



Méthodes avancées en probabilités numériques

Projet : Estimation efficace du risque via des simulations séquentielles imbriquées

Isaac Cohen Sabban – Jérôme Ramassamy

Master 2 Pro Ingénierie Financière et Modèles Aléatoires (IFMA)
Février 2016

Table des Matières

I. Introduction.....	3
II. Présentation des algorithmes.....	4
1. Estimateur UNIFORM.....	4
a. Paramètres non optimaux.....	5
b. Paramètres optimaux.....	6
2. Estimateur SEQUENTIAL.....	7
3. Estimateur ADAPTATIVE.....	8
4. Estimateur ADAPTATIVE 2.....	10
5. Test des algorithmes.....	11
III. Résultats et interprétations.....	15
1. Tableaux des résultats numériques et graphiques.....	15
2. Comparaison des algorithmes.....	20
a. Analyse des résultats.....	20
b. Avantages et inconvénients.....	22
IV. Conclusion.....	24

I. Introduction

Nous analysons le problème de calcul de l'estimation du risque financier dans une simulation imbriquée. Dans cette approche, une première simulation est utilisée pour générer des scénarios financiers, et une seconde simulation est utilisée pour estimer les valeurs futures des portefeuilles pour chaque scénario généré. L'article que nous étudions se concentre sur une mesure du risque, la probabilité d'une perte importante. C'est cette probabilité que nous allons tenter d'estimer via un nouvel algorithme qui nous est proposé dans l'article « *Efficient Risk Estimation via Nested Sequential Simulation* » (par Mark Broadie, Yiping Du, Ciamac C. Moallemi).

Il faut se replacer dans le contexte. La mesure et la gestion des risques est de plus en plus importante dans les institutions financières. Un des principaux objectifs de cette mesure est de veiller à ce que les banques et autres sociétés financières aient suffisamment de réserves de capital relatives à leurs avoirs financiers et les activités d'investissement. Les échecs récents de grands et petits investissements et les banques commerciales mettent en évidence la nécessité d'une meilleure modélisation et calcul de mesures de risque. Cette mesure est généralement divisée en deux étapes: l'élaboration de scénarios et de portefeuille réévaluation. La génération de scénario se réfère à l'échantillonnage des facteurs de risque sur un horizon de temps donné. Cette première étape est souvent réalisée avec des simulations de Monte Carlo, en particulier quand des modèles plus réalistes avec un grand nombre de facteurs de risque corrélés sont utilisés. La réévaluation du portefeuille se réfère au calcul de la valeur à risque du portefeuille à un horizon donné pour un scénario particulier de facteurs de risque. Souvent, les portefeuilles contiennent des titres dérivés avec des payoffs non linéaires qui, en conjonction avec des modèles financiers plus réalistes, nécessitent des simulations de Monte Carlo. Ainsi, dans des applications réalistes, le calcul de la mesure du risque implique un à deux niveaux de simulations de Monte Carlo imbriquées.

Les simulations de Monte Carlo représentant un défi de calcul prohibitif, diverses approches sont souvent employées. L'objectif de notre étude est sur les améliorations algorithmiques de l'approche directe des simulations de Monte Carlo imbriquée, de sorte que le calcul du risque peut être fait sur les portefeuilles de titres dérivés des modèles financiers multifactorielle plus réalistes. Nous considérons ce qui est peut-être la mesure la plus fondamentale des risques : la probabilité que la valeur future du portefeuille est inférieure à un seuil prédéfini, en d'autres termes, **la probabilité d'une perte importante**. Lorsque des formules analytiques sont disponibles pour l'étape du portefeuille de réévaluation, un des défis principaux des simulations de Monte Carlo est de réduire la variance de l'estimateur des risques de simulation. Dans le cadre imbriqué, la simulation est également utilisée pour l'étape de réévaluation du portefeuille et d'autres sources de variabilité sont introduites. Le deuxième niveau de simulation introduit un biais dans le calcul, et donc à la fois biais et variance doivent être équilibrés et réduits pour minimiser l'erreur totale dans l'estimation du risque de simulation.

Pour répondre à la problématique, nous allons implémenter les algorithmes fournis dans l'article afin d'estimer la mesure de risque tout en minimisant l'erreur quadratique moyenne.

II. Présentation des Algorithmes

Plusieurs algorithmes nous sont proposés dans cet article. Nous les avons donc implémentés. Cette partie est consacrée à la description de ces algorithmes.

Tout d'abord nous allons commencer par une légère introduction des différentes procédures afin d'expliquer de ce qui les différencie les uns des autres.

Tout d'abord l'algorithme UNIFORM est un algorithme où le nombre de scénario 1 et de scénario 2 sont fixés au début du programme, puis nous faisons n simulations du scénario 1 et m simulations du scénario 2 en fonction d'un scénario 1.

Cependant, nous n'avons pas toute l'information pour les autres algorithmes.

En effet, pour le SEQUENTIAL, on connaît juste le nombre exact de scénario 1 et le nombre minimal de scénario 2 sachant le scénario 1 et le nombre moyen de scénario 2, mais en fonction d'un scénario 1 on peut avoir 100 scénarios 2 comme 10.

Le nombre de scénario 2 est donc déterminé par une formule qui sera donné plus tard dans le rapport.

Tandis que pour l'ADAPTATIVE, on connaît juste le nombre initial de scénario 1 et de scénario 2 et le but de l'algorithme est de déterminer le nombre de scénarios 1 et de scénario 2 de façons à minimiser la MSE.

