{T}(t_{k-1}, \hat{X}_{t_{k-1}}))\right|^{2p}\right]$ *lorsque la projection se fait à l'aide du k-mean*[4]*.*

Démonstration : Il suffit d'appliquer le lemme 4.3.3 en remarquant que

$$\hat{X}_{t_k} - \hat{X}_{t_{k-1}} = (\hat{X}_{t_k} - \hat{X}_{t_{k-1}} - \mathcal{T}(t_{k-1}, \hat{X}_{t_{k-1}})) + (\mathcal{T}(t_{k-1}, \hat{X}_{t_{k-1}}))$$

et en utilisant les mêmes majorations que précédemment. De plus, lorsque la projection se fait sur une grille fixe de pas δ, On peut majorer le premier terme par $\frac{\sqrt{d}}{2}\delta$ ce qui nous assure le résultat. □

Lemme 4.3.6 *Soient S un vecteur aléatoire de $\mathbb{R}^d$ et $\hat{S}$ une quantification de S telle que $\mathbb{E}[S - \hat{S}|\hat{S}] = 0$*[5]*. Soit $\phi : \mathbb{R}^d \to \mathbb{R}$ une fonction différentiable telle que*

$$|d\phi(x) - d\phi(y)| \leq g(x,y)\,|x - y| \quad \forall (x,y) \in \mathbb{R}^d \times \mathbb{R}^d$$

avec g une fonction continue de $\mathbb{R}^d \times \mathbb{R}^d$ dans $\mathbb{R}^+$; Alors il existe $\Theta : \Omega \to [0,1]$ telle que

$$\left|\mathbb{E}[\phi(S)] - \mathbb{E}[\phi(\hat{S})]\right| \leq \mathbb{E}\left[g(\hat{S}, \Theta S + (1-\Theta)\hat{S}))\left|S - \hat{S}\right|^2\right]$$

Démonstration : Les formules de Taylor nous assurent que quelque soit $(x,y) \in \mathbb{R}^d \times \mathbb{R}^d$, il existe $\theta \in [0,1]$ telle que

$$\phi(x) - \phi(y) = d\phi(\theta x + (1-\theta)y).(x - y)$$

Ainsi, il existe $\Theta : \Omega \to [0,1]$ vérifiant

$$\begin{aligned}\left|\phi(S) - \phi(\hat{S}) - d\phi(\hat{S})(S - \hat{S})\right| &\leq g(\hat{S}, \Theta S + (1-\Theta)\hat{S}))\left|S - \hat{S}\right|^2 \\ \left|\mathbb{E}[\phi(S)] - \mathbb{E}[\phi(\hat{S})] - \mathbb{E}[d\phi(\hat{S})(S - \hat{S})]\right| &\leq \mathbb{E}\left[g(\hat{S}, \Theta S + (1-\Theta)\hat{S}))\left|S - \hat{S}\right|^2\right].\end{aligned}$$

On conclut comme pour la proposition 2.1.2. □

[4] On trouve exactement la distorsion $2p$ de la kième projection

[5] Propriété vraie si $\hat{S}$ est une quantification optimale ou un point fixe du k-mean

4.4 Convergence du schéma

Afin de prouver la convergence du schéma, nous allons reprendre la trame présente dans [4] en l'adaptant. Notons que notre démonstration diffère néanmoins de celle des auteurs de [4] sur deux points. Tout d'abord, leur modèle est plus général car il suppose que b et σ dépendent de y et que f et b dépendent de z[6]. Par contre, ces mêmes auteurs considèrent uniquement les hypothèses $(\mathcal{H}_1)$ et la projection sur la grille fixe. Les résultats que nous avons obtenus sur la vitesse de convergence se trouvent à la fin du chapitre (théorèmes 4.4.2 et 4.4.3).

Le but de la démonstration est de comparer $\bar{u}(0,x_0)$ avec $u(0,x_0)$, c'est à dire $\hat{Y}_0$ avec $\bar{Y}_0$. Pour cela, nous allons comparer les $\hat{Y}_{t_k}$ avec $\bar{Y}_{t_k}$ pour ensuite utiliser le lemme de Gronwall discret.

Première étape : application de la formule d'Itô au processus $\bar{Y}$. Comme $u \in C^{1,2}([0,T],\mathbb{R}^d)$, on peut appliquer la formule d'Itô au processus $\bar{Y}$ entre t_k et t_{k+1} en utilisant le fait que u est solution de $(\mathcal{E})$. On a alors pour tout $i \in \{0,...,n-1\}$,

$$\begin{aligned}\bar{Y}_{t_{i+1}} - \bar{Y}_{t_i} &= \bar{Y}_{t_{i+1}} - \bar{Y}_{t_{i+1}^-} + \int_{t_i}^{t_{i+1}} [F(s,\hat{X}_s,\hat{X}_{t_i}) - F(s,\hat{X}_s,\hat{X}_s)]ds \\ &\quad - \int_{t_i}^{t_{i+1}} f(s,\hat{X}_s,\bar{Y}_s)ds + \int_{t_i}^{t_{i+1}} \bar{Z}_s dB_s\end{aligned}$$

avec

$$F(t,x,y) = \langle \nabla_x u(t,x), b(y)\rangle + \frac{1}{2} tr((\sigma\sigma^*)(y)\nabla^2_{x,x}u(t,x)).$$

Deuxième étape : expression de la différence entre u et $\bar{u}$ aux temps discrétisés. A l'aide de l'expression obtenue précédemment et la définition de $\hat{Y}$, on a

$$\begin{aligned}[\bar{Y}_{t_{i+1}} - \hat{Y}_{t_{i+1}}] - [\bar{Y}_{t_i} - \hat{Y}_{t_i}] &= \bar{Y}_{t_{i+1}} - \bar{Y}_{t_{i+1}^-} \\ &\quad + \int_{t_i}^{t_{i+1}} [F(s,\hat{X}_s,\hat{X}_{t_i}) - F(s,\hat{X}_s,\hat{X}_s)]ds \\ &\quad - \int_{t_i}^{t_{i+1}} [f(s,\hat{X}_s,\bar{Y}_s) - f(t_i,\hat{X}_{t_i},\hat{Y}_{t_{i+1}})]ds \\ &\quad + \int_{t_i}^{t_{i+1}} [\bar{Z}_s - \hat{Z}_s]dB_s \\ &= \Delta E_{i+1}(1) + \Delta E_{i+1}(2) + \Delta E_{i+1}(3) + \Delta E_{i+1}(4)\end{aligned}$$

[6] On parle alors d'équation différentielle stochastique progressive retrograde (forward backward).

Le but est alors d'exprimer $(\bar{Y}_0 - \hat{Y}_0)$ en fonction des $\Delta E_{i+1}(1)$ et des $(\bar{Y}_{t_k} - \hat{Y}_{t_k})$. Pour cela, nous allons utiliser la formule d'Itô discrète tirée de [23] [7].

Proposition 4.4.1 (Formule d'Itô discrète) *Soit $S = (S_n)_{0 \leq n \leq N}$ une suite de variables aléatoires avec $S_0 = 0$. Soit $f : \mathbb{R} \to \mathbb{R}$ une fonction intégrable sur tout compact de $\mathbb{R}$ et F une primitive de f. Alors*

$$F(S_n) = F(S_0) + \sum_{k=1}^{n} f(S_{k-1})(S_k - S_{k-1}) + \frac{1}{2}\sum_{k=1}^{n}(f(S_k) - f(S_{k-1}))(S_k - S_{k-1}) + R_n(S, f(S))$$

avec

$$R_n(S, f(S)) = \sum_{k=1}^{n} \int_{S_{k-1}}^{S_k} \left[f(x) - \frac{f(S_{k-1}) + f(S_k)}{2} \right] dx.$$

On applique la formule d'Itô discrète avec $S_{i-1} = (\bar{Y}_i - \hat{Y}_i)$ et $f : x \mapsto x$ pour obtenir

$$\begin{aligned}(\bar{Y}_T - \hat{Y}_T)^2 &= (\bar{Y}_0 - \hat{Y}_0)^2 + \sum_{i=0}^{n}(\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}})(\Delta E_i(1) + ... + \Delta E_i(4)) \\ &\quad + \frac{1}{2}\sum_{i=1}^{n}(\Delta E_i(1) + ... + \Delta E_i(4))^2 + 0\end{aligned}$$

Comme $\mathbb{E}[(\bar{Y}_{t_{i-1}} - \hat{Y}_{t_{i-1}})\Delta E_i(4)] = 0$, on obtient finalement l'inégalité suivante

$$\left|\bar{Y}_0 - \hat{Y}_0\right|^2 \leq \mathbb{E}\left|\bar{Y}_T - \hat{Y}_T\right|^2 + \sum_{i=1}^{n}\left|\mathbb{E}\left[(\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}})(\Delta E_i(1) + \Delta E_i(2) + \Delta E_i(3))\right]\right| \tag{4.7}$$

Troisième étape : majoration de l'erreur Continuons la majoration de (4.7) en appliquant la formule de Young.

$$\begin{aligned}\left|\bar{Y}_0 - \hat{Y}_0\right|^2 &\leq \mathbb{E}\left|\bar{Y}_T - \hat{Y}_T\right|^2 \\ &\quad + \sum_{i=1}^{n}\left|\mathbb{E}\left[(\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}})(\Delta E_i(1))\right]\right| \\ &\quad + h\sum_{i=1}^{n}\mathbb{E}\left|\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}}\right|^2 \\ &\quad + \frac{1}{2h}\sum_{i=1}^{n}\mathbb{E}\left|\Delta E_i(2)\right|^2 \\ &\quad + \frac{1}{2h}\sum_{i=1}^{n}\mathbb{E}\left|\Delta E_i(3)\right|^2 \\ &\leq D_1 + D_2 + D_3 + D_4 + D_5\end{aligned}$$

[7] chapitre VII, Section 9 page 556

Tout d'abord, $\bar{Y}_T = \hat{Y}_T$ donc $D_1 = 0$. Ensuite, occupons nous de D_4.

$$\begin{aligned}
D_4 &= \frac{1}{2h}\sum_{i=1}^{n}\mathbb{E}\left|\int_{t_i-1}^{t_i}[F(s,\hat{X}_s,\hat{X}_{t_{i-1}}) - F(s,\hat{X}_s,\hat{X}_s)]ds\right|^2 \\
&\leq \frac{1}{2h}\sum_{i=1}^{n}\mathbb{E}\left[h\int_{t_i-1}^{t_i}\left|F(s,\hat{X}_s,\hat{X}_{t_{i-1}}) - F(s,\hat{X}_s,\hat{X}_s)\right|^2 ds\right] \\
&\leq C\sum_{i=1}^{n}\int_{t_i-1}^{t_i}\mathbb{E}\left[\left|\langle\nabla_x u(s,\hat{X}_s), b(\hat{X}_s - \hat{X}_{t_{i-1}})\rangle\right|^2\right. \\
&\quad \left.+\left|tr(((\sigma\sigma^*)(\hat{X}_s) - (\sigma\sigma^*)(\hat{X}_{t_{i-1}}))\nabla^2_{x,x}u(s,\hat{X}_s))\right|^2\right] ds
\end{aligned}$$

Sous les hypothèses $(\mathcal{H}_1)$, b et σ sont bornés ainsi que le gradient et la hessienne de u. La majoration ne pose donc pas de problèmes. Intéressons nous aux hypothèses $(\mathcal{H}_2)$. En utilisant le théorème 1.2.6, le lemme 4.3.3 et le lemme 4.3.4 on a

$$\begin{aligned}
\mathbb{E}\left[\left|\langle\nabla_x u(s,\hat{X}_s), b(\hat{X}_s - \hat{X}_{t_{i-1}})\rangle\right|^2\right] &\leq \mathbb{E}\left[\left|\nabla_x u(s,\hat{X}_s)\right|^2\left|b(\hat{X}_s) - b(\hat{X}_{t_{i-1}})\right|^2\right] \\
&\leq \left[\mathbb{E}\left|\nabla_x u(s,\hat{X}_s)\right|^4\right]^{1/2}\left[\mathbb{E}\left|b(\hat{X}_s) - b(\hat{X}_{t_{i-1}})\right|^4\right]^{1/2} \\
&\leq C\left[\mathbb{E}\left|1 + \left|\hat{X}_s\right|^{q'}\right|\right]^{1/2}\left[\mathbb{E}\left|\hat{X}_s - \hat{X}_{t_{i-1}}\right|^4\right]^{1/2} \\
&\leq Ch
\end{aligned}$$

En utilisant les mêmes arguments, et en s'assurant que $|\nabla_{x,x}u(t,.)|$ est à croissance au plus polynomiale, on montre que

$$\mathbb{E}\left[\left|tr(((\sigma\sigma^*)(\hat{X}_s) - (\sigma\sigma^*)(\hat{X}_{t_{i-1}}))\nabla^2_{x,x}u(s,\hat{X}_s))\right|^2\right] \leq Ch$$

On en conclut

$$D_4 \leq Ch$$

Regardons maintenant ce qui se passe pour D_5.

$$\begin{aligned}
D_5 &\leq C\sum_{i=1}^{n}\int_{t_{i-1}}^{t_i}\mathbb{E}\left[f(s,\hat{X}_s,\bar{Y}_s) - f(t_{i-1},\hat{X}_{t_{i-1}},\hat{Y}_{t_i})\right]^2 ds \\
&\leq C\sum_{i=1}^{n}\int_{t_{i-1}}^{t_i}\mathbb{E}\left[\left(1 + \left|\hat{X}_s\right|^q + \left|\hat{X}_{t_{i-1}}\right|^q + |\bar{Y}_s|^q\right)\left(|s - t_{i-1}|^2 + \left|\hat{X}_s - \hat{X}_{t_{i-1}}\right|^2\right)\right. \\
&\quad \left.+\left|\bar{Y}_s - \hat{Y}_{t_i}\right|^2\right] ds
\end{aligned}$$

On applique alors le résultat de croissance de u du théorème 1.2.6. Notons que lorsque nous utilisons les hypothèses $(\mathcal{H}_1)$, $q = 0$ et donc l'inégalité qui suit reste vraie.

$$\begin{aligned} D_5 &\leq C\sum_{i=1}^{n}\int_{t_{i-1}}^{t_i}\mathbb{E}\left[\left(1+\left|\hat{X}_s\right|^{q'}+\left|\hat{X}_{t_{i-1}}\right|^{q'}\right)\left(|s-t_{i-1}|^2+\left|\hat{X}_s-\hat{X}_{t_{i-1}}\right|^2\right)\right. \\ &\quad \left.+\left|\bar{Y}_s-\hat{Y}_{t_i}\right|^2\right]ds \\ &\leq C\sum_{i=1}^{n}\int_{t_{i-1}}^{t_i}\mathbb{E}\left[\left(1+\left|\hat{X}_s\right|^{q'}+\left|\hat{X}_{t_{i-1}}\right|^{q'}\right)\left(|s-t_{i-1}|^2+\left|\hat{X}_s-\hat{X}_{t_{i-1}}\right|^2\right)\right. \\ &\quad \left.+\left|u(s,\hat{X}_s)-u(t_i,\hat{X}_{t_{i-1}})\right|^2+\left|u(t_i,\hat{X}_{t_{i-1}})-u(t_i,\hat{X}_{t_i})\right|^2+\left|\bar{Y}_{t_i}-\hat{Y}_{t_i}\right|^2\right]ds \end{aligned}$$

On a, en appliquant le lemme 4.3.3, le lemme 4.3.4 et le théorème de Cauchy-Schwartz,

$$\mathbb{E}\left[\left(1+\left|\hat{X}_s\right|^{q'}+\left|\hat{X}_{t_{i-1}}\right|^{q'}\right)\left(|s-t_{i-1}|^2+\left|\hat{X}_s-\hat{X}_{t_{i-1}}\right|^2\right)\right]\leq Ch.$$

De plus, en appliquant le lemme 4.3.3, le lemme 4.3.4 et le théorème 1.2.5 ou 1.2.6[8], on obtient

$$\begin{aligned} \mathbb{E}\left|u(s,\hat{X}_s)-u(t_i,\hat{X}_{t_{i-1}})\right|^2 &\leq \mathbb{E}\left(1+\left|\hat{X}_s\right|^{q'}\right)\left(\left|\hat{X}_s-\hat{X}_{t_{i-1}}\right|^2+|s-t_i|\right) \\ &\leq Ch \end{aligned}$$

Enfin, pour l'avant-dernier terme nous utiliserons en plus le lemme 4.3.5

$$\begin{aligned} \mathbb{E}\left|u(t_i,\hat{X}_{t_{i-1}})-u(t_i,\hat{X}_{t_i})\right|^2 &\leq C\mathbb{E}\left(1+\left|\hat{X}_{t_i}\right|^{q'}\right)\left|\hat{X}_{t_i}-\hat{X}_{t_{i-1}}\right|^2 \\ &\leq C\sqrt{h^2+\varepsilon_{4,i}} \end{aligned}$$

Si $\sup_{1\leq i\leq n}\varepsilon_{4,i} < C_1h^2$ avec C_1 une constante arbitraire, on obtient finalement

$$\mathbb{E}\left|u(t_i,\hat{X}_{t_{i-1}})-u(t_i,\hat{X}_{t_i})\right|^2\leq Ch$$

Notons que pour les hypothèses $(\mathcal{H}_1)$, $\varepsilon_{4,i}$ est remplacé par $\varepsilon_{2,i}$ et l'inégalité supposée devient $\sup_{1\leq i\leq n}\varepsilon_{2,i} < C_1h$. Finalement,

$$D_5 \leq C\left(h+h\sum_{i=1}^{n}\mathbb{E}\left[\left|\bar{Y}_{t_i}-\hat{Y}_{t_i}\right|^2\right]\right)$$

[8]Tout dépend des hypothèses envisagées

Il reste à regarder le terme D_2. Notons que pour celui-ci nous n'avons pas fait de majoration large car nous voulons utiliser le lemme 4.3.6.

$$\begin{aligned}
D_2 &= \sum_{i=1}^{n} \left| \mathbb{E}\left[(\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}})(\bar{Y}_{t_i} - \bar{Y}_{t_i^-}) \right] \right| \\
&\leq \sum_{i=1}^{n} \left| \mathbb{E}\left[\left(\hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}} \right) \left(\bar{Y}_{t_i} - u(t_i, \hat{X}_{t_{i-1}} + \mathcal{T}(t_{i-1}, \hat{X}_{t_{i-1}})) \right.\right.\right. \\
&\quad \left.\left.\left. + \mathbb{E}[u(t_i, \hat{X}_{t_{i-1}} + \mathcal{T}(t_{i-1}, \hat{X}_{t_{i-1}})) - \bar{Y}_{t_i^-} | \mathcal{F}_{t_{i-1}}] \right) \right] \right| \\
&\leq \sum_{i=1}^{n} \mathbb{E}\left[\left| \hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}} \right| \left(\left| \bar{Y}_{t_i} - u(t_i, \hat{X}_{t_{i-1}} + \mathcal{T}(t_{i-1}, \hat{X}_{t_{i-1}})) \right| \right.\right. \\
&\quad \left.\left. + \left| \mathbb{E}[\phi_{i-1}(\Delta B^{i-1}) - \phi_{i-1}(\Delta \hat{B}^{i-1}) | \mathcal{F}_{t_{i-1}}] \right| \right) \right]
\end{aligned}$$

avec $\phi_{i-1}(x) = u(t_{i-1}, \hat{X}_{t_{i-1}} + b(\hat{X}_{t_{i-1}})h + \sigma(\hat{X}_{t_{i-1}})x)$. On applique le lemme 4.3.6 au dernier membre

$$\begin{aligned}
\left| \mathbb{E}[\phi_{i-1}(\Delta B^{i-1}) - \phi_{i-1}(\Delta \hat{B}^{i-1}) | \mathcal{F}_{t_{i-1}}] \right| &\leq C \left(1 + \left| \hat{X}_{t_{i-1}} \right|^{q'} \right) \mathbb{E}\left[\left(1 + \left| \Delta B^{i-1} \right|^{q'} + \left| \Delta \hat{B}^{i-1} \right|^{q'} \right) \right. \\
&\quad \left. \left| \Delta B^{i-1} - \Delta \hat{B}^{i-1} \right|^2 | \mathcal{F}_{t_{i-1}} \right] \\
&\leq C \left(1 + \left| \hat{X}_{t_{i-1}} \right|^{q'} \right) \left| \mathbb{E}\left[\left| \Delta B^{i-1} - \Delta \hat{B}^{i-1} \right|^4 | \mathcal{F}_{t_{i-1}} \right] \right|^{1/2} \\
&\leq Ch N^{-2/d} \left(1 + \left| \hat{X}_{t_{i-1}} \right|^{q'} \right)
\end{aligned}$$

La dernière inégalité découle du théorème de Zador (2.1.3). Pour le premier terme, on applique une n-ième fois les mêmes méthodes

$$\begin{aligned}
\sum_{i=1}^{n} \mathbb{E}\left[\left| \hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}} \right| \left| \bar{Y}_{t_i} - u(t_i, \hat{X}_{t_{i-1}} + \mathcal{T}(t_{i-1}, \hat{X}_{t_{i-1}})) \right| \right] &\leq Ch \sum_{i=1}^{n} \mathbb{E}\left| \hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}} \right|^2 \\
&\quad + \frac{C}{h} \sum_{i=1}^{n} \varepsilon_{4,i}(projection)^{1/2}
\end{aligned}$$

Remarquons une nouvelle fois que sous les hypothèses $(\mathcal{H}_1)$ $\varepsilon_{4,i}^{1/2}$ est remplacé par $\varepsilon_{2,i}$. On a finalement

$$D_2 \leq Ch \sum_{i=1}^{n} \mathbb{E}\left| \hat{Y}_{t_{i-1}} - \bar{Y}_{t_{i-1}} \right|^2 + \frac{C}{h} \sum_{i=1}^{n} \varepsilon_{4,i}^{1/2} + N^{-2/d}$$

En rassemblant tous les résultats, on obtient

$$\left| \bar{Y}_0 - \hat{Y}_0 \right|^2 \leq C \left(h + \frac{1}{h} \sum_{i=1}^{n} \varepsilon_{4,i}^{1/2} + N^{-2/d} + h \sum_{i=0}^{n} \mathbb{E}\left| \hat{Y}_{t_i} - \bar{Y}_{t_i} \right| \right)$$

$$\begin{aligned} |\bar{u}(0,x_0)-\hat{u}(0,x_0)|^2 &\leq C\Big(h+\tfrac{1}{h}\textstyle\sum_{i=1}^n \varepsilon_{4,i}^{1/2}+N^{-2/d} \\ &\quad +h\textstyle\sum_{i=0}^n \sup_{x\in\mathcal{C}_i}|\hat{u}(t_i,x)-\bar{u}(t_i,x)|^2\Big) \end{aligned} \tag{4.8}$$

Quatrième étape : lemme de Gronwall L'inégalité (4.8) reste valable pour n'importe quel point de départ (t_i, x) avec $0 \leq i \leq n$ et $x \in \mathcal{C}_i$. Donc, il existe C_2 et C_3 deux constantes telles que pour $h < C_2$ pour tout $k \in \{0, \ldots, n\}$ et pour tout $x \in \mathcal{C}_k$ on a

$$\begin{aligned} |\bar{u}(t_k,x)-\hat{u}(t_k,x)|^2 &\leq C_3\Big(h+\tfrac{1}{h}\textstyle\sum_{i=k+1}^n \varepsilon_{4,i}^{1/2}+N^{-2/d} \\ &\quad +h\textstyle\sum_{i=k+1}^n \sup_{x'\in\mathcal{C}_i}|\hat{u}(t_i,x')-\bar{u}(t_i,x')|^2\Big) \end{aligned} \tag{4.9}$$

On peut alors appliquer le lemme de Gronwall discret à l'inégalité (4.9) pour obtenir les deux résultats qui suivent sur la convergence du schéma.

Théorème 4.4.2 *On suppose les hypothèses $(\mathcal{H}_1)$ et $(\mathcal{H}_3)$ vérifiées. On suppose également qu'il existe une constante positive C_1 telle que $\sup_{1\leq i\leq n}\varepsilon_{2,i} < C_1 h$. Alors il existe deux constantes positives C_2 et C_3 telles que, pour $h < C_2$ on a*

$$|\bar{u}(0,x_0)-\hat{u}(0,x_0)|^2 \leq C_3\left(h+\frac{1}{h}\sum_{i=1}^n \varepsilon_{2,i}+N^{-2/d}\right)$$

Théorème 4.4.3 *On suppose les hypothèses $(\mathcal{H}_2)$ et $(\mathcal{H}_3)$ vérifiées. On suppose également qu'il existe une constante positive C_1 telle que $\sup_{1\leq i\leq n}\varepsilon_{4,i} < C_1 h^2$. Alors il existe deux constantes positives C_2 et C_3 telles que, pour $h < C_2$ on a*

$$|\bar{u}(0,x_0)-\hat{u}(0,x_0)|^2 \leq C_3\left(h+\frac{1}{h}\sum_{i=1}^n \varepsilon_{4,i}^{1/2}+N^{-2/d}\right)$$

Remarques :

1. Lorsque les projections se font sur une grille fixe, les conditions $\sup_{1\leq i\leq n}\varepsilon_{2,i} < C_1 h$ et $\sup_{1\leq i\leq n}\varepsilon_{4,i} < C_1 h^2$ sont toutes les deux identiques : $\delta^2 < C_1 h$. S'il existe une constante positive C_δ telle que $\delta \leq C_\delta$, alors il suffit juste de vérifier les hypothèses $(\mathcal{H}_3)$. En effet, l'existence d'une constante positive C_1 telle que $\delta \leq C_1 h$ entraîne que $\delta^2 \leq C_q C_1 h$.
2. Il est possible de prendre un autre schéma en modifiant (4.4) de la façon suivante :

   $$\begin{aligned} \bar{u}(t_k,x) &= \mathbb{E}\left[\bar{u}(t_{k+1},\Pi_{k+1}(x+\mathcal{T}(t_k,x))\right. \\ &\quad \left.+f(t_k,\Pi_{k+1}(x+\mathcal{T}(t_k,x)),\bar{u}(t_{k+1},\Pi_{k+1}(x+\mathcal{T}(t_k,x))))h\right] \quad \forall x\in\mathcal{C}_k \end{aligned}$$

   Dans ce cas, les majorations sont plus compliquées. En effet, dans la majoration de D_5 le terme $\left|\hat{X}_s-\hat{X}_{t_{i-1}}\right|^2$ est remplacé par $\left|\hat{X}_s-\hat{X}_{t_i}\right|^2$ ce qui ajoute des termes contraignants dans le cas des projections à l'aide

du k-mean[9]. Ainsi ce schéma ne sera pas étudié théoriquement même si, numériquement, il fournit des résultats semblables.

3. En pratique, la projection sur une grille fixe n'est pas viable car la taille des grilles explose lorsque h est trop petit. les auteurs de [4] considèrent une grille fixe tronquée. Si l'on note R le rayon de troncature, alors ces derniers ont montré qu'un terme supplémentaire en $1/R$ apparaissait dans la majoration de l'erreur. Dans notre cas, il est très certainement possible d'appliquer le même type de démonstration pour ajouter un terme en $1/R$ dans les théorèmes 4.4.2 et 4.4.3.

[9] cf les lemmes 4.3.3 et 4.3.5

Chapitre 5

Résultats numériques

Rappel des notations :

- n : nombre de pas de temps.
- N : nombre de quantificateurs du brownien.
- d : dimension du processus forward.
- q : nombre total minimal de quantificateurs du processus forward.
- q_i : nombre de quantificateurs effectif du processus forward au temps t_i.
- δ le pas de la grille fixe.
- R le rayon de troncature dans le cas de la projection sur une grille fixe.

5.1 Complexité des algorithmes

L'étude de la complexité des deux algorithmes détaillés dans le chapitre précédent peut nous permettre d'avoir un premier élément de comparaison. Bien entendu, nous ne prétendons pas réaliser une étude fine de la complexité, étude qui nécessiterait de définir une ou des opération(s) élémentaire(s)[1] puis de les dénombrer. Nous nous contenterons juste de considérer le nombre de boucles. Pour faciliter l'étude, nous découperons les algorithmes en deux phases : la phase de discrétisation temporelle et spatiale du processus forward X et la phase de calcul du processus backward Y. Enfin nous évoquerons rapidement la complexité spatiale.

5.1.1 Complexité de l'algorithme utilisant le k-mean

- La première phase consiste à générer, à chaque temps t_i, $q_i N$ vecteurs puis à leur appliquer l'algorithme du k-mean pour en créer q_{i+1}. Si l'on note $kmean(q_i, q_{i+1})$ la complexité de l'algorithme du k-mean créant q_{i+1}

[1] Nous pourrions considérer les opérations arithmétiques par exemple.

vecteurs à partir de $q_i N$ vecteurs, alors nous avons une complexité en

$$O(\sum_{i=1}^{n}(q_{i-1}N + kmean(q_{i-1}, q_i)))$$

Nous avons vu dans le chapitre consacré au k-mean que

$$kmean(q_i, q_{i+1}) = O\left((n_{iter}q_{i-1} + Nq_i)\log(q_i N)\right)$$

avec n_{iter} le nombre d'itérations de l'algorithme de k-mean. Une question subsiste alors : comment allons nous fixer les q_i sachant que l'on souhaite avoir une quantité totale de quantificateurs à peu près égale à q ? Dans le cas de la quantification marginale optimale de chaîne de Markov, les auteurs de [1] proposent la répartition suivante :

$$q_i = \left\lceil \frac{t_i^{\frac{d}{2(d+1)}}(q-1)}{t_1^{\frac{d}{2(d+1)}} + \ldots + t_n^{\frac{d}{2(d+1)}}} \right\rceil, \quad 1 \leq i \leq n. \tag{5.1}$$

Ce choix s'appuie sur certains arguments d'optimalité qui ne sont pas transposables à notre schéma. Cette question sera étudiée de nouveau dans la partie 5.3.2 mais pour le moment nous nous contenterons de cette répartition. Notons au passage que

$$q \leq \sum_{i=0}^{n} q_i \leq q + n.$$

En tenant compte de (5.1), on trouve finalement, après quelques calculs, une complexité en

$$O\left((n_{iter} + N)q\log(q)\right)$$

en supposant $n \ll q$ et $N \ll q$, ce qui sera toujours vérifié en pratique.

- La seconde phase consiste, à chaque temps t_i, à parcourir les q_i quantificateurs et, pour chaque quantificateur, l'ensemble des transitions vers l'instant suivant. A première vue, cela représente $O(q_i q_{i+1})$ opérations. Cependant, on aura souvent très en pratique $N \ll q_i$, la matrice de transition est donc une matrice creuse. Il est alors possible de se ramener à $O(q_i N)$ opérations. On obtient ainsi une complexité en

$$O(qN).$$

- En ce qui concerne la place mémoire nécessaire, l'algorithme a besoin de stocker l'ensemble du processus forward discrétisé en temps et en espace et l'ensemble des transitions entre deux instants consécutifs. Comme nous l'avons déjà évoqué, il n'est pas nécessaire de stocker $q_i q_{i+1}$ réels pour chaque instant t_i, mais on peut se contenter de $q_i N$ entiers. Cela nous donne finalement une complexité spatiale de
  - $(d+1)\sum_{i=0}^{n} q_i$ flottants,

– $N \sum_{i=0}^{n-1} q_i$ entiers.