1. Estimateur UNIFORM

L'estimateur uniforme de l'algorithme 1 décrit une procédure de simulation imbriquée qui combine l'estimation du premier et deuxième niveau de simulation afin de produire une estimation globale de la probabilité de perte. L'estimateur est une fonction de deux paramètres: n , le nombre d'échantillons de la première étape, et m , le nombre d'échantillons de la deuxième étape.

Protocole :

Conception d'une classe Scénario. Mis en place de l'algorithme UNIFORM. Nous permet de générer les scénarios Gaussien et Brownien géométrique. Permet de générer le scénario 1 et le scénario 2 (scénario 1 ω_i « + » scénario 2), puis d'estimer la perte conditionnellement à chaque ω_i .

Algorithme 1

Cet algorithme va nous permettre d'estimer la probabilité d'une perte importante en utilisant une simulation imbriquée uniforme. Le paramètre m est le nombre d'échantillons internes par scénario. Le paramètre n est le nombre de scénarios externes.

Dans un premier temps, nous simulons des scénarios ω_i jusqu'à un temps τ . Conditionnellement à ces scénarios ω_i , nous simulons des $\widehat{Z}_{i,j}$, qui représentent les pertes du portefeuille pour le scénario ω_i associé, puis nous calculons l'estimation des pertes \hat{L}_i pour chaque scénarios.

Enfin, nous estimons la probabilité d'une grande perte en choisissant un seuil c via la formule suivante :

$$\hat{\alpha} \sim \frac{1}{n} \sum_{i=1}^n 1_{\{\hat{L}_i \geq c\}}$$

où \hat{L}_i est donnée par :

$$\hat{L}_i \triangleq \frac{1}{m} \sum_{j=1}^m \hat{Z}_{i,j}.$$

a. Paramètres non optimaux

Dans un premier temps, nous utilisons l'estimateur « 1/3 : 2/3 UNIFORM » pour les valeurs de n et m . C'est l'estimateur UNIFORM avec $m = k^{1/3}$ et $n = k^{2/3}$. Ces valeurs ne sont pas optimales. Ces paramètres sont déterminés de manière analytique grâce au théorème et corollaire qui vont suivre.

Théorème :

Soit f la densité de la variable L (variable des pertes). Le biais asymptotique de l'estimateur satisfait

$$E[\hat{\alpha}_{m,n} - \alpha] = \frac{\theta_c}{m} + o\left(m^{-\frac{3}{2}}\right),$$

où

$$\theta_c \triangleq -Y'(c),$$

avec

$$Y(c) \triangleq \frac{1}{2} f(c) E[\sigma^2(\omega) | L(\omega) = c],$$

et $\sigma^2(\omega)$ est la variance des échantillon de scénarios 2.

Corollaire :

Sous les conditions du théorème énoncé précédemment, comme $m \rightarrow \infty$, la variance de l'estimateur UNIFORM satisfait

$$Var(\hat{\alpha}_{m,n}) = \frac{\alpha(1-\alpha)}{n} + \vartheta(m^{-1}n^{-1})$$

b. Paramètres optimaux

Les paramètres n et m doivent être choisis afin de minimiser au mieux l'erreur quadratique moyenne (MSE). Ces paramètres optimaux sont obtenus de manière théorique, ainsi, on a $m = k^{1/3}/\beta^*$ et $n = \beta^* \cdot k^{2/3}$.

Remarque : les paramètres non optimaux sont en fait $m = k^{1/3}/\beta^*$ et $n = \beta^* \cdot k^{2/3}$ avec $\beta^* = 1$.