Pour compléter ces considérations théoriques, il peut être intéressant d'utiliser des outils permettant de profiler le programme, c'est à dire de déterminer comment se réparti le temps d'exécution entre les différentes parties du programme. Pour cela, nous avons utilisé l'outil *gprof* qui a le mérite d'être extrêmement simple d'utilisation. Tous les cas testés montrent que la majeure partie du temps d'exécution est utilisée par l'algorithme du k-mean et plus précisément la phase de recherche du plus proche voisin.

5.1.2 Complexité de l'algorithme utilisant une grille fixe

- La première phase consiste à générer, à chaque temps t_i, $q_i N$ vecteurs. La projection sur la grille fixe se fait sans surcoût car, dans notre cas, la recherche du plus proche voisin est triviale. Ainsi, nous obtenons une complexité en

$$O(\sum_{i=1}^{n} q_i).$$

  Remarquons que, contrairement à l'algorithme précédent, nous n'avons pas de contrôle direct sur les q_i. Nous savons juste qu'ils sont bornés par $(\frac{2R}{\delta})^d$.
- La seconde phase est, quant à elle, identique à celle déjà décrite dans l'algorithme précédent. Cela nous donne une complexité en

$$O(\sum_{i=1}^{n} q_i N).$$

- Enfin, la complexité spatiale est identique,
  – $(d+1) \sum_{i=0}^{n} q_i$ flottants,
  – $N \sum_{i=0}^{n-1} q_i$ entiers.

Cette fois, l'utilisation d'un profiler montre que le temps d'exécution est à peu près équiréparti entre les deux phases. De plus, il est bon de noter que le nombre de données à stocker sur le disque est bien plus important que dans le cas précédent. En prenant, par exemple, $n = 100$ et $\delta = 0.01$ en dimension 2, nous avons eu sur le disque plus de 2 Go de données. Ce phénomène provient du fait que l'on a pas de contrôle fort sur le nombre de quantificateurs à chaque étape. En pratique, les q_i évoluent de façon explosive avec la dimension comme le laisse suggérer la borne supérieure $(\frac{2R}{\delta})^d$.

5.2 Premiers tests

5.2.1 Cas-tests envisagés

On prendra pour tous les cas $T = 1$.

Cas-test 1

$$u_1(t,x) = sin(2x+t)$$

avec

$$\begin{cases} b_1(x) &= -0.5 \\ \sigma_1(x) &= 1 \\ f_1(t,x,u) &= 2u \end{cases}$$

Ce cas-test vérifie les hypothèses $(\mathcal{H}_1)$ et $(\mathcal{H}_2)$.

Cas-test 2

$$u_2(t,x) = 0.1sh(2(1-e^{-x^2})+2t)$$

avec

$$\begin{cases} b_2(x) &= th(x) \\ \sigma_2(x) &= 1 \\ f_2(t,x,u) &= -\left[2+4xe^{-x^2}th(x)+2(1-2x^2)e^{-x^2}\right]\sqrt{u^2+0.01} - 8x^2e^{-2x^2}u \end{cases}$$

Ce cas-test vérifie les hypothèses $(\mathcal{H}_2)$.

Cas-test 3

$$u_3(t,x) = 0.1sh(4th(10x)+t)$$

avec

$$\begin{cases} b_3(x) &= \frac{x}{10} \\ \sigma_3(x) &= \frac{1}{10} \\ f_3(t,x,u) &= -\left[1+4x(1-th^2(10x))-4th(10x)(1-th^2(10x))\right]\sqrt{u^2+0.01} \\ & \quad -8(1-th^2(10x))^2u \end{cases}$$

Ce cas-test vérifie les hypothèses $(\mathcal{H}_2)$.

Cas-test 4

$$u_4(t,x,y) = \sin(2x+y+t)$$

avec

$$\begin{cases} b_4(x) &= \frac{1}{3} \\ \sigma_4(x) &= 1 \\ f_4(t,x,u) &= 2.5u \end{cases}$$

Ce cas-test vérifie les hypothèses $(\mathcal{H}_1)$ et $(\mathcal{H}_2)$.

Cas-test 5

$$u_5(t,x,y) = 0.1sh(th(10x) + th(10y) + t)$$

avec

$$\begin{cases} b_5(x,y) &= \begin{pmatrix} x/10 \\ y/5 \end{pmatrix} \\ \sigma_5(x) &= \begin{pmatrix} 1/10 & 0 \\ 0 & 1/5 \end{pmatrix} \\ f_5(t,x,u) &= -\left[1 + x(1-th^2(10x)) + 2y(1-th^2(10y)) \right. \\ & \quad \left. -th(10x)(1-th^2(10x)) - 4th(10y)(1-th^2(10y))\right]\sqrt{u^2+0.01} \\ & \quad -\left[0.5(1-th^2(10x))^2 + 2(1-th^2(10y))^2\right]u \end{cases}$$

Ce cas-test vérifie les hypothèses $(\mathcal{H}_2)$.

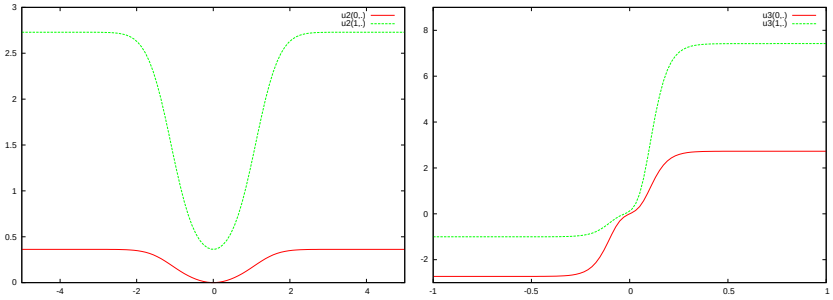

FIG. 5.1 – Cas-test 2 (gauche) et cas-test 3 (droite)

Remarques :

- Tous ces cas-tests ont été construits artificiellement, ils n'ont pas de relations avec des équations classiques de la physique. Les premiers fabriqués furent les cas-test 1 et 4. Ceux-ci ne nous ont pas paru pleinement satisfaisant car les fonctions f_1 et f_4 sont linéaires vis-à-vis de la variable u ce qui signifie que la formule de Feynman-Kac est applicable pour calculer ce que l'on souhaite : cela diminue grandement l'utilité de notre algorithme. Il convenait donc d'élargir notre batterie de tests en construisant des cas-tests admettant une fonction f non-linéaire en u. Les cas-tests 2, 3 et 5 sont tous construits à partir d'un même modèle :

  $$u(t,x) = \lambda sh(\phi(x) + \psi(t)).$$

  On remarque alors que

  $$\lambda ch(\phi(x) + \psi(t)) = \sqrt{u^2(t,x) + \lambda^2}.$$

  Nous nous sommes efforcés de prendre λ petit pour augmenter la non-linéarité de f vis-à-vis de u. Pour exploiter au maximum la non-linéarité de la fonction $u \mapsto \sqrt{u^2+\lambda^2}$, nous avons choisis pour u_3 une fonction qui change de signe avec une forte pente.

- Tous les tests numériques ont été effectués sur un PC *Pentium III* d'1 GHz avec 256 Mo de RAM. Les différents algorithmes ont été implantés en *C++* et compilés avec la version 3.4.4 de *gcc* en utilisant le niveau d'optimisation *-O3*.

5.2.2 Sensibilité aux paramètres

Sensibilité à n_{iter}

n_{iter} est le nombre d'itérations souhaité pour l'algorithme du k-mean. En jouant sur ce paramètre, le but est à la fois de réduire la distorsion produite par cet algorithme et de se rapprocher d'un point fixe à chaque pas de temps[2]. A première vue, le nombre d'itérations optimal va dépendre des paramètres pouvant influencer la vitesse de convergence de l'algorithme du k-mean, à savoir, le nombre moyen par pas de temps de quantificateurs pour le processus forward, le nombre N de quantificateurs pour le brownien et la dimension d du processus forward. En pratique, tous les tests numériques réalisés montrent qu'il n'en est rien : il suffit de choisir n_{iter} entre 10 et 20 pour avoir des résultats quasiment optimaux en un minimum de temps. Les figures 5.2, 5.3, 5.4 et 5.5 illustrent ce phénomène. A première vue, un nombre d'itération compris entre 10 et 20 semble faible. En guise de comparaison, les problèmes de classification non supervisée classiques nécessitent en moyenne une centaine d'itérations afin d'obtenir une solution satisfaisante. Ce phénomène peut s'expliquer par la nature des données : dans notre cas, toutes les données sont agrégées sur un seul amas et le rapport entre le nombre de données et le nombre de points à extraire est plus faible que dans les cas d'application habituels de l'algorithme. Ainsi, il y a une plus forte probabilité pour que les points tirés au hasard lors de l'initialisation de l'algorithme soient proches d'un minimum local. Le fait que nous puissions prendre un nombre d'itérations faible est une bonne nouvelle car il permet de réduire sensiblement les temps de calcul : La figure 5.6 illustre la dépendance linéaire de la complexité temporelle de l'algorithme vis-à-vis du nombre d'itérations.

[2] Ces deux critères interviennent dans les résultats théoriques sur la convergence du schéma.

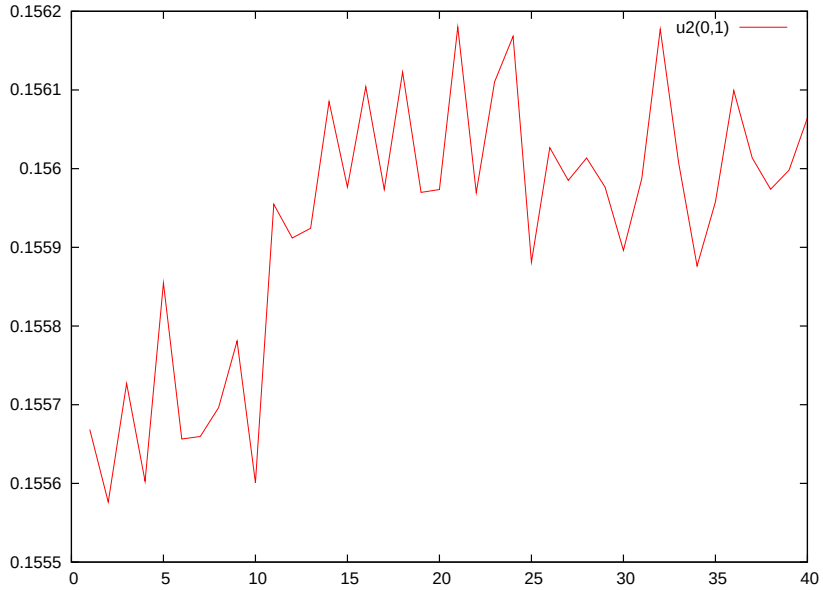

FIG. 5.2 – Résultats numériques obtenus en fonction du nombre d'itérations (Cas-test 2, $n = 100$, $N = 10$, $q = 30000$, $x_0 = 1$, $u(0, x_0) = 0.1629$).

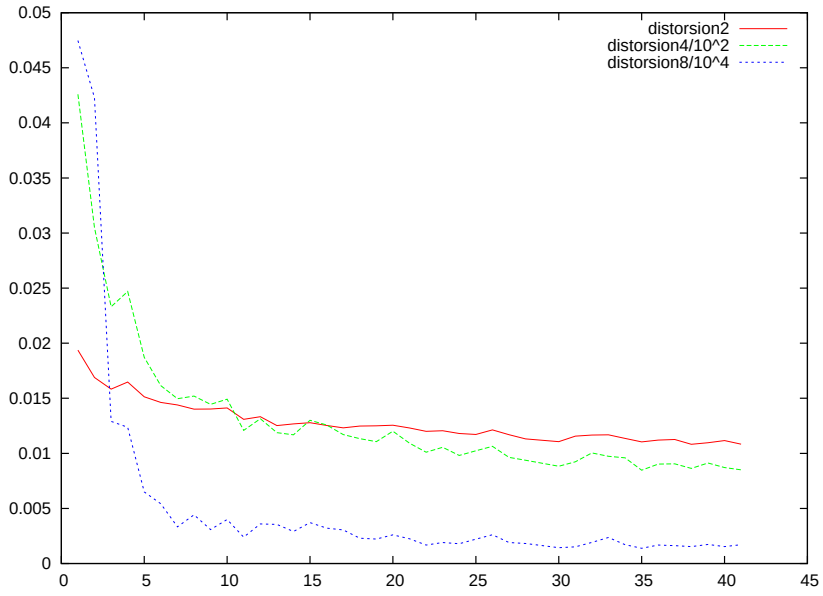

FIG. 5.3 – Sommes des distorsions 2, 4 et 8 du k-mean sur tous les pas de temps en fonction du nombre d'itérations (Cas-test 2, $n = 100$, $N = 10$, $q = 30000$).

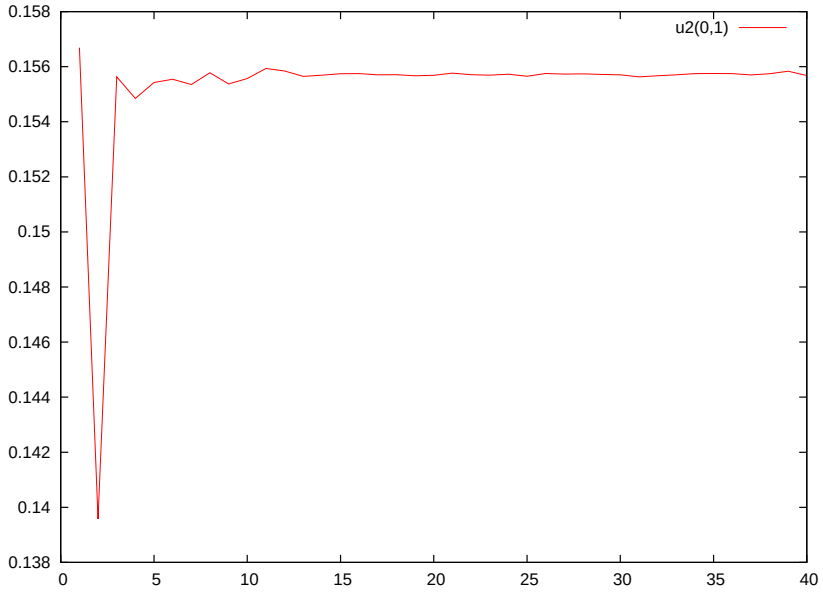

FIG. 5.4 – Résultats numériques obtenus en fonction du nombre d'itérations (Cas-test 2, $n = 1000$, $N = 100$, $q = 200000$, $x_0 = 1$, $u(0, x_0) = 0.1629$).