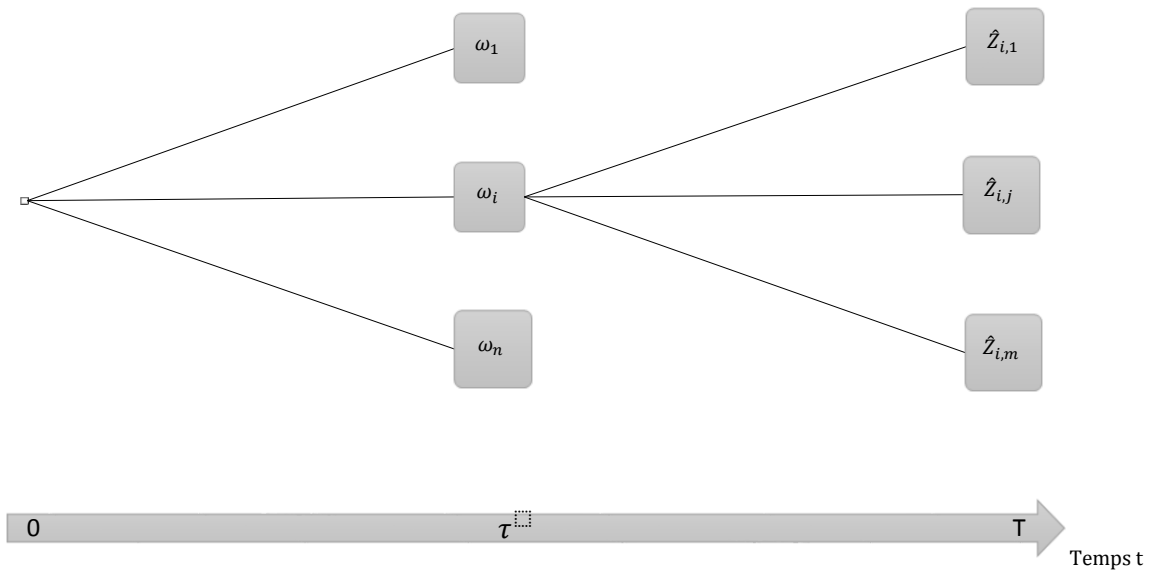


Figure 2 : Illustration de l'échantillonnage uniforme

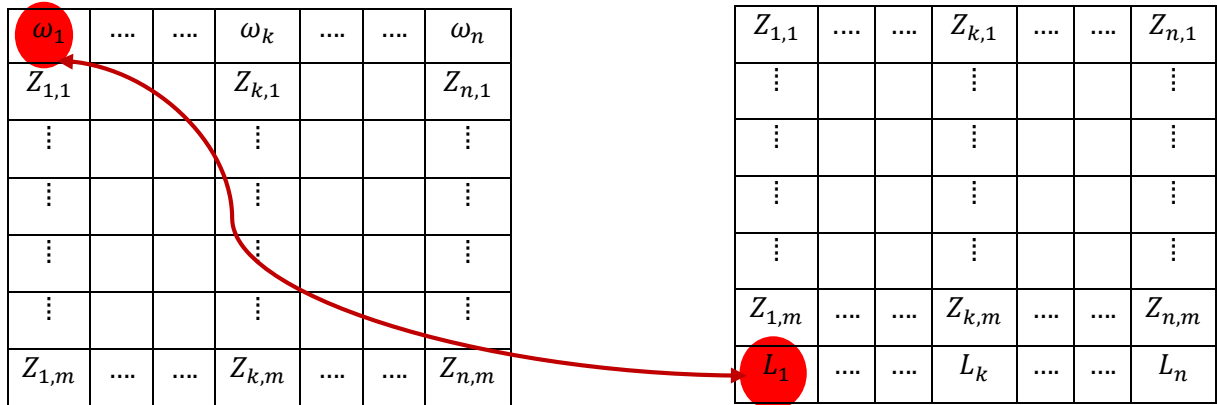


Figure 2 : Fonctionnement de l'algorithme

Ce schéma montre ce qui se passe en mémoire pour l'algorithme UNIFORM. Dans un tableau de taille $n \times m$, nous stockons les valeurs simulées des pertes de chacun des portefeuilles L_i .

2. Estimateur SEQUENTIAL

Nous allons maintenant décrire l'algorithme SEQUENTIAL.

Il s'agit d'estimer la probabilité d'une perte importante en utilisant une simulation imbriquée non uniforme séquentielle. Le paramètre m_0 est le nombre initial d'échantillons intérieurs par scénario. Le paramètre \bar{m} est le nombre moyen d'échantillons intérieurs par scénario à la fin de la simulation. Le paramètre n est le nombre de scénarios.

L'estimateur séquentiel nécessite l'accès à l'écart type conditionnel σ_i^2 des pertes dans chaque scénario i , pour calculer la marge d'erreur. Ceux-ci ne sont pas nécessaires pour l'estimateur uniforme et, de plus, ne sont généralement pas connus dans la pratique. Toutefois, ces écarts-types conditionnels peuvent être estimés dans un mode en ligne au cours de l'algorithme d'estimation.

L'algorithme 2 SEQUENTIAL :

Dans un premier temps, nous simulons des scénarios ω_i jusqu'à un temps τ et on initialise les m_i à m_0 c'est à dire 2. Les m_i représentent le nombre de scénario 2 simulé pour un scénario ω_i .

Puis, tant que la somme des $m_i < n * \bar{m}$, on va simuler un scénario 2 pour ω_i .

Le ω_i en question sera déterminé par l'argmin des $m_i * \text{abs}(L_i - c) / \sigma_i$, cette quantité représente l'erreur de marge associé à un scénario ω_i . Elle est trouvée par l'inégalité de Chebyshev et est censée maximiser $P(L_i \geq c)$.

L'argmin lui représente l'indice du tableau pour lequel le minimum est atteint.

On stocke donc dans un tableau les valeurs $m_i * \text{abs}(L_i - c) / \sigma_i$ associé à chaque ω_i . Puis on cherche l'argmin de ce tableau et au ω_i associé on lui génère un deuxième scénario et on met à jour les $L_i, m_i = m_i + 1, Z_i$ et donc aussi $m_i * \text{abs}(L_i - c) / \sigma_i$, et on continue jusqu'à ce que la condition d'arrêt soit atteinte.

Puis on calcule alpha chapeau.

3. Estimateur ADAPTATIVE

Nous allons maintenant décrire l'estimateur ADAPTATIVE.

La simulation est initialisée en générant des scénarios avec n_0 avec m_0 échantillons internes pour chaque scénario.

Le budget de travail de la simulation k est divisé en intervalles $K \triangleq k / \tau_e$ (ou époques) de longueur τ_e (il est supposé, pour simplifier l'exposé, que K est intégrable et que le premier intervalle est seulement de longueur $\tau_e - n_0 m_0$ en raison de l'initialisation).

Au début du l -ème intervalle, les estimations sont faites pour le carré et la variance de l'estimation de perte de probabilité, compte tenu des scénarios et des échantillons qui ont été générés à ce moment-là. Plus précisément, compte tenu de l'estimation de perte de probabilité

$$\hat{\alpha} = \frac{1}{n} \sum_{i=1}^n 1_{\{\hat{L}_i \geq c\}},$$

le biais est approché par

$$E[\hat{\alpha} - \alpha] \approx \hat{B} \triangleq \hat{\alpha} - \bar{\alpha},$$

où

$$\bar{\alpha} \triangleq \frac{1}{n} \sum_{i=1}^n \Phi \left(\frac{m_i (\hat{L}_i - c)}{\sigma_i} \right).$$