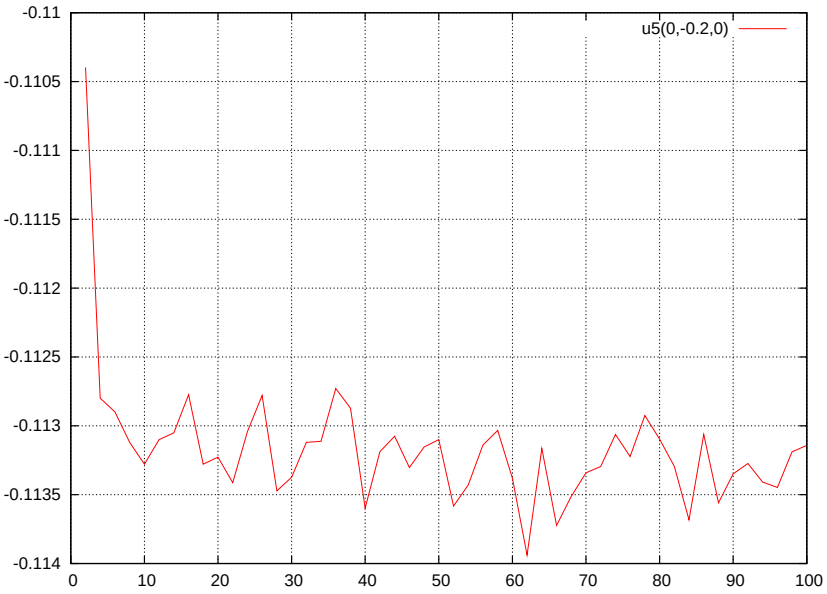

FIG. 5.5 – Résultats numériques obtenus en fonction du nombre d'itérations (Cas-test 5, $n = 100$, $N = 40$, $q = 20000$, $(x_0, y_0) = (-0.2, 0)$, $u(0, x_0, y_0) = -0.1120$).

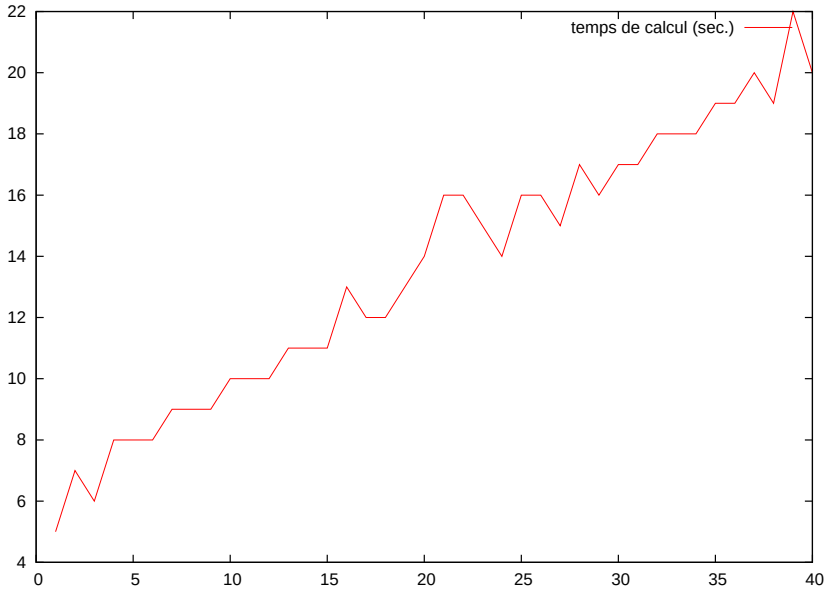

FIG. 5.6 – Temps de calcul (secondes) en fonction du nombre d'itérations (Castest 2, $n = 100$, $N = 10$, $q = 30000$, $x_0 = 1$, $u(0, x_0) = 0.1629$).

sensibilité à N

Si nous nous référons aux résultats théoriques établis sur la convergence du schéma, il conviendrait de prendre N de l'ordre de $n^{d/2}$. Les résultats des figures 5.7 et 5.8 semblent valider partiellement cette règle en dimension 1. Le fait de passer de $n = 100$ à $n = 1000$ nécessite effectivement un changement d'ordre de grandeur pour le choix de N : $N = 10$ convient pour le premier cas tandis qu'il s'agit de prendre N au moins égal à 60 pour le second cas. Par contre, les résultats de la figure 5.9 minimisent l'importance de la dimension dans le choix de N : En effet, avec $n = 100$, $N = 30$ semble suffire pour le cas étudié.

En regardant de plus près le schéma, il semble que le choix de q risque également d'être corrélé avec le choix de N. En effet, si q est trop « petit »[3], alors il faudra peut-être plus de quantificateurs pour le brownien afin d'améliorer substantiellement le calcul des transitions entre chaque instant. Ainsi le nombre de quantificateurs N optimal pourrait varier en fonction de q. Les résultats de la figure 5.10 semblent pourtant témoigner du contraire : Si nous laissons de côté les problèmes de convergence qui apparaissent lorsque q est beaucoup trop faible ($q \leq 4000$), le N optimal est indépendant de q. Il est également intéressant de noter que, pour tous les tests numériques réalisés, les distorsions totales dues au k-mean ne varient pas significativement avec N. Par conséquent, il est possible de fixer q et N de façon indépendante.

Enfin, la figure 5.11 illustre la dépendance linéaire de la complexité temporelle de l'algorithme vis-à-vis de N. Le fait que la pente de cette courbe soit forte confirme l'importance de choisir N de façon optimale afin de réduire substantiellement le temps de calcul. Notons également que cette pente est bien plus importante que celle de la dépendance vis-à-vis du nombre d'itérations (figure 5.6). Or, rappelons que la complexité temporelle de l'algorithme du k-mean à l'instant t_i est en $O\left((n_{iter} q_{i-1} + N q_i) \log(q_i N)\right)$, le terme en N étant lié à la construction de l'arbre de recherche et le terme en n_{iter} à l'algorithme même. Nous pouvons donc en déduire que la phase de construction de l'arbre est prépondérante en temps par rapport à une simple itération du k-mean.

[3] Ici, la notion de petitesse est fonction de n.

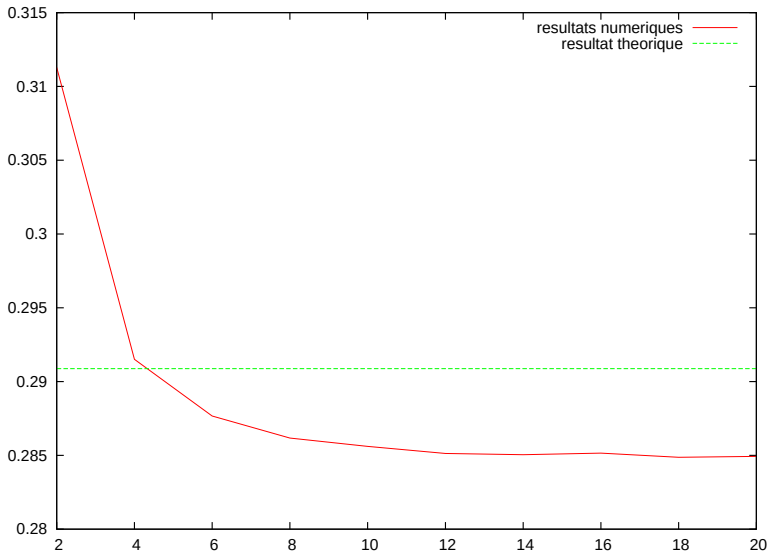

FIG. 5.7 – Résultats numériques obtenus en fonction du nombre de quantificateurs pour le brownien (Cas-test 2, $n = 100$, $q = 20000$, $n_{iter} = 20$, $x_0 = 1.5$, $u(0, x_0) = 0.2909$).

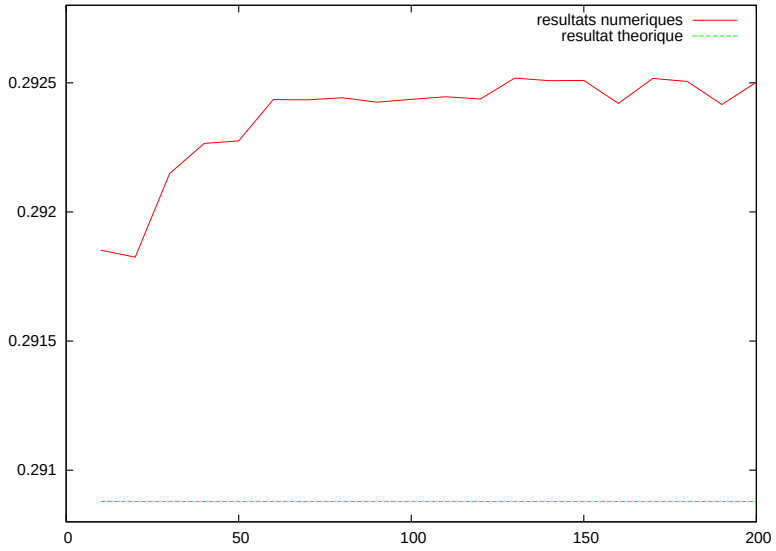

FIG. 5.8 – Résultats numériques obtenus en fonction du nombre de quantificateurs pour le brownien (Cas-test 2, $n = 1000$, $q = 400000$, $n_{iter} = 20$, $x_0 = 1.5$, $u(0, x_0) = 0.2909$).

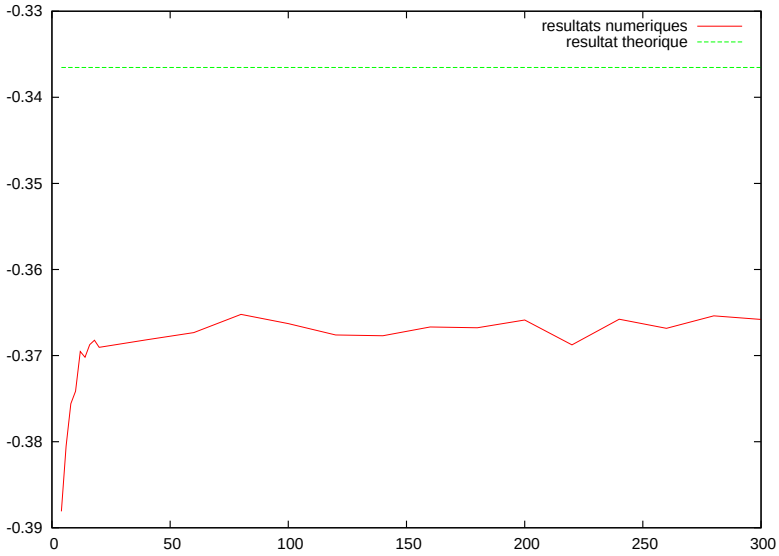

FIG. 5.9 – Résultats numériques obtenus en fonction du nombre de quantificateurs pour le brownien (Cas-test 5, $n = 100$, $q = 20000$, $n_{iter} = 20$, $(x_0, y_0) = (-0.2, -0.2)$, $u(0, x_0, y_0) = -0.3365$).

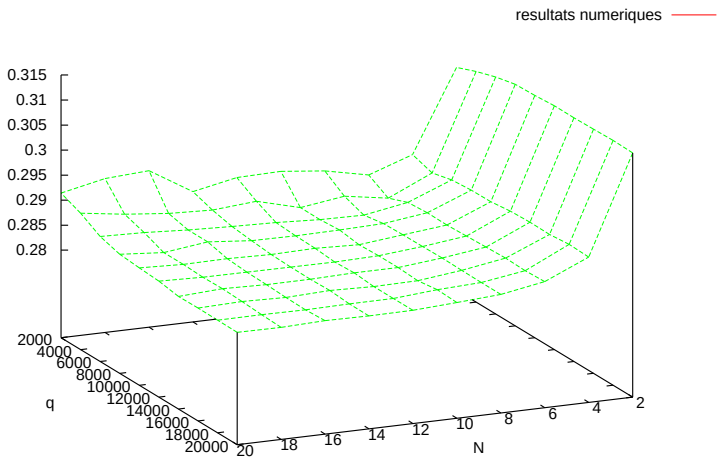

FIG. 5.10 – Résultats numériques obtenus en fonction du nombre de quantificateurs pour le brownien et du nombre total minimal de quantificateurs pour le processus forward (Cas-test 2, $n = 100$, $n_{iter} = 20$, $x_0 = 1.5$, $u(0, x_0) = 0.2909$).

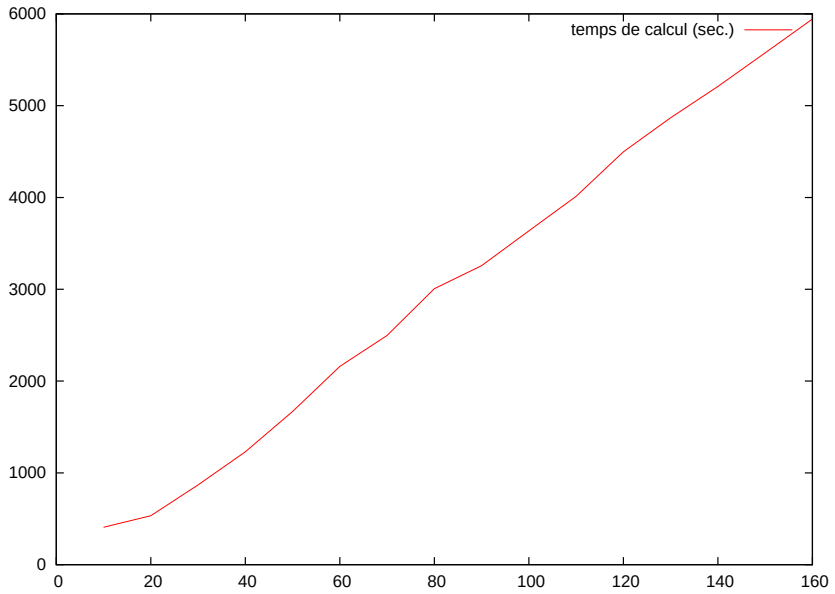

FIG. 5.11 – Temps de calcul (en secondes) en fonction du nombre de quantificateurs pour le brownien (Cas-test 2, $n = 1000$, $q = 400000$, $n_{iter} = 20$, $x_0 = 1.5$, $u(0, x_0) = 0.2909$).

Sensibilité à q et n

A priori n_{iter}, N et q permettent de modifier les distorsions totales lorsque n est fixé. En fait, nous avons vus que N n'intervient pas et que n_{iter} est fixé une fois pour toute. Ainsi, q est le seul paramètre nous permettant d'intervenir sur les distorsions totales et donc sur l'erreur spatiale. La figure 5.12 illustre cette dépendance. Concernant la dépendance du temps de calcul vis-à-vis de q, la figure 5.13 ne permets pas de confirmer ou d'infirmer la complexité temporelle estimée.

En ce qui concerne n, l'existence d'une valeur optimale lorsque les autres paramètres sont fixés semble logique. En effet, si ce dernier est trop petit, l'erreur de discrétisation temporelle devient trop importante et s'il est trop grand, le nombre moyen de quantificateurs par pas de temps est trop faible et donc l'erreur de discrétisation spatiale devient prépondérante. La figure 5.14 illustre ce phénomène. Une étude plus approfondie, s'appuyant sur des résultats de [2], est menée dans la partie 5.3.2.

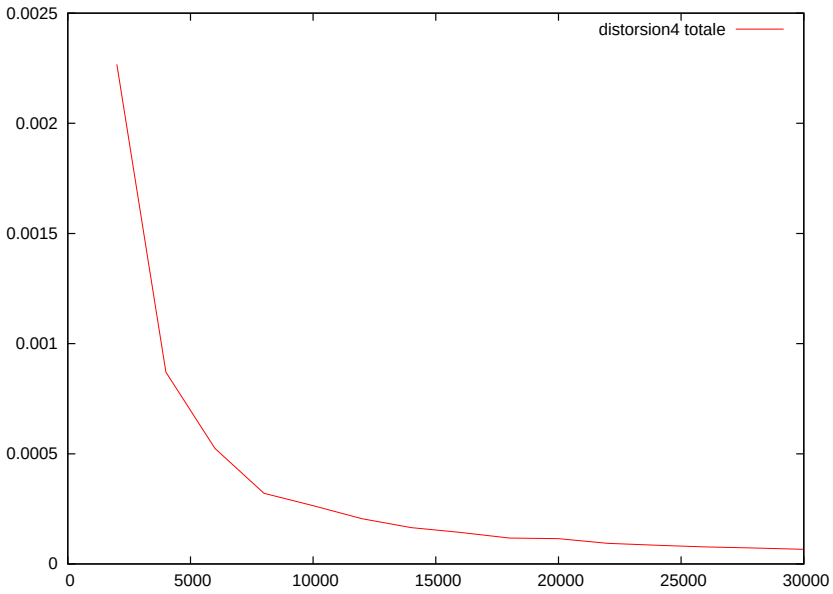

FIG. 5.12 – Somme des distorsions 4 du k-mean sur tous les pas de temps en fonction du nombre total de quantificateurs (Cas-test 2, $n = 100$, $N = 10$, $n_{iter} = 20$).

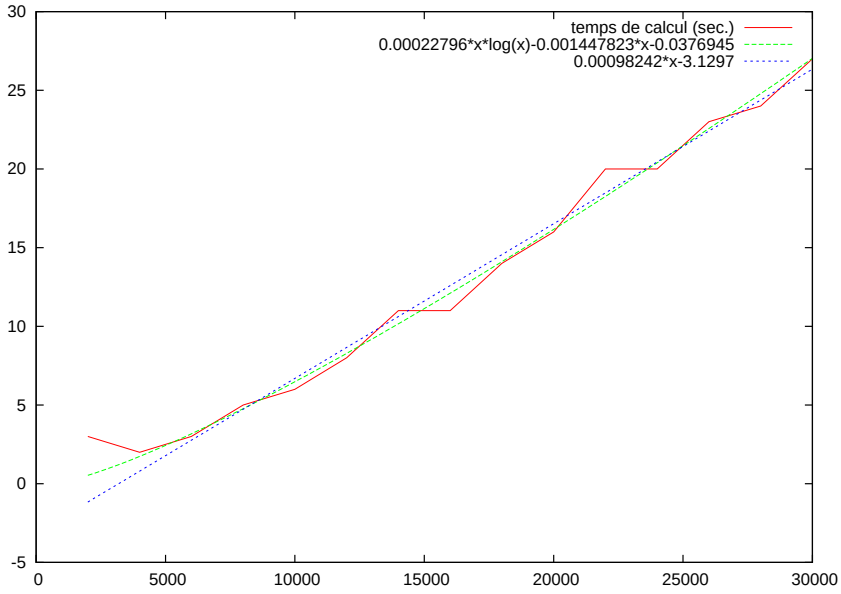

FIG. 5.13 – Temps de calcul (en secondes) en fonction du nombre total de quantificateurs (Cas-test 2, $n = 100$, $N = 10$, $n_{iter} = 20$). Sont également représentées, la régression linéaire et la régression du type $ax \log x + bx + c$.

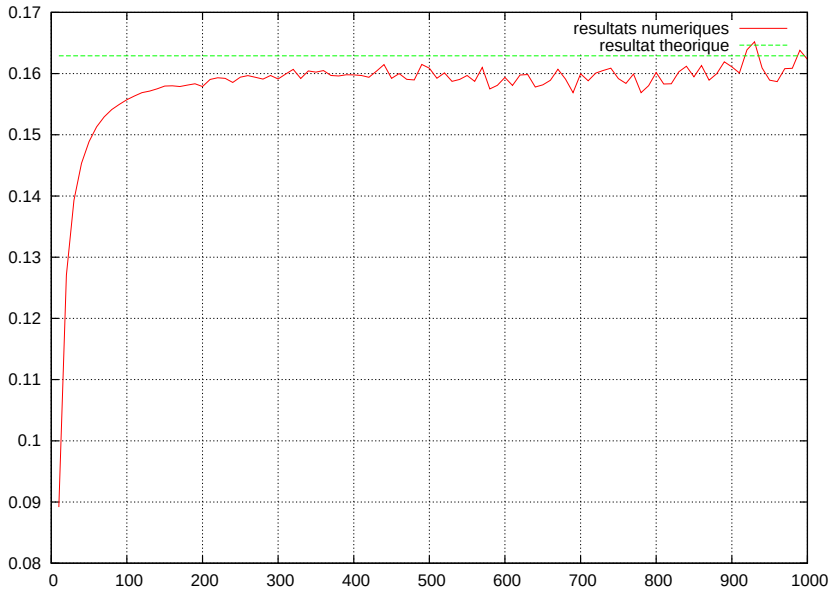

FIG. 5.14 – Résultats numériques obtenus en fonction du nombre de pas de temps (Cas-test 2, $N = 10$, $q = 20000$, $n_{iter} = 20$).

5.2.3 Comparaisons entre les deux types de projections

Tous les tests réalisés en dimension 1 montrent que, à temps de calcul équivalent, les deux algorithmes fournissent des résultats similaires. Les figures 5.15, 5.16, 5.17, 5.18, 5.19 et 5.20 représentent les résultats obtenus pour les cas-tests 1, 2 et 3 à l'aide des deux types d'algorithmes. Ils ont été obtenus avec approximativement les mêmes temps de calcul, à savoir entre 10 et 30 secondes par points. En ce qui concerne la dimension 2, nous n'aboutissons pas aux mêmes conclusions. Les figures 5.21 et 5.22 représentent les résultats obtenus pour le cas-test 5 à l'aide de l'algorithme du k-mean. Le temps de calcul moyen par point fut d'environ 300 secondes. Par contre, nous n'avons pas pu obtenir de résultats corrects pour l'algorithme utilisant la projection sur une grille fixe. Si l'on prend des paramètres trop fins ($\delta = 0.01$ par exemple), la complexité spatiale devient trop importante ce qui a pour conséquence de faire exploser les temps de calcul. En effet, lorsque le nombre de quantificateurs par pas de temps devient trop important, la mémoire vive sature[4] et le programme utilise le disque dur comme de la mémoire virtuelle ce qui a pour conséquence de ralentir énormément la vitesse d'exécution du programme[5]. Concrètement, une nuit complète de calcul ne suffit pas pour l'obtention d'un point. Certes, en utilisant des paramètres un peu plus grossiers et un ordinateur plus puissant, il doit être possible d'obtenir des résultats en un temps raisonnable. Néanmoins, ces problèmes montrent que l'utilisation de grilles fixes est à proscrire, au profit du k-mean, pour les dimensions supérieures à 1.

[4] Les programmes ont pourtant été codés avec l'idée de minimiser la quantité d'information en mémoire vive. Ainsi, pour la quantification de $\bar{X}$ à l'instant t_i, la mémoire vive ne contient « que » les Nq_{i-1} quantificateurs générés à partir de l'instant précédent, les q_i quantificateurs en cours de calcul et la matrice de transition entre les deux instants, de taille Nq_{i-1}. Les autres informations sont stockées sur le disque dur.

[5] Ce phénomène est amplifié par la faible capacité de la mémoire vive de l'ordinateur sur lequel ont été menés les essais.

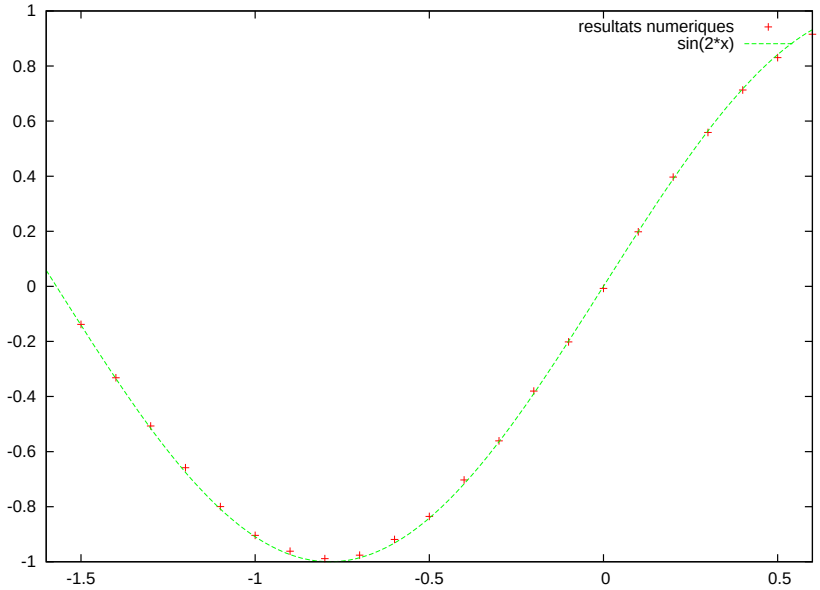

FIG. 5.15 – Résultats numériques obtenus pour le cas-test 1 ($n = 100$, $N = 10$, $q = 20000$, $n_{iter} = 20$).

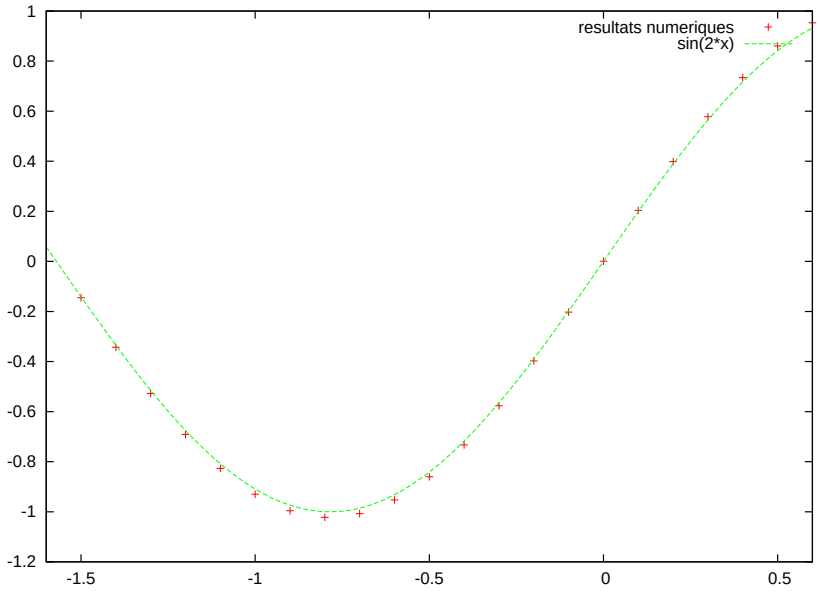

FIG. 5.16 – Résultats numériques obtenus pour le cas-test 1 à l'aide de la projection sur une grille fixe ($n = 100$, $N = 10$, $\delta = 0.001$, $R = 4$).

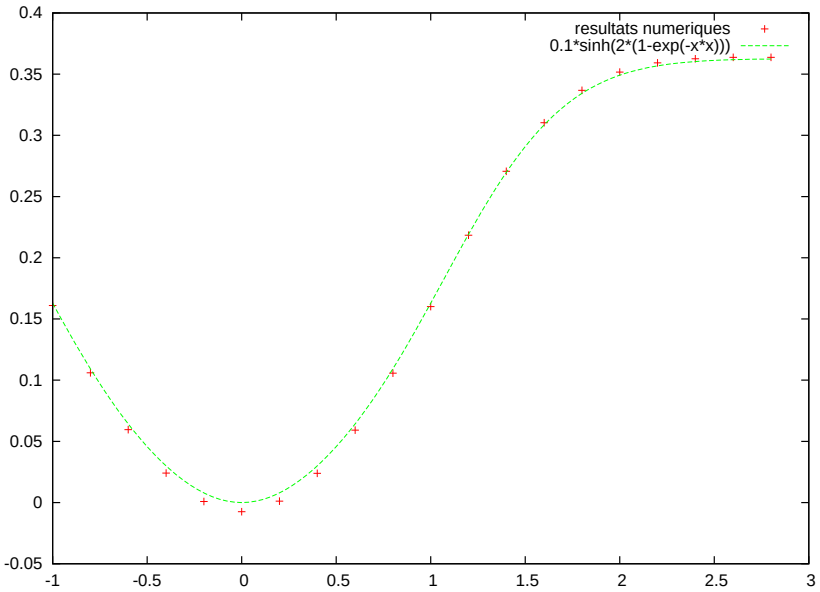

FIG. 5.17 – Résultats numériques obtenus pour le cas-test 2 ($n = 1000$, $N = 10$, $q = 100000$, $n_{iter} = 20$).

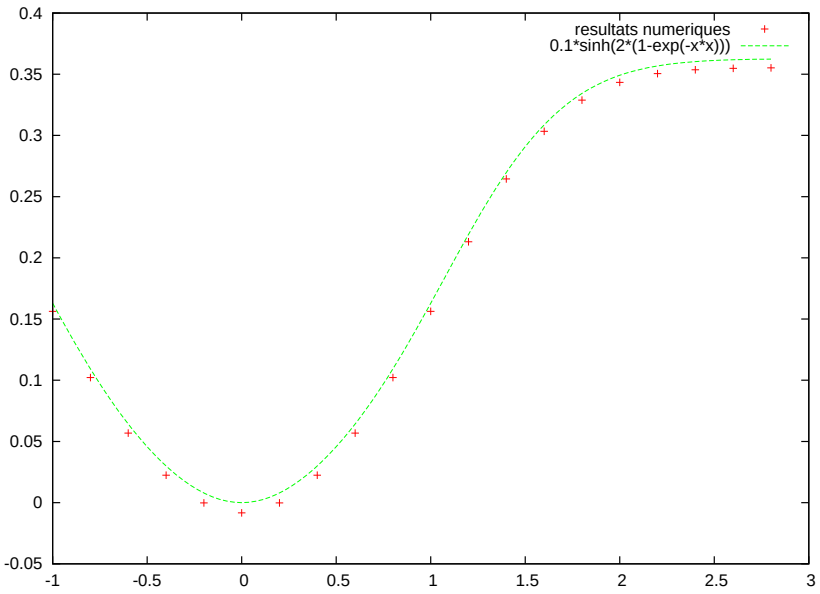

FIG. 5.18 – Résultats numériques obtenus pour le cas-test 2 à l'aide de la projection sur une grille fixe ($n = 100$, $N = 10$, $\delta = 0.001$, $R = 4$).

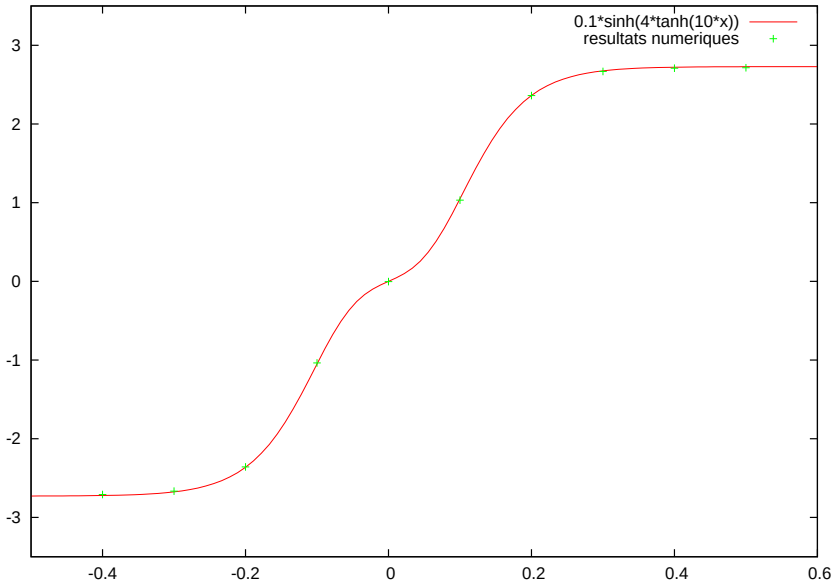

FIG. 5.19 – Résultats numériques obtenus pour le cas-test 3 ($n = 100$, $N = 10$, $q = 10000$, $n_{iter} = 20$).