Cette approximation est basée sur un théorème limite heuristique central : dans chaque scénario ω_i , lorsque le nombre d'échantillons m_i est grand, chaque estimation de la perte \hat{L}_i peut être approchée par une distribution normale de moyenne $L(\omega_i)$ et de variance σ_i^2 / m_i . Par conséquent, étant donné un ensemble fixe de scénarios $\omega_1, \dots, \omega_n$, on peut estimer le biais via

$$E[\hat{\alpha} - \alpha] = \frac{1}{n} \sum_{i=1}^n P(\hat{L}_i \geq c) - 1_{\{L(\omega_i) \geq c\}}$$

$$E[\hat{\alpha} - \alpha] \approx \frac{1}{n} \sum_{i=1}^n \left\{ \Phi \left(\frac{m_i(\hat{L}_i - c)}{\sigma_i} \right) - 1_{\{L(\omega_i) \geq c\}} \right\}$$

Parce que chaque véritable perte $L(\omega_i)$ est inconnue dans la pratique, nous pouvons l'approximer avec son estimation réalisée \hat{L}_i . Cela se traduit par (20). En faisant une approximation heuristique similaire pour la variance, nous arrivons à l'expression

$$Var(\hat{\alpha}) \approx \hat{V} \triangleq \frac{\bar{\alpha}(1 - \bar{\alpha})}{n}$$

Il faut noter que les estimateurs du biais et de la variance sont destinés uniquement comme heuristiques. De meilleurs estimateurs peuvent être possibles, et le biais en particulier est notoirement difficile à estimer. Pour nos besoins, cependant, ils ne doivent être exacts dans des ordres de grandeur, afin d'allouer l'effort de calcul entre les échantillons internes et externes scénarios.

$$n' = \min \left\{ \max \left\{ \left(\left(\frac{\hat{V}n}{4\hat{B}^2\bar{m}^4} (\bar{m}n + \tau_e)^4 \right)^{\frac{1}{5}} \right), n \right\}, n + \tau_e \right\},$$

et

$$\bar{m}' = \frac{\bar{m}n + \tau_e}{n'},$$

sont trouvés comme solution du problème d'optimisation qui consiste à minimiser $B2(\bar{m}\bar{m}')^4 + V(nn')$

Minimiser ce problème, revient à minimiser la MSE en fonction du nombre de scénario 1 et du nombre de scénario 2 à un scénario 1 associé.

4. Estimateur ADAPTATIVE 2

C'est le même que le précédent sauf que la variance n'est pas constante. En effet, il s'agit d'une variante de l'ADAPTATIVE. Dans cette variante, chaque σ_i est estimé « en temps réel ». Une telle variation σ_i remplace chaque dans l'Algorithme ADAPTATIVE avec l'estimation

$$\hat{\sigma}_i = \frac{m_i}{m_i + b} \tilde{\sigma}_i + \frac{b}{m_i + b} \bar{\sigma},$$

avec

$$\tilde{\sigma}_i \triangleq \left[\frac{1}{m_i - 1} \sum_{j=1}^m (\widehat{Z}_{i,j} - \widehat{L}_i)^2 \right]^{\frac{1}{2}},$$

qui correspond à la *standard deviation* des scénarios 2 pour un ω_i et

$$\bar{\sigma} \triangleq \frac{1}{n} \sum_{i=1}^n \tilde{\sigma}_i,$$

qui est la moyenne globale de l'ensemble de écarts-types de l'échantillon.

Malheureusement, nous n'avons pas réussi à obtenir de bons résultats pour ces algorithmes. Par ailleurs, l'ADAPTATIVE 2 dépendant de l'ADAPTATIVE 1, nous ne l'avons même pas codé.

Nous avons donc essayé plusieurs solutions. Au début, nous nous sommes dit que Φ était la fonction de répartition de la loi Normale centrée réduite, mais nous n'avons pas obtenu le résultat voulu, puis étant donnée que Φ correspond à $P(\widehat{L}_i \geq c)$ par définition de la fonction de répartition, nous avons essayé $1 - \Phi$, mais aussi (ce qui est équivalent avec une loi normal) $\Phi\left(-\frac{m_i(\widehat{L}_i - c)}{\sigma_i}\right)$, cela ne marchait toujours pas, nous avons aussi essayé avec la densité de la loi normal centrée réduite, mais toujours pas le bon résultats.

Puis, vu que le biais est aussi défini comme ceci

$$E[\hat{\alpha} - \alpha] \approx \frac{1}{n} \sum_{i=1}^n \left\{ \Phi\left(\frac{m_i(\widehat{L}_i - c)}{\sigma_i}\right) - 1_{\{L(\omega_i) \geq c\}} \right\},$$

nous avons donc essayé $\hat{B} \triangleq \bar{\alpha} - \hat{\alpha}$ au lieu de $\hat{B} \triangleq \hat{\alpha} - \bar{\alpha}$, mais nous n'avons malheureusement toujours pas obtenu le bon résultat.

On a donc essayé avec la formule de la Variance définie pour les algorithmes UNIFORM et SEQUENTIAL, mais ce fut toujours sans réussite.

Nous avons pourtant codé une fonction de réallocation de mémoire sans utilisé la classe « vector », mais le problème ne venait pas de là, car même avec celle-ci nous n'obtenions pas les bonnes valeurs de n et de m .

5. Test des algorithmes

Pour tester nos algorithmes, nous considérons un portefeuille avec des flux de trésorerie de type Gaussien, où les deux scénarios de la scène extérieure et des échantillons de scène internes sont générées à partir des distributions normales, et un exemple d'option de vente (put), où le portefeuille est constitué d'une seule vente l'option sur un actif sous-jacent dont le prix suit un processus de mouvement brownien géométrique. Pour ces deux exemples, nous sommes intéressés par le calcul de la probabilité d'une perte.

Cas Gaussien :

Dans l'exemple gaussien, nous considérons un portefeuille avec des facteurs et des flux de trésorerie (cashflows) de risque normalement distribués. Ceci est le paramètre le plus simple permettant de tester une procédure de simulation imbriquée.

Plus précisément, nous considérons un portefeuille d'une valeur $X_0 = 0$ en $t=0$ et $X_\tau(\omega) = \omega$ à l'horizon de risque τ . Il est supposé que le facteur de risque réel estimé ω suit une loi normale $N(0,1)$. La perte $L(\omega) = X_0 - X_\tau(\omega) = -\omega$ est également une variable aléatoire suivant une loi normale standard.

Pour un scénario ω_i , chaque échantillon de perte est de la forme $\widehat{Z}_{i,j} = -\omega_i + \sigma_{inner}W_{i,j}$, où $W_{i,j}$ est une loi normal standard.

Dans ce cas, compte tenu de la perte seuil c , la probabilité d'une perte excédant c est donnée par $\alpha = \Phi(-c)$. Nous choisissons, comme dans l'article, les valeurs 1.282, 2.326 et 3.090 pour le seuil de perte c , correspondant à des probabilités de perte de 10 %, 1 % et 0,1 %, respectivement.

Option PUT :

Dans l'exemple d'option de vente, nous supposons que Le portefeuille est constitué d'une position longue sur une seule option de vente. Cet exemple est plus complexe parce que les flux de trésorerie du portefeuille ne sont pas linéaires et suivent des distributions très asymétriques, qui varient selon les scénarios de formation sensiblement extérieures.

Ici, l'actif sous-jacent suit un mouvement brownien géométrique avec un prix initial de $S_0 = 100$.

La *drift* de ce processus, dans la Probabilité monde réel, utilisé dans l'étape extérieure c'est-à-dire le scénario 1 de la simulation est $\mu = 8 \%$.

La volatilité annualisée est $\sigma = 20 \%$. Le taux sans risque est $r = 3 \%$.

Le *Strike* de l'option de vente est $K = 95$, et la maturité est $T = 0,25 \text{ ans}$ (à savoir trois mois). L'horizon de risque est $\tau = \frac{1}{52} \text{ ans}$ (à savoir une semaine).

Avec ces paramètres, la valeur initiale de l'option PUT est $X_0 = 1,669$ (donnée par la formule de Black-Scholes).

Nous considérons $S_\tau(\omega)$ le prix de l'actif sous-jacent à l'horizon de risque. Cette variable aléatoire est générée par $S_\tau(\omega) \triangleq S_0 e^{\left(r - \frac{\sigma^2}{2}\right)\tau + \sigma\sqrt{\tau}\omega}$ où le facteur de risque réel ω est une variable aléatoire normale standard.

La perte du portefeuille à l'horizon de risque est donnée par

$$L(\omega) = E\left[e^{-r(T-\tau)} \max(K - S_T(\omega, W), 0) \mid \omega\right],$$

avec W étant indépendamment et identiquement distribuée (iid) et suivant une loi normale standard et $S_T(\omega, W)$ donné par l'expression suivante :

$$S_T(\omega, W) \triangleq S_\tau(\omega) e^{\left(r - \frac{\sigma^2}{2}\right)(T-\tau) + \sigma\sqrt{T-\tau}W}.$$

Notons que, compte tenu de la valeur fixe et une variable aléatoire W de loi normale standard, la variable aléatoire $S_T(\omega, W)$ est distribué selon la probabilité risque neutre. Compte tenu d'un scénario ω_i , chaque échantillon de perte est donnée par

$$\widehat{Z}_{i,j} = X_0 - e^{-r(T-\tau)} \max(K - S_T(\omega_i, W_{i,j}), 0),$$

$W_{i,j}$: Variable aléatoire tel que $W_{i,j} \sim N(0,1)$;

T : Maturité de l'option.

Il faut noter que les scénarios 1 (externes) sont générés en utilisant la probabilité monde réelle régi par le *drift* μ , alors que les scénarios 2 (internes) utilisés pour générer des futurs prix des options de vente sont générés en utilisant la probabilité risque-neutre régi par le *drift* r .

Nous pouvons voir que la perte $L(\omega)$ est strictement croissante en le facteur de risque. Par conséquent, la probabilité d'une perte supérieure à un seuil c peut être calculé comme suit :

$$\alpha = P(L \geq c) = P(\omega \geq \omega^*) = \Phi(-\omega^*),$$

où ω^* est l'unique solution de $L(\omega^*) = c$.

Nous prenons les valeurs (les mêmes que dans l'article) 0.859, 1.221 et 1.390 pour le seuil de perte c , correspondant respectivement à des probabilités de perte de 10 %, 1 % et 0.1 %.

Nous allons à présent donner les formules de biais, de variance et de MSE que nous utilisons dans chaque algorithme :

Pour l'algorithme 1 :

Variance :

$$Var(\hat{\alpha}_{m,n}) = \frac{\alpha(1-\alpha)}{n} + \vartheta(m^{-1}n^{-1})$$

Biais :

$$Biais^2 = (\hat{\alpha}_{m,n} - \alpha)^2$$

Preuve :

$$(E[\hat{\alpha}_{m,n} - \alpha])^2 = \left(E\left[\frac{1}{n}\sum_{i=1}^n 1_{\{\hat{L}_i \geq c\}} - \alpha\right]\right)^2 \text{ par définition de } \hat{\alpha}_{m,n}.$$

Les \hat{L}_i étant i.i.d, on a donc

$$\left(\frac{1}{n}\sum_{i=1}^n E[1_{\{\hat{L}_1 \geq c\}}] - \alpha\right)^2 = (E[1_{\{\hat{L}_1 \geq c\}}] - \alpha)^2$$

$$= (P[\hat{L}_1 \geq c] - \alpha)^2 = (\hat{\alpha}_{m,n} - \alpha)^2.$$

Erreur Quadratique Moyenne (MSE):

$$E[\hat{\alpha}_{m,n} - \alpha] = E\left[\left(\hat{\alpha}_{m,n} - E[\hat{\alpha}_{m,n}]\right)^2\right] + (E[\hat{\alpha}_{m,n} - \alpha])^2$$

Pour l'algorithme 2 :

Variance :

$$Var(\hat{\alpha}_{m,n}) = \frac{\alpha(1-\alpha)}{n} + \vartheta(m^{-1}n^{-1})$$

Biais :

$$Biais^2 = (\hat{\alpha}_{m,n} - \alpha)^2$$

Les formules de la variance, du biais de l'algorithme ADAPTATIVE ont été énoncées précédemment, lors de la description de ce dernier.