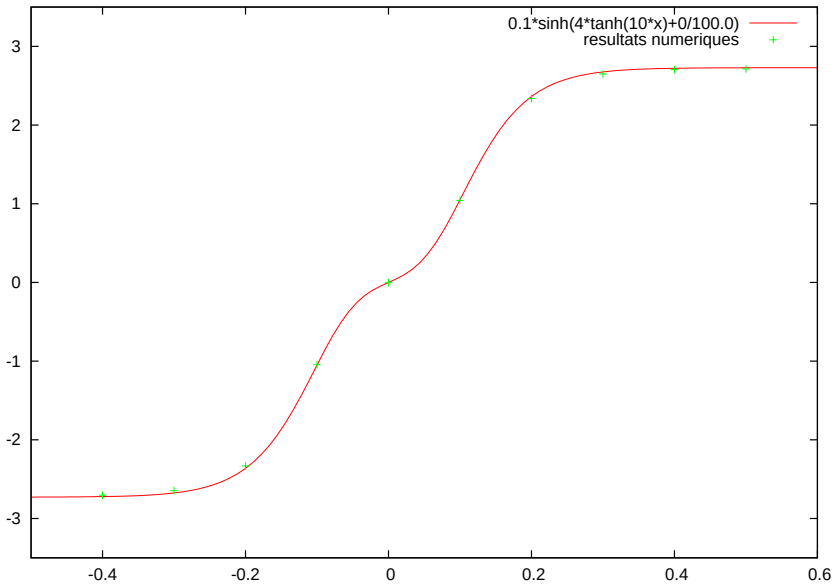

FIG. 5.20 – Résultats numériques obtenus pour le cas-test 3 à l'aide de la projection sur une grille fixe ($n = 100$, $N = 10$, $\delta = 0.001$, $R = 4$).

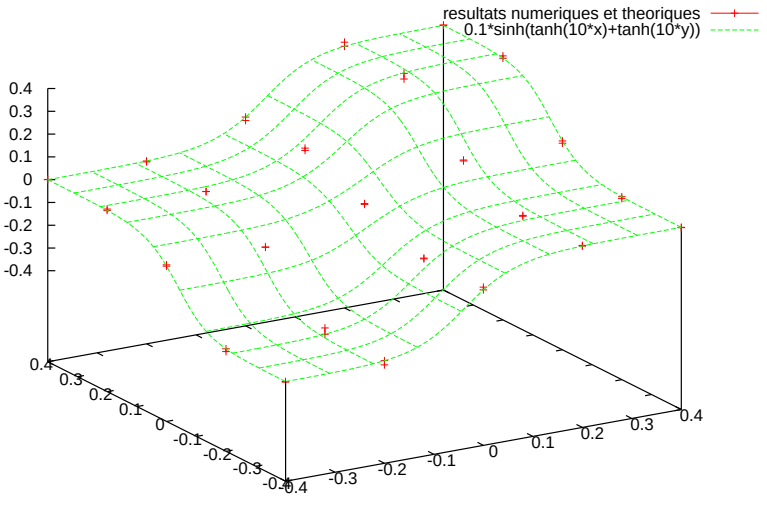

FIG. 5.21 – Résultats numériques obtenus pour le cas-test 5 ($n = 100$, $N = 10$, $q = 20000$, $n_{iter} = 20$). Les résultats théoriques et numériques sont représentés par deux croix reliées par un segment.

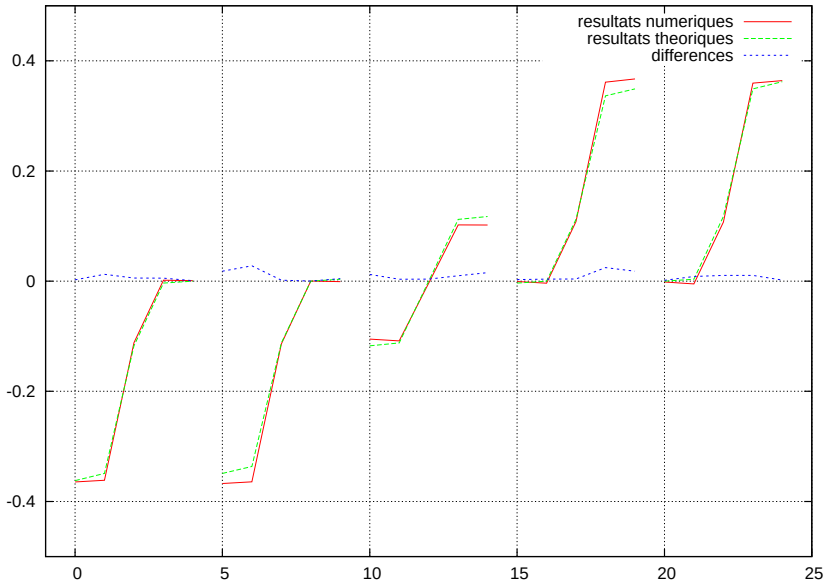

FIG. 5.22 – Résultats numériques obtenus pour le cas-test 5 ($n = 100$, $N = 10$, $q = 20000$, $n_{iter} = 20$). Ce sont ceux de la figure 5.21 présentés par « tranches ».

5.3 Applications à la finance

Dans toute cette section, nous n'utiliserons que le schéma reposant sur l'utilisation du k-mean. De plus, on note

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} du.$$

5.3.1 Evaluation d'options européennes

L'évalution d'options européennes est, certes, une application possible de notre algorithme, mais ce n'est pas un but en soi. En effet, par l'intermédiaire de la formule de Feynman-Kac (1.1), il est possible d'utiliser des méthodes de Monte-Carlo qui sont bien plus simples à mettre en œuvre. Néanmoins, l'existence de solutions analytiques nous permet de comparer les résultats obtenus.

Reprenons les calculs de la partie 1.4.1 en supposant que le processus X vérifie

$$dX_t^i = X_t^i \left[\mu^i dt + \sum_{j=1}^{d} \sigma^{i,j} dB_t^j \right], \quad 1 \leq i \leq d,$$

avec σ inversible. Alors,

$$Y_t = g(X_T) - \int_t^T r Y_s ds + \int_t^T \sum_{i=1}^{n} \theta_s^{1,i} X_s^i \left[(\mu^i - r) ds + \sum_{j=1}^{n} \sigma^{i,j} dB_s^j \right].$$

D'après le théorème de Girsanov ([11]), il existe une probabilité $\mathbb{P}^*$ appelée probabilité risque-neutre telle que, sous cette probabilité, le processus W, défini par[6]

$$dW_t = dB_t + \sigma^{-1}(\mu - r\mathbb{1})dt,$$

est un mouvement brownien de dimension d. Alors, on a

$$dX_t^i = X_t^i \left[r dt + \sum_{j=1}^{d} \sigma^{i,j} dW_t^j \right]$$

et

$$Y_t = g(X_T) - \int_t^T r Y_s ds + \int_t^T \sum_{i=1}^{n} \theta_s^{1,i} X_s^i . \left[\sum_{j=1}^{n} \sigma^{i,j} dW_s^j \right].$$

On peut donc appliquer notre algorithme avec

$$\begin{cases} b(x) &= rx \\ \sigma(x) &= (\sigma^{i,j} x^i)_{i,j} \\ f(t,x,y) &= -ry \\ g & \text{le payoff de l'option} \end{cases}$$

Pour nos tests numériques nous allons considérer deux cas-tests en dimension 2.

[6] $\mathbb{1}$ représente le vecteur de dimension d dont toutes les coordonnées sont 1.

Digitale Le payoff vaut $\mathbb{1}_{X_T^1 > X_T^2}$. On a [7]

$$Y_0 = e^{-rT}\Phi\left(\frac{\log(x_0^1/x_0^2) - \frac{T}{2}((\sigma^{1,1})^2 + (\sigma^{1,2})^2 - (\sigma^{2,1})^2 - (\sigma^{2,2})^2)}{\sqrt{T((\sigma^{1,1}-\sigma^{2,1})^2 + (\sigma^{1,2}-\sigma^{2,2})^2)}}\right).$$

Put géométrique Le payoff vaut $\left(K - \sqrt{X_T^1 X_T^2}\right)^+$. Lorsque $\sigma^{1,1} = \sigma^{2,2} = c$ et $\sigma^{1,2} = \sigma^{2,1} = 0$, on montre que[8]

$$Y_0 = Ke^{-rT}\Phi(-d_2) - \sqrt{x_0^1 x_0^2}e^{-c^2T/4}\Phi(-d_1)$$

avec

$$d_1 = \frac{\log(\sqrt{x_0^1 x_0^2}/K) + rT}{c\sqrt{T/2}}, \quad d_2 = d_1 - c\sqrt{T/2}.$$

Les figures 5.23 et 5.24 représentent des résultats obtenus pour ces deux exemples avec les paramètres

$$\begin{cases} r &= 0.04 \\ \sigma &= \begin{pmatrix} 0.2 & 0 \\ 0 & 0.2 \end{pmatrix} \\ K &= 110 \end{cases}.$$

[7] cf [12] pour une démonstration du résultat.

[8] Pour démontrer le résultat il suffit de se ramener à un problème de dimension 1 et d'appliquer la formule de Black-Scholes.

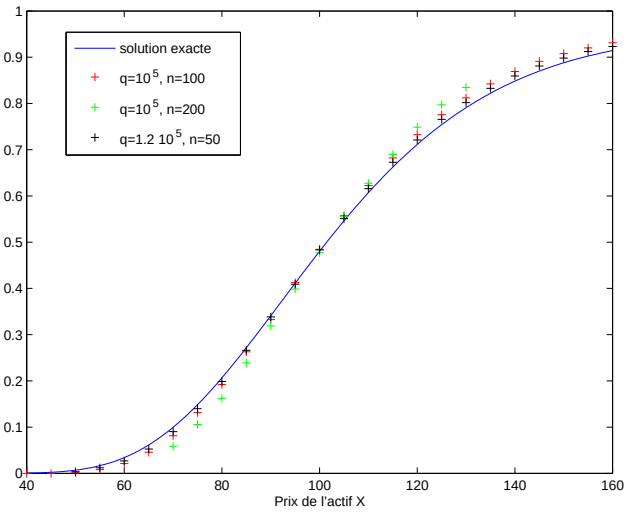

FIG. 5.23 – Résultats obtenus pour une digitale en dimension 2. Le prix est donné en fonction de x_0^1, x_0^2 étant fixé à 100. On a $N = 30$ et $n_{iter} = 20$.

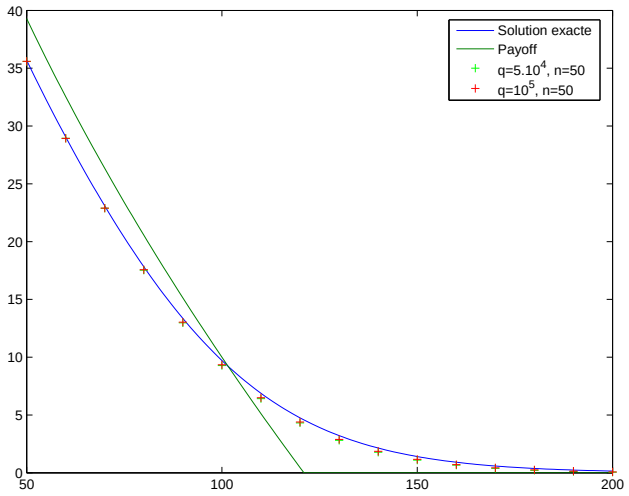

FIG. 5.24 – Résultats obtenus pour un put géométrique en dimension 2. Le prix est donné en fonction de x_0^1, x_0^2 étant fixé à 100. On a $N = 30$ et $n_{iter} = 20$.

5.3.2 Résultats complémentaires sur les paramètres

Le choix des q_i

Les auteurs de [1] énoncent le résultats suivant :

Théorème 5.3.1 *On suppose que b et σ sont infiniments différentiables, que leurs différentielles sont bornées et qu'il existe $\varepsilon > 0$ tel que $\sigma - \varepsilon I_d$ soit positive. Alors il existe deux constantes α et β telles que, pour h suffisamment petit, les densités par rapport à la mesure de Lebesgue de $\bar{X}_{t_0}, \ldots, \bar{X}_{t_n}$, notées $\bar{p}_{t_0}, \ldots, \bar{p}_{t_n}$, vérifient*

$$\bar{p}_{t_i}(x) \leq \frac{\alpha}{(2\pi t_k)^{d/2}} \exp\left(-\frac{|x - x_0|^2}{2\beta t_k}\right)$$

les références pour la démonstrations sont données dans [1]. En introduisant cette majoration dans le théorème de Zador (2.1.3), on obtient, pour h suffisamment petit,

$$N^{2/d} \min_{|\Gamma| \leq N} \left\| \bar{X} - \hat{\bar{X}}^{\Gamma} \right\|_2^2 \leq C_{d,\alpha,\beta} t_k.$$

Il semble donc intéressant de prendre

$$q_i = \left\lceil \frac{t_i^{\frac{d}{2}}(q-1)}{t_1^{\frac{d}{2}} + \ldots + t_n^{\frac{d}{2}}} \right\rceil, \quad 1 \leq i \leq n. \tag{5.2}$$

En pratique, tous les tests réalisés en dimension 1 et 2 avec les formules (5.1) et (5.2) ne montrent pas de différences significatives entre ces deux types de répartitions. Pour ajouter des critères de jugement, nous avons également comparé la propagation des erreurs dans le temps en regardant l'évolution de

$$e_{t_k} := \sum_{i=1}^{q_{t_k}} \mathbb{P}\left[\hat{X}_{t_k} = x_{t_k}^i\right] \left| Y_{t_k}^{x_{t_k}^i} - \hat{Y}_{t_k}^{x_{t_k}^i} \right|^2.$$

La figure 5.25 illustre un exemple de résultat obtenu. Là encore, nous n'avons pas noté de différences significatives entre les deux types de répartitions parmi tous les tests réalisés. Ainsi, toute proportion gardée, le choix des q_i ne semble pas revêtir une importance extrême.

Retour sur la sensibilité au paramètre n

Les auteurs de [1] montrent empiriquement, en s'appuyant sur des résultats établis théoriquement, qu'à $\bar{q} = q/n$ fixé, il existe un n optimal qui minimise l'erreur de leur schéma et que cette erreur est de la forme $\frac{C_1}{n} + \frac{C_2 n}{\bar{q}^{1/d}}$ avec C_1 et C_2 deux constantes qui dépendent des autres paramètres. Nous avons donc voulu voir si nous pouvions obtenir empiriquement les mêmes types de conclusion. Les résultats de la figure 1 semblent indiquer qu'à $\bar{q} = q/n$ fixé, il existe bien un n optimal qui minimise l'erreur mais que, dans notre cas, celle-ci se comporte comme $C_1(\bar{q})/n + C_2((\bar{q}))\sqrt{n}$. Les constantes interpolées sont regroupées dans le tableau suivant :

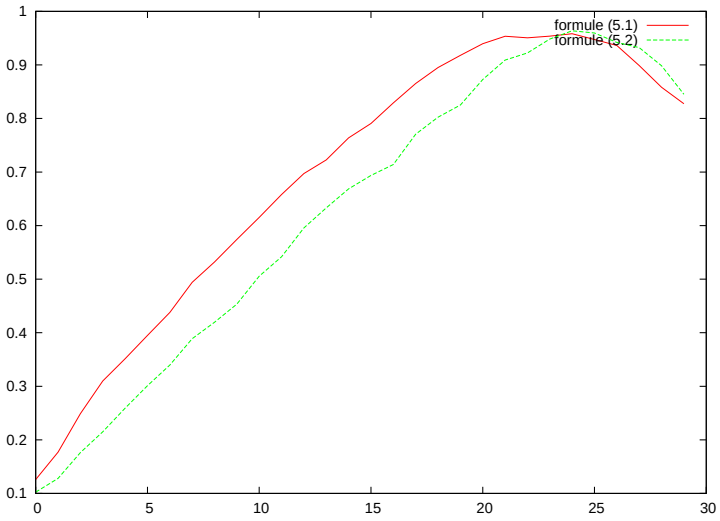

FIG. 5.25 – e_{t_k} en fonction de t_{n-k+1} pour les deux types de répartitions (5.1) et (5.2). Les valeurs ont été obtenues pour un put géométrique européen en dimension 2 avec $r = 0.04$, $\sigma = 0.2I_d$, $K = 110$, $x_0^1 = x_0^2 = 100$, $n = 20$, $n_{iter} = 20$ et $q = 10000$.