III. Résultats et interprétations

1. Tableaux des résultats et graphiques

Cette partie est destinée à exposer nos résultats numériques obtenus après test de nos algorithmes. Nous allons ainsi présenter ces résultats sous forme de 6 tableaux, un pour chaque α .

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$1,61363.10^{-4}$	$3.96845.10^{-6}$	$1,65332.10^{-4}$	11,2703
Optimal Uniform	4499	889	$1,09683.10^{-5}$	$1,94131.10^{-5}$	$3,03814.10^{-5}$	9,866882
Adaptative (σ_i)	14968	281	-	-	-	-
Adaptative	12802	321	-	-	-	-
Optimal Sequential	12395	323	$1,77205.10^{-6}$	$7,34677.10^{-6}$	$9,11881.10^{-6}$	10,1331

Figure 3 : Tableau des résultats numériques pour le cas Gaussien 10%

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$1,48206.10^{-5}$	$5,42003.10^{-7}$	$1,53626.10^{-5}$	1,38498
Optimal Uniform	5089	786	$2,53967.10^{-6}$	$2,25176.10^{-6}$	$4,791443.10^{-6}$	1,08177
Adaptative (σ_i)	16177	250	-	-	-	-
Adaptative	16118	251	-	-	-	-
Optimal Sequential	30860	130	$1,0353.10^{-6}$	$3,53082.10^{-7}$	$1,38839.10^{-6}$	1,10175

Figure 4 : Tableau des résultats numériques pour le cas Gaussien 1%

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$4,99025.10^{-8}$	$6,76021.10^{-8}$	$5,66627.10^{-7}$	0,170642
Optimal Uniform	7788	514	$2,42188.10^{-8}$	$1,48214.10^{-7}$	$1,72433.10^{-7}$	0,115562
Adaptative (σ_i)	30798	132	-	-	-	-
Adaptative	30628	132	-	-	-	-
Optimal Sequential	56686	71	$1,24067.10^{-8}$	$1,95842.10^{-8}$	$3,19909.10^{-8}$	0,111139

Figure 5 : Tableau des résultats numériques pour le cas Gaussien 0.1%

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$5,37289.10^{-4}$	$4,28613.10^{-6}$	$5,41575.10^{-7}$	12,3179
Optimal Uniform	5095	785	$3,47663.10^{-6}$	$1,73709.10^{-5}$	$2,08476.10^{-5}$	9,81354
Adaptative (σ_i)	6671	601	-	-	-	-
Adaptative	7325	547	-	-	-	-
Optimal Sequential	12395	323	$7,97339.10^{-8}$	$7,27921.10^{-6}$	$7,35894.10^{-6}$	10,0282

Figure 6 : Tableau des résultats numériques pour le cas PUT 10%

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$6,55317.10^{-3}$	$7,41519.10^{-7}$	$6,55391.10^{-3}$	1,90484
Optimal Uniform	3143	1273	$6,68617.10^{-7}$	$3,4046.10^{-6}$	$4,07322.10^{-6}$	1,08177
Adaptative (σ_i)	10085	401	-	-	-	-
Adaptative	9992	405	-	-	-	-
Optimal Sequential	19558	205	$8,87014.10^{-7}$	$5,53333.10^{-7}$	$1,44035.10^{-6}$	1,09418

Figure 7 : Tableau des résultats numériques pour le cas PUT 1%

Algorithmes	n	\bar{m}	Biais	Variance	MSE	α
1/3 :2/3 Uniform	25199	159	$5,63184.10^{-6}$	$1,33409.10^{-7}$	$5,76525.10^{-6}$	0,337315
Optimal Uniform	2570	1556	$2,79944.10^{-8}$	$4,53678.10^{-7}$	$4,81672.10^{-7}$	0,115562
Adaptative (σ_i)	14884	274	-	-	-	-
Adaptative	14364	284	-	-	-	-
Optimal Sequential	26508	151	$3,6726.10^{-10}$	$3,69652.10^{-8}$	$3,73325.10^{-8}$	0.0980836

Figure 8 : Tableau des résultats numériques pour le cas PUT 0.1%

Algorithmes	1/3 :2/3 Uniform	Optimal Uniform	Adaptative (σ_i)	Adaptative	Optimal Sequential
α gaussien en %	10	10	10	10	10
Temps d'exécution des algorithmes en secondes	0.541878	0.585131	-	-	341.261
α gaussien en %	1	1	1	1	1
Temps d'exécution des algorithmes en secondes	0.550039	0.58544	-	-	847.789
α gaussien en %	0.1	0.1	0.1	0.1	0.1
Temps d'exécution des algorithmes en secondes	0.571277	0.623717	-	-	1609.26
α put en %	10	10	10	10	10
Temps d'exécution en secondes des algorithmes	0.689161	0.7142432	-	-	337.08
α put en %	1	1	1	1	1
Temps d'exécution des algorithmes en secondes	0.693407	0.734798	-	-	590.723
α put en %	0.1	0.1	0.1	0.1	0.1
Temps d'exécution des algorithmes en secondes	0.761319	0.829921	-	-	734.332