	$\bar{q} = 20$	$\bar{q} = 30$	$\bar{q} = 40$	$\bar{q} = 50$
C_1	$0,32$	$0,41$	$0,42$	$0,46$
C_2	$0,078$	$0,055$	$0,044$	$0,036$
α	$0,86$	$0,77$	$0,91$	$\times$

Dans ce tableau, α est l'estimation de la puissance de $\bar{q}$ pour un modèle d'erreur de la forme $\frac{C_1}{n} + \frac{C_2\sqrt{n}}{\bar{q}^\alpha}$:

$$\alpha_i := \frac{\log(C_2(\bar{q}_{i+1})/C_2(\bar{q}_i))}{\log \bar{q}_i/\bar{q}_{i+1}}.$$

Contrairement aux valeurs de [1], les résultats obtenus fluctuent trop pour pouvoir valider ce modèle d'erreur : C_1 n'est pas une constante indépendante de $\bar{q}$ et α a un comportement irrégulier. Pour compléter ces observations, il conviendrait de faire des tests dans des dimensions supérieures afin d'étudier l'influence de d.

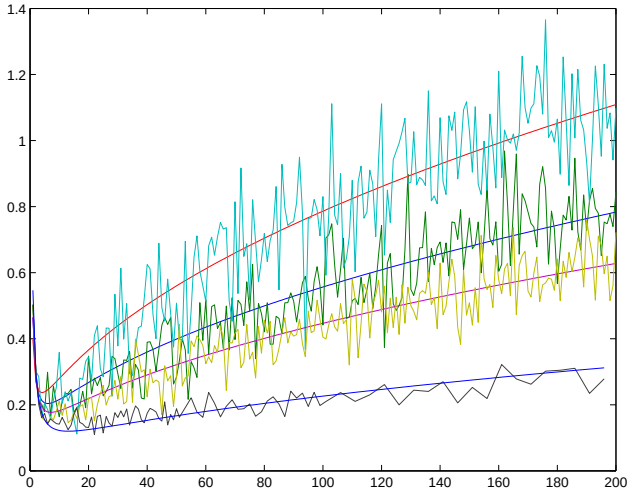

FIG. 5.26 – Variations de l'erreur en fonction de n pour différents $\bar{q}$ (20, 30, 40 et 50). Les interpolations du type $a/x + b\sqrt{x}$ sont également représentées. Ces résultats ont été obtenus pour un put européen en dimension 1 avec $r = 0.04$, $\sigma = 0.2$, $K = 110$, $x_0 = 100$, $n_{iter} = 20$ et $N = 30$.

5.3.3 Evaluation d'options américaines

Au cours de la construction du schéma d'approximation pour Y, nous avons vu qu'il était simple de le généraliser au cas d'EDSR réfléchies[9]. Ainsi, en reprenant les notations des chapitres 1 et 4, on a le nouveau schéma :

$$\hat{Y}_{t_k} = \bar{u}(t_k, \hat{X}_{t_k}) \tag{5.3}$$

avec

$$\left\{ \begin{array}{rcl} \bar{u}(T,x) & = & g(x) \\ \bar{u}(t_k,x) & = & \sup\left(h(t_k,x), \mathbb{E}\left[\bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k,x))) \right.\right. \\ & & \left.\left. + f(t_k, x, \bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k,x))))h\right]\right) \quad \forall x \in \mathcal{C}_k \end{array} \right. . \tag{5.4}$$

Nous allons utiliser ce schéma pour l'évaluation d'options américaines. Contrairement au cas européen, nous n'avons pas de solution explicite, il faudra donc faire appel à d'autres méthodes de résolution afin de pouvoir comparer les résultats obtenus. Pour la dimension 1 nous avons à notre disposition un solveur, utilisant la méthode des différences finies, développé en *Matlab* pour un projet de Supaero [10]. Pour des dimensions supérieures, nous avons téléchargé le logiciel *Premia* développé par le projet *MATHFI* de l'INRIA et disponible

[9] cf la partie 1.5.

[10] Ce solveur utilise, au choix, un schéma explicite ou un θ-schéma implicite.

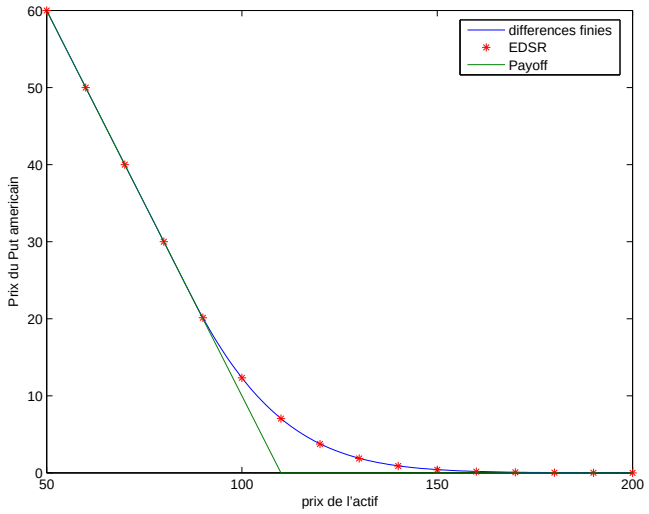

FIG. 5.27 – Prix d'un put américain de dimension 1 en fonction du prix de l'actif sous-jacent. On a $r = 0,4$, $\sigma = 0,2$, $K = 110$, $n = 20$, $q = 10000$, $n_{iter} = 20$ et $N = 30$.

à l'adresse `http://www-rocq.inria.fr/mathfi/Premia/down.html`. Malheureusement, nous n'avons pas eu le temps de l'utiliser. Ainsi, nous avons juste à notre disposition les résultats obtenus pour un put américain en dimension 1. Dans ce cas, le payoff vaut $(K - x)^+$ et $h(t, x) = g(x)$. La figure 5.27 a été obtenue avec les paramètres

$$\left\{ \begin{array}{rcl} r & = & 0.04 \\ \sigma & = & 0.2 \\ K & = & 110 \end{array} \right. .$$

5.3.4 Où l'on reparle du processus Z

Construction d'une approximation de Z

Les auteurs de [4] proposent une méthode pour approcher le processus Z, nous allons donc reprendre leur démarche. Comme on l'a déjà fait dans la partie 1.5, nous allons réécrire l'EDSR entre les instants t_k et t_{k+1} :

$$Y_{t_k}^{t_k,X_{t_k}} = Y_{t_{k+1}}^{t_k,X_{t_k}} + \int_{t_k}^{t_{k+1}} f(s, X_s^{t_k,X_{t_k}}, Y_s^{t_k,X_{t_k}}, Z_s^{t_k,X_{t_k}})ds - \int_{t_k}^{t_{k+1}} Z_s^{t_k,X_{t_k}} dB_s.$$

En multipliant par ΔB^k et en prenant l'espérance, on a

$$\mathbb{E}\left[Y_{t_k}^{t_k,X_{t_k}} \Delta B^k\right] = 0$$

et

$$\mathbb{E}\left[\int_{t_k}^{t_{k+1}} Z_s^{t_k,X_{t_k}} dB_s \int_{t_k}^{t_{k+1}} dB_s\right] = \mathbb{E}\left[\int_{t_k}^{t_{k+1}} Z_s^{t_k,X_{t_k}} ds\right].$$

Donc

$$\mathbb{E}\left[\int_{t_k}^{t_{k+1}} Z_s^{t_k,X_{t_k}} ds\right] = \mathbb{E}\left[Y_{t_{k+1}}^{t_k,X_{t_k}} \Delta B^k\right] + \mathbb{E}\left[\int_{t_k}^{t_{k+1}} f(s, X_s^{t_k,X_{t_k}}, Y_s^{t_k,X_{t_k}}, Z_s^{t_k,X_{t_k}}) ds \Delta B^k\right].$$

Notons A le dernier terme. En appliquant plusieurs fois l'inégalité de Cauchy-Schwartz, on trouve

$$|A| \leq h\mathbb{E}\left[\int_{t_k}^{t_{k+1}} f^2(s, X_s^{t_k,X_{t_k}}, Y_s^{t_k,X_{t_k}}, Z_s^{t_k,X_{t_k}}) ds\right]^{1/2}.$$

Sous les hypothèses $(\mathcal{H}_1)$, f est bornée vis-à-vis des deux premières variables. De plus, le théorème 1.2.5 assure que Y et Z sont bornés, donc, comme f est continue, on va pouvoir la majorer par une constante[11] :

$$|A| \leq Ch^{3/2}.$$

Ainsi, nous allons approcher $Z_{t_k}^{t_k,X_{t_k}}$ par le processus discrétisé en temps

$$\bar{Z}_{t_k}^{t_k,\bar{X}_{t_k}} = h^{-1}\mathbb{E}\left[\bar{Y}_{t_{k+1}}^{t_k,\bar{X}_{t_k}} \Delta B^k\right].$$

Il reste alors à discrétiser en espace. On pose finalement

$$\hat{Z}_{t_k}^{t_k,\hat{X}_{t_k}} = h^{-1}\mathbb{E}\left[\hat{Y}_{t_{k+1}}^{t_k,\hat{X}_{t_k}} \hat{\Delta B}^k\right].$$

On peut alors réécrire le schéma de discrétisation sous la forme

$$\hat{Y}_{t_k} = \bar{u}(t_k, \hat{X}_{t_k}) \tag{5.5}$$
$$\hat{Z}_{t_k} = \bar{v}(t_k, \hat{X}_{t_k}) \tag{5.6}$$

avec

$$\begin{cases} \bar{v}(T,x) = \sigma\nabla g(x) \\ \bar{u}(T,x) = g(x) \\ \bar{v}(t_k,x) = h^{-1}\mathbb{E}\left[\bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k,x)))\hat{\Delta B}^k\right] \\ \bar{u}(t_k,x) = \mathbb{E}\left[\bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k,x))) \right. \\ \qquad \left. + f(t_k, x, \bar{u}(t_{k+1}, \Pi_{k+1}(x + \mathcal{T}(t_k,x))), \bar{v}(t_k,x),)h\right] \quad \forall x \in \mathcal{C}_k \end{cases} \tag{5.7}$$

tandis que X vérifie toujours (4.1). Notons que l'on pourrait également définir un schéma où le calcul de $\hat{Y}_{t_k}$ dépend de $\hat{Z}_{t_{k+1}}$ au lieu de $\hat{Z}_{t_k}$.

[11] Sous les hypothèses $(\mathcal{H}_2)$, les majorations sont plus difficiles à mener. De plus, Les résultats de majoration des processus Y et Z établis par [18] ne sont pas suffisants pour aboutir à la même majoration de A.

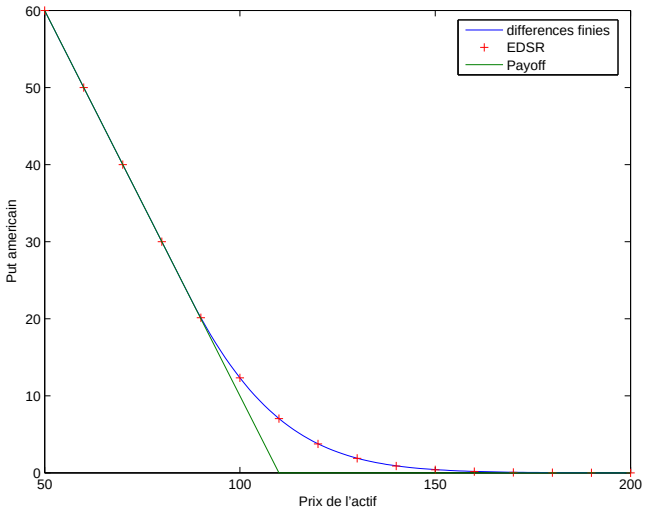

FIG. 5.28 – Prix d'un put américain de dimension 1 en fonction du prix de l'actif sous-jacent. L'évaluation est réalisée sans utiliser la probabilité risque neutre. On a $r = 0,4$, $\sigma = 0,2$, $K = 110$, $n = 20$, $q = 10000$, $n_{iter} = 20$ et $N = 30$.

Résultat numériques

Afin de juger la pertinence de ce nouveau schéma, nous l'avons appliqué pour évaluer des options sans utiliser la probabilité risque neutre. Par exemple, prenons le cas d'un put américain[12] en dimension 1 avec X qui suit un processus de Black-Scholes. On a alors

$$\begin{cases} b(x) &= \mu x \\ \sigma(x) &= \sigma x \\ f(t,x,y,z) &= -ry + \frac{z}{\sigma}(\mu - r)x \\ g(x) = (K-x)^+ \end{cases}$$

La figure 5.28 a été obtenue avec les paramètres

$$\begin{cases} r &= 0.04 \\ \mu &= 0.1 \\ \sigma &= 0.2 \\ K &= 110 \end{cases}.$$

Dans la pratique, l'évaluation du contrôle Z_0 est aussi importante que la valeur Y_0 car c'est en connaissant ce contrôle que l'on peut mener à bien la stratégie

[12] Dans ce cas, le schéma est modifié en conséquence pour traiter les EDSR réfléchies.

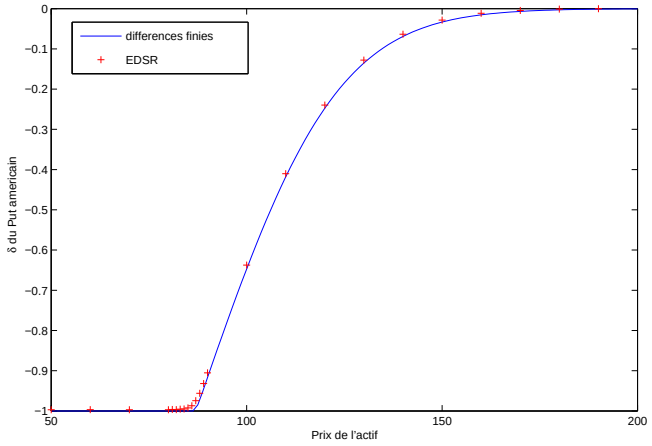

FIG. 5.29 – Delta d'un put américain de dimension 1 en fonction du prix de l'actif sous-jacent. On a $r = 0,4$, $\sigma = 0,2$, $K = 110$, $n = 20$, $q = 10000$, $n_{iter} = 20$ et $N = 30$.

de réplication[13] (cf la partie 1.4.1). Notons que la quantité $\theta_0^1 = Z_0^* \sigma^{-1}(x_0)$ est appelée le « delta » de l'option. Si l'on note $u(t,x) = Y_t^{t,x}$, elle est également égale à $\frac{\partial u}{\partial x}(0, x_0)$. Ainsi, on peut utiliser l'estimateur de Z défini plus haut pour approcher Z_0 et donc le delta de l'option :

$$\hat{Z}_{t_0}^{t_0,x_0} = h^{-1}\mathbb{E}\left[\hat{Y}_{t_1}^{t_0,x_0}\Delta\hat{B}^0\right].$$

Nous pouvons remarquer qu'il est possible d'utiliser cette approximation avec le schéma défini initialement, c'est-à-dire lorsque f est indépendante de z. La figure 5.29 représente les résultats obtenus pour le calcul du delta d'un put américain en dimension 1, sous la probabilité risque neutre, avec les paramètres

$$\begin{cases} r &= 0.04 \\ \sigma &= 0.2 \\ K &= 110 \end{cases}.$$

5.3.5 Quelques développements envisageables

Utilisation de l'interpolation

En tentant d'améliorer les résultats de la figure 5.24, nous nous sommes aperçus que ces derniers étaient très sensibles au paramètre N. Le tableau qui suit illustre ce phénomène[14].