Figure 9 : Tableau des temps d'exécutions de chaque algorithme

Graphiques des pertes :

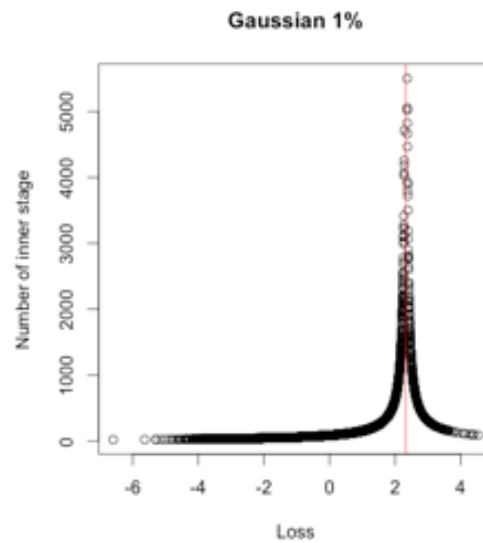


Figure 10 : Distribution des Pertes pour le cas Gaussien 1%

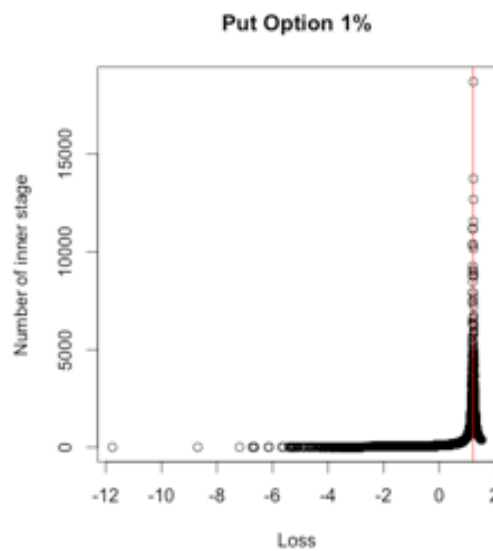


Figure 11 : Distribution des pertes pour le cas de l'Option PUT 1%

Cette figure montre le nombre d'échantillons de scénario 2 en fonction de la perte dans chaque scénario, en moyenne plus de 1000 essais. Le nombre de scénario 2 c'est une fonction de la perte de chaque scénario. Ici, $k = 4\,000\,000$ ($n \cdot m$) qui représente le nombre de chemins total. Là où le max est atteint cela correspond au c ($L(\omega^*) = c$) (cf page 13).

2. Comparaison des algorithmes

a. Analyse des résultats

Nous allons d'abord faire quelques remarques :

- Les cas gaussien sont exécutés pour tester les algorithmes ;
- Les résultats manquant sont ceux de l'adaptive. Nous ne les avons pas mis car les valeurs obtenues sont fausses. Nous avons émis quelques hypothèses sur les possibles facteurs de défaillance de notre algorithme que nous avons détaillé précédemment dans cet exposé ;

Tableau 1 (Figure 3)

L'estimateur UNIFORM « optimal » est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-4}) et la variance (ordre 10^{-6}), pour le cas gaussien 10%, de l'Uniform 1/3:2/3 sont plus grand que ceux de l'optimal (ordre 10^{-5} , ordre 10^{-5} respectivement). On a donc moins de variabilité et par conséquent une MSE plus faible.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Tableau 2 (Figure 4)

L'estimateur UNIFORM « optimal » est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-5}) et la variance (ordre 10^{-7}), pour le cas gaussien 1%, de l'Uniform « 1/3:2/3 » sont plus grand que ceux de l'optimal (ordre 10^{-6} , ordre 10^{-6} respectivement). On a donc moins de variabilité et par conséquent une MSE plus faible.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Tableau 3 (Figure 5)

L'estimateur UNIFORM « optimal » est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-8}) et la variance (ordre 10^{-8}), pour le cas gaussien 0,1%, de l'Uniform « 1/3:2/3 » sont plus grand que ceux de l'optimal (ordre 10^{-8} , ordre 10^{-7} respectivement). On a donc moins de variabilité.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur

UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Tableau 4 (Figure 6)

L'estimateur UNIFORM optimal est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-4}) et la variance (ordre 10^{-6}), pour le cas PUT 10%, de l'Uniform « 1/3:2/3 » sont plus grand que ceux de l'optimal (ordre 10^{-6} , ordre 10^{-5} respectivement). On a donc moins de variabilité.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Tableau 5 (Figure 7)

L'estimateur UNIFORM « optimal » est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-3}) et la variance (ordre 10^{-7}), pour le cas gaussien 1%, de l'Uniform 1/3:2/3 sont plus grand que ceux de l'optimal (ordre 10^{-7} , ordre 10^{-6} respectivement). On a donc moins de variabilité.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Tableau 6 (Figure 8)

L'estimateur UNIFORM optimal est plus précis que le « 1/3:2/3 ». En effet, on peut voir que le biais (ordre 10^{-6}) et la variance (ordre 10^{-7}), pour le cas gaussien 0,1%, de l'Uniform « 1/3:2/3 » sont plus grand que ceux de l'optimal (ordre 10^{-8} , ordre 10^{-7} respectivement). On a donc moins de variabilité.

Le SEQUENTIAL offre une meilleure approximation de la probabilité de perte. En effet, on constate que le biais et la variance sont bien plus réduits par rapport à l'estimateur UNIFORM. Cela implique également que l'erreur quadratique moyenne est elle aussi fortement réduite.

Ce qu'il ressort de l'analyse de nos tableaux de résultats c'est que le SEQUENTIAL nous fournit une meilleure approximation de la probabilité des pertes α tout en réduisant la variance, le biais et l'erreur quadratique moyenne.

b. Avantages et inconvénients

UNIFORM

D'un point de vue purement algorithmique, c'est le plus facile à implémenter. Cependant pour qu'il soit efficace il faut déterminer les paramètres optimaux. En effet, en considérant un $\beta^* = 1$ on a $m = k^{1/3}$ et $n = k^{2/3}$. Ce sont des valeurs « naïves » qui ne minimisent pas la variance et le biais, et par conséquent l'erreur quadratique moyenne (MSE).

Les valeurs optimales sont obtenu de manière théorique, ainsi, on a $m = k^{1/3}/\beta^*$ et $n = \beta^* \cdot k^{2/3}$.

Dans ce projet, on considère donc deux estimateur UNIFORM : « le 1/3 : 2/3 » et l' « optimal ».

Ce sont deux estimateurs uniformes asymptotiquement optimaux ; ils ne diffèrent que par le choix de la constante β^* . L'exécution pratique de ces deux estimateurs est cependant très différente. En effet, après avoir implémenté et tester l'algorithme, les résultats mettent en évidence la sensibilité de l'estimateur uniforme, dans la pratique, au choix de la constante. A noter que le calcul de la constante β^* ,

$$\beta^* \triangleq \left(\frac{\alpha(1-\alpha)}{2\theta_c^2} \right)^{\frac{1}{3}},$$

nécessite la connaissance de la constante θ_c défini précédemment. Dans la pratique, on ne sait pas comment estimer cette constante, et cette dernière peut varier considérablement entre les différentes instances de problèmes.

SEQUENTIAL

En outre, l'estimateur séquentiel nécessite temps de calcul supplémentaire au-delà de celle de l'estimateur uniforme, la seule exigence supplémentaire étant de suivre les scénarios de l'ordre de la marge d'erreur. Ceci peut être accompli de manière efficace par l'intermédiaire d'une structure de données de file d'attente de priorité.

UNIFORM & SEQUENTIAL

Ceux-ci représentent la meilleure performance possible qui peut être atteint par les estimateurs uniformes et séquentiels. Cependant, aucun de ces estimateurs est mis en œuvre dans la pratique, le premier parce qu'il dépend d'un paramètre qui ne peut pas être facilement déterminée à partir des données du problème, le second parce qu'il nécessite l'exploration sur le choix des paramètres. Mais, en les opposants, nous pouvons voir une

comparaison uniforme et un échantillonnage non uniforme sur un pied d'égalité. Cette comparaison illustre clairement les avantages de l'échantillonnage non uniforme. Dans tous les cas de test, l'estimateur séquentiel optimal a le MSE le plus bas. L'amélioration MSE par rapport à l'estimateur uniforme optimale se situe entre un facteur de 4 et 10. Cette amélioration est le plus grand lors de l'estimation des probabilités de perte qui sont rares.

Notons que l'estimation séquentielle optimale emploie de nombreux échantillons de scénario 1 et beaucoup de scénarios 2. Ceci est en accord avec la théorie développée dans l'article et les expériences réalisées. L'estimateur séquentiel optimal est en mesure d'atteindre un faible biais avec moins d'échantillons de scénarios interne, d'où le fait qu'il peut employer plusieurs scénarios avec le même budget de calcul.

Problème pour les algorithmes SEQUENTIAL et ADAPTATIVE

Contrairement à l'algorithme UNIFORME, dans le SEQUENTIAL nous ne connaissons pas le nombre de scénario 2 que nous allons simuler tandis que pour l'ADAPTATIVE nous ne connaissons ni le nombre de scénario 1 ni le du nombre de scénario 2 finaux que nous allons avoir.

En effet, le fait que ces algorithmes présente des Biais et des Variances faibles (et donc une MSE faible) résident dans le fait que la génération d'un scénario 2 (ou 1 et 2 dans le cas ADAPTATIVE) vient d'une condition permettant justement de minimiser la MSE.

IV. Conclusion

La simulation imbriquée à deux niveaux peut fournir une évaluation plus réaliste du risque financier, mais avec un coût de calcul considérable. Dans cet article, on nous propose une procédure de simulation séquentielle imbriquée qui réduit considérablement la charge de calcul.

Les économies sont réalisées à l'aide d'une procédure d'échantillonnage interne non uniforme qui alloue plus de ressources là où l'effet sur l'estimation du risque est le plus grand, ce qui permet de consacrer relativement plus d'efforts à la génération de scénarios externes.

Cependant, n'avons pas pu fournir les résultats pour l'algorithme ADAPTATIVE car nous avons rencontrés un problème que nous n'avons pas pu identifier ce qui a abouti à des résultats faux. Nous avons donc préféré ne pas les mettre dans cet exposé.

L'effet combiné produit un estimateur de risque qui converge à un rythme plus rapide de la valeur réelle. Dans les expériences numériques, l'erreur quadratique moyenne a été réduite par des facteurs allant de 4 à plus de 100.

La procédure d'estimation séquentielle peut être combinée à des recherches antérieures sur la réduction de la variance pour la génération de scénarios 1 (externe) pour réaliser des économies supplémentaires de calcul.

Après exécution de nos algorithmes, les résultats que nous avons obtenus sont proches de ceux fournis par l'article.

Les algorithmes et les résultats ont été présentés dans le cadre de l'estimation de la probabilité d'une perte importante, mais il peut être possible d'appliquer des idées similaires pour développer des algorithmes non uniformes pour d'autres mesures de risque.