[13] Rappelons que c'est cette stratégie qui justifie le prix de l'option.

[14] Les valeurs ont été obtenues pour un put géométrique européen en dimension 2 avec $r = 0.04$, $\sigma = 0.2I_d$, $K = 110$, $x_0^1 = x_0^2 = 100$, $n = 20$, $n_{iter} = 20$ et $q = 20000$.

valeur théorique	$N = 40$	$N = 100$	$N = 500$	$N = 600$
$9,71$	$9,37$	$9,41$	$9,48$	$9,54$

Pour tenter de comprendre l'origine de ces résultats, nous avons étudié plus en détail les matrices de transition obtenues. En se plaçant à un instant t_k et en prenant pour origine un quantificateur $x^i_{t_k}$, nous nous sommes rendu compte que le nombre de quantificateurs atteints à l'instant t_{k+1}, parmi $\left\{x^1_{t_{k+1}}, \ldots, x^{q_{t_{k+1}}}_{t_{k+1}}\right\}$, n'évolue pas. Cela signifie que l'amélioration des résultats est uniquement due à la meilleure approximation des probabilités de transition. À partir de ce constat, plusieurs idées sont envisageables pour améliorer l'estimation de ces probabilités.

- La solution la plus simple qui consiste à augmenter N ne semble pas être le remède le plus efficace car il accentue énormément les temps de calcul de l'algorithme du k-mean.
- Une autre idée consiste à réaliser la quantification de $\bar{X}$ par les mêmes méthodes que précedemment, puis de réestimer les transitions à l'aide de méthodes de Monte-Carlo. Malheureusement, cela nécessiterait d'appliquer $\sum_{i=0}^{n} q_i$ fois Monte-Carlo, ce qui demanderait encore plus de temps que la solution précedente.
- Il est possible de coupler les deux idées précédentes : Dans un premier temps, on réalise la quantification de $\bar{X}$ avec $N = N_1$, puis, les transitions sont réevaluées en utilisant une quantification optimale de la gaussienne avec N_2 quantificateurs. Bien sur, on prend $N_1 \ll N_2$. Cela permet de ne pas surcharger inutilement l'algorithme du k-mean dans la première phase et d'avoir des temps de calcul beaucoup plus faibles que pour les méthodes de Monte-Carlo dans la seconde phase.
- On peut remarquer que la mauvaise approximation des probabilités de transition est due aux projections du k-mean. Il serait donc intéressant d'éviter ces projections dans le calcul du processus $\hat{Y}$. Dans ce but, les auteurs de [3] proposent d'utiliser des fonctions interpolées dans l'algorithme de résolution. En pratique, cela nous donne le schéma suivant

$$\hat{Y}_{t_k} = \bar{u}(t_k, \hat{X}_{t_k}) \tag{5.8}$$

avec

$$\begin{cases} \bar{u}(T,x) &= g(x) \\ \bar{u}(t_k,x) &= \mathbb{E}\left[\tilde{u}(t_{k+1}, x + \mathcal{T}(t_k,x)) \right. \\ & \quad \left. + f(t_k, x, \tilde{u}(t_{k+1}, x + \mathcal{T}(t_k,x)))h\right] \quad \forall x \in \mathcal{C}_k \\ \tilde{u}(t_k,.) & \quad \text{L'interpolation linéaire de } \bar{u}(t_k,.) \text{ sur } \mathbb{R}^d \end{cases} . \tag{5.9}$$

Dans le cas de projections sur des grilles de pas fixe, Le calcul de cette fonction interpolée ne pose pas de soucis car l'espace est alors « naturellement » pavé par des hyper-rectangles de $\mathbb{R}^d$. En ce qui concerne les projections utilisant le k-mean, le problème est beaucoup plus complexe car il faut construire une triangulation de Delaunay pour chaque grille $\mathcal{C}_k$. De plus, lorsque $x \in \mathcal{C}_k$, alors $x + \mathcal{T}(t_k, x)$ n'arrive pas forcément dans l'enveloppe convexe de $\mathcal{C}_k$.

Pour finir, notons que le phénomène étudié au départ, à savoir la forte sensibilité de la solution au paramètre N, vient contredire les conclusions de la partie 5.2.2. Cela pourrait s'expliquer par le manque de régularité de la fonction g. En tout état de cause, ces observations montrent que nos différents résultats sur la fixation des paramètres fluctuent suivant les différents cas-tests envisagés et doivent donc être considérés avec circonspection.

Le calcul des grecs

En finance, l'évaluation du prix d'une option est souvent couplée à des calculs de sensibilité. Les dérivées du prix vis-à-vis de différents paramètres[15] sont appelées les « grecs ». Nous avons déjà vu comment estimer directement le delta. Pour les autres grecs, il est possible de faire des différences finies. Cependant, cela nécessite de lancer deux fois l'algorithme de résolution : une fois avec le paramètre et une seconde fois avec le paramètre légèrement modifié. Il serait donc intéressant de pouvoir calculer directement les grecs, dès la première utilisation de l'algorithme. Une voie prometteuse est l'utilisation des méthodes de détermination des grecs appliquant le calcul de Malliavin. Sans rentrer dans la théorie, il semble possible d'appliquer certains résultats établis dans l'article [9]. Par exemple, si l'on note $u(t,x) = Y_t^{t,x}$, alors on a[16], lorsque $d = 1$,

$$\frac{\partial u}{\partial x}(0,x) = \mathbb{E}\left[e^{-rT} g(X_T) \frac{B_T}{x\sigma T}\right]$$

et

$$\frac{\partial^2 u}{\partial x^2}(0,x) = \mathbb{E}\left[e^{-rT} g(X_T) \frac{1}{x^2 \sigma T}\left(\frac{B_T^2}{\sigma T} - B_T - \frac{1}{\sigma}\right)\right].^{17}$$

Or, une option de maturité T et de payoff g a le même prix $u(0,x)$ qu'une option sur le même sous-jacent, de maturité t_1 et de payoff $u(t_1,.)$. Ainsi on a

$$\frac{\partial u}{\partial x}(0,x) \simeq h^{-1}\mathbb{E}\left[\hat{Y}_{t_1}^{0,x} \frac{\Delta \hat{B}^0}{x\sigma}\right]$$

et

$$\frac{\partial^2 u}{\partial x^2}(0,x) \simeq h^{-1}\mathbb{E}\left[\hat{Y}_{t_1}^{0,x} \frac{1}{x^2\sigma}\left(\frac{h(\Delta \hat{B}^0)^2}{\sigma} - \Delta \hat{B}^0 - \frac{1}{\sigma}\right)\right].$$

On remarque tout de suite que, pour le delta, on retombe sur l'estimation établie dans la partie 5.3.4. Pour le gamma, nous avons tenté de l'estimer sur un put européen. Malheureusement, les résultats, représentés sur la figure 5.30, ne sont pas concluant. Il pourrait néanmoins être intéressant d'étudier les techniques de réduction de variance développées dans [9] pour améliorer la vitesse de convergence de l'estimation.

[15] Le prix du sous-jacent, la volatilité ou la maturité par exemple.
[16] cf [9], partie 4.
[17] Cette quantité est appelée « gamma ».

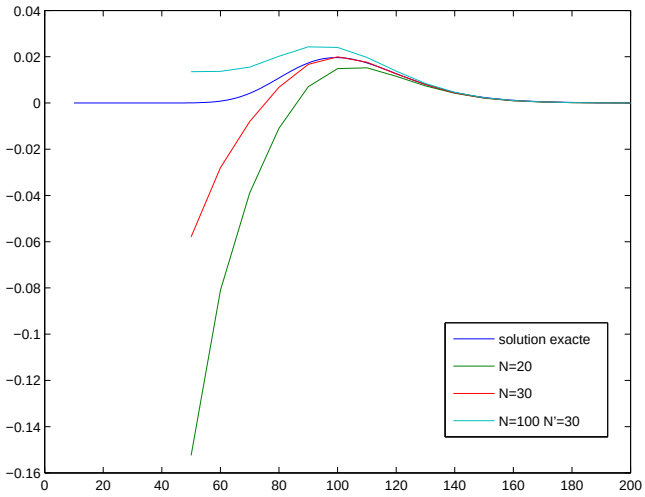

FIG. 5.30 – Estimation du gamma pour un put européen de dimension 1 en fonction du prix de l'actif sous-jacent. On a $r = 0,4$, $\sigma = 0,2$, $K = 110$, $n = 20$, $q = 10000$, $n_{iter} = 20$. La légende « $N = 100$, $N' = 30$ » signifie que $N = 100$ pour la premier pas du schéma et $N = 30$ pour les autres pas.

Conclusions

En guise de conclusion, nous nous contenterons de souligner les forces et les faiblesses du schéma étudié en traitant des solutions alternatives qui s'offrent à nous.

- Lorsqu'il n'y a pas de réflexions, que f est indépendante du contrôle et qu'elle est linéaire en y, alors on optera plutôt pour des méthodes de Monte-Carlo en appliquant la formule de Feynman-Kac. Cette approche est plus simple à mettre en œuvre, s'adapte bien aux grandes dimensions et permet d'obtenir des intervalles de confiance pour la solution.
- Dans le cas où le processus X peut s'écrire comme une fonction explicite du brownien, alors il est préférable de calculer une quantification du brownien en utilisant des méthodes de quantification marginale. Cette phase est, certes, plus longue que la phase de quantification de notre schéma, mais elle a le grand mérite d'être réalisé une bonne fois pour toute. Par exemple, cette méthode est bien adaptée pour l'évaluation d'options américaines en grande dimension lorsque les sous-jacents sont des processus de Black-Scholes.
- Pour les petites dimensions, les méthodes de résolution des équations aux dérivées partielles, telles que les différences finies ou les éléments finis, sont intéressantes car les paramètres sont plus simples à fixer grâce à l'existence, notamment, de conditions du type CFL. De plus, les erreurs sont beaucoup mieux majorées. Il conviendrait néanmoins de se pencher sur la convergence de telles méthodes lorsque f dépend du contrôle et manque de régularité.

Ainsi, le véritable enjeu des méthodes que nous avons développées dans ce rapport est la résolution d'EDSR réfléchies ou non réfléchies, pour des fonctions f, b et σ ne permettant pas d'utiliser les deux premiers points, et pour des dimensions où les méthodes classiques de résolution des équations aux dérivées partielles sont inapplicables.

Bibliographie

[1] Vlad Bally and Gilles Pagès. A quantization algorithm for solving multi-dimensional discrete-time optimal stopping problems. *Bernoulli*, 9(6) :1003–1049, 2003.

[2] Vlad Bally, Gilles Pagès, and Jacques Printems. A quantization tree method for pricing and hedging multidimensional American options. *Math. Finance*, 15(1) :119–168, 2005.

[3] F. Delarue and S. Menozzi. An interpolated stochastic algorithm for quasi-linear pdes. Mars 2006.

[4] François Delarue and Stéphane Menozzi. A forward-backward stochastic algorithm for quasi-linear PDEs. *Ann. Appl. Probab.*, 16(1) :140–184, 2006.

[5] Qiang Du, Vance Faber, and Max Gunzburger. Centroidal Voronoi tessellations : applications and algorithms. *SIAM Rev.*, 41(4) :637–676 (electronic), 1999.

[6] N. El Karoui, C. Kapoudjian, E. Pardoux, S. Peng, and M. C. Quenez. Reflected solutions of backward SDE's, and related obstacle problems for PDE's. *Ann. Probab.*, 25(2) :702–737, 1997.

[7] N. El Karoui, E. Pardoux, and M. C. Quenez. Reflected backward SDEs and American options. In *Numerical methods in finance*, Publ. Newton Inst., pages 215–231. Cambridge Univ. Press, Cambridge, 1997.

[8] N. El Karoui, S. Peng, and M. C. Quenez. Backward stochastic differential equations in finance. *Math. Finance*, 7(1) :1–71, 1997.

[9] Eric Fournié, Jean-Michel Lasry, Jérôme Lebuchoux, Pierre-Louis Lions, and Nizar Touzi. Applications of Malliavin calculus to Monte Carlo methods in finance. *Finance Stoch.*, 3(4) :391–412, 1999.

[10] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R Silverman, and A. Y. Wu. An efficient k-means clustering algorithm : Analysis and implementation. *IEEE Transactions on Patterns Analysis and Machine Intelligence*, 24(7) :881–892, 2002.

[11] I. Karatzas and S. E. Shreve. *Brownian Motion and Stochastic Calculus*. Springer, second edition, 1991.

[12] Ralf Korn and Elke Korn. *Option pricing and portfolio optimization*, volume 31 of *Graduate Studies in Mathematics*. American Mathematical So-

ciety, Providence, RI, 2001. Modern methods of financial mathematics, Translated from the 1999 German original by the authors.

[13] O. Ladyzenskaja, V. Solonnikov, and N. Ural'ceva. *Linear and Quasilinear Equations of Parabolic Type*. AMS, Providence, RI, 1968.

[14] Jin Ma, Philip Protter, and Jiong Min Yong. Solving forward-backward stochastic differential equations explicitly—a four step scheme. *Probab. Theory Related Fields*, 98(3) :339–359, 1994.

[15] Andrew Moore. An introductory tutorial on kd-trees. Technical Report Technical Report No. 209, Computer Laboratory, University of Cambridge, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1991.

[16] Gilles Pagès, Huyên Pham, and Jacques Printems. Optimal quantization methods and applications to numerical problems in finance. In *Handbook of computational and numerical methods in finance*, pages 253–297. Birkhäuser Boston, Boston, MA, 2004.

[17] Gilles Pagès and Jacques Printems. Optimal quadratic quantization for numerics : the Gaussian case. *Monte Carlo Methods Appl.*, 9(2) :135–165, 2003.

[18] É. Pardoux and S. Peng. Backward stochastic differential equations and quasilinear parabolic partial differential equations. In *Stochastic partial differential equations and their applications (Charlotte, NC, 1991)*, volume 176 of *Lecture Notes in Control and Inform. Sci.*, pages 200–217. Springer, Berlin, 1992.

[19] É. Pardoux and S. G. Peng. Adapted solution of a backward stochastic differential equation. *Systems Control Lett.*, 14(1) :55–61, 1990.

[20] Étienne Pardoux. Backward stochastic differential equations and viscosity solutions of systems of semilinear parabolic and elliptic PDEs of second order. In *Stochastic analysis and related topics, VI (Geilo, 1996)*, volume 42 of *Progr. Probab.*, pages 79–127. Birkhäuser Boston, Boston, MA, 1998.

[21] Shi Ge Peng. Probabilistic interpretation for systems of quasilinear parabolic partial differential equations. *Stochastics Stochastics Rep.*, 37(1-2) :61–74, 1991.

[22] S. Z. Selim and M. A. Ismail. K-means-type algorithms : A generalized convergence theorem and characterization of local optimality. *IEEE Transactions on Patterns Analysis and Machine Intelligence*, 6(1) :81–87, 1984.

[23] A. Shiryaev. *Probability*. Number 2nd ed. Springer, New York, 1996. MR1368405.