Option Pricing

Simulation et Monte Carlo

Suzie Grondin, Thomas Aujoux, Elena Loumagne

25/04/2023

1. Présentation du Pricing Asian
2. Monte Carlo classique
3. Multi Level Monte Carlo
4. Quasi Monte Carlo
5. Résultats et conclusion
6. Question bonus

- But : évaluer le prix de l'Asian Option entre $t_0 = 0$ et $t_k = T$

$$C = \mathbb{E}(\mathrm{e}^{-rT}(\frac{1}{k}(\sum_{i=1}^{k} S(t_i) - K)^+))$$

- On considère un modèle CIR pour modéliser le processus S_t :

$$dS_t = \alpha(b - S_t)dt + \sigma\sqrt{S_t}dW_t \text{ avec } W_t \sim \mathcal{N}(0, dt)$$

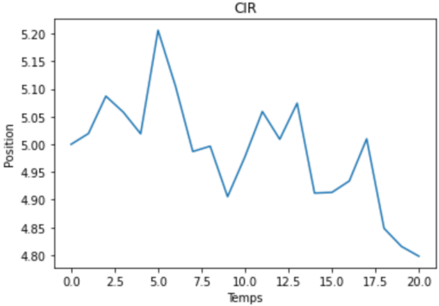
$$\text{et } \alpha = 0.15, b = 0, \sigma = 0.2, T = 1, r = 0.05, K = 4,$$
$$k = 20, t_i = i/20$$

Utilisation de trois méthodes :

1. Monte Carlo standard
2. Multi-level Monte Carlo
3. Quasi-Monte Carlo

Comparaison des méthodes en utilisant le CPU time et les MSE.

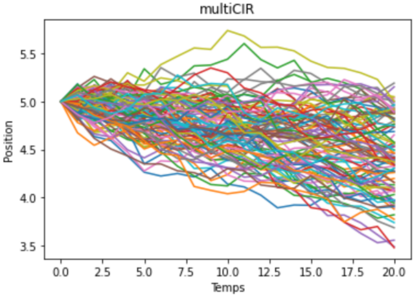
1. **Première étape :** modéliser le processus CIR S_t avec un pas de discrétisation de $\frac{1}{k} := \frac{1}{20}$ pour réaliser un total de k simulations au cours du temps T



2. **Deuxième étape :** effectuer la moyenne des k simulations de S sur les différents pas de temps

$$\frac{1}{k}\sum_{i=1}^{k} S(t_i)$$

3. **Troisième étape :** Simuler N échantillons de différents CIR, avec toujours le même pas de discrétisation



4. **Quatrième étape :** pour chaque échantillon, on calcule le payoff et la present value :

$$payoff = max(0, \frac{1}{k}(\sum_{i=1}^{k} S(t_i) - K))$$

$$PresentValue = \mathrm{e}^{-rT} * payoff$$

Rappel : on veut calculer
$C = \mathbb{E}(\mathrm{e}^{-rT}(\frac{1}{k}(\sum_{i=1}^{k} S(t_i) - K)^{+})) = \mathbb{E}(PresentValue)$

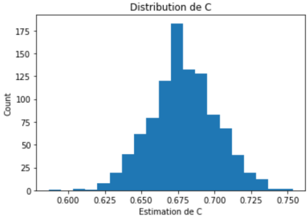
5. **Cinquième étape :** Estimation de l'Asian Pricing

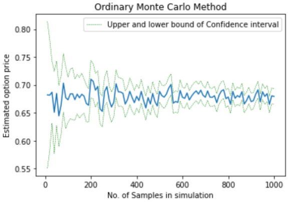
   On effectue la moyenne des Present Value sur les N échantillons :

$$\hat{C} = \frac{1}{N}\sum_{i=1}^{N} PresentValue_i$$

En effectuant plusieurs simulations, on trouve environ :

$$\hat{C} \approx 0.66$$





On note à partir de maintenant:

- $\hat{P}_l$ la present value de l'action simulée par un modèle CIR avec un pas de discrétisation 2^{-l}
- N_l le nombre d'échantillons au niveau l

- Pour un Monte Carlo de niveau L, on veut estimer Y par:

$$\hat{Y} = \sum_{l=0}^{L} \hat{Y}_l$$

avec :

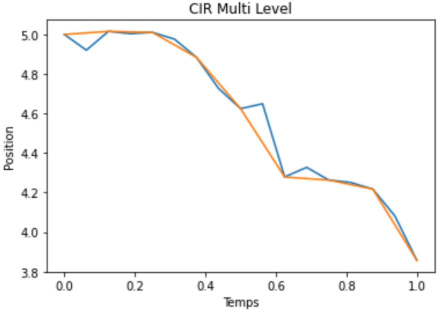
$$\forall l > 0, \quad \hat{Y}_l = N_l^{-1} \sum_{i=1}^{N_l} (\hat{P}_l^{(i)} - \hat{P}_{l-1}^{(i)})$$

et

$$\hat{Y}_0 = N_0^{-1} \sum_{i=1}^{N_0} \hat{P}_0^{(i)}$$

- Justification : Somme télescopique $\mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{l=1}^{L} \mathbb{E}[P_l - P_{l-1}]$

- **Point clé** : la quantité $\hat{P}_l^{(i)} - \hat{P}_{l-1}^{(i)}$ est simulée grâce à deux approximations discrètes ayant des pas de discrétisation différents $(2^{-l}$ et $2^{-(l-1)})$ mais ayant le même mouvement Brownien



- A ce stade, on peut déjà obtenir une estimation pour des tailles d'échantions N_l fixés et L fixé
- Pour que la méthode Multi Level fonctionne de manière optimale, il faut l'appliquer L et N_l optimaux
- Comment trouver L et N_l optimaux ?

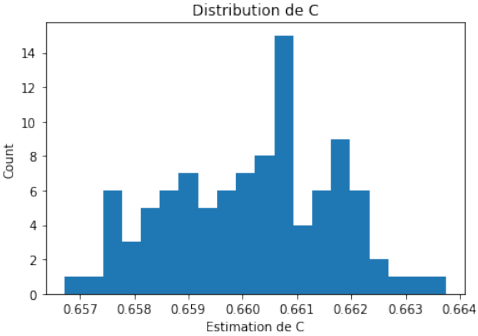
- Voici l'algorithme pour trouver le L optimal et les N_l optimaux:

1. Commencer à L = 0
2. Estimer V_L la variance de $\hat{P}_L^{(i)} - \hat{P}_{L-1}^{(i)}$ (ou $\hat{P}_0^{(i)}$) pour $N_L = 10^4$
3. Calculer la taille des échantillons optimaux pour l = 0, ..., L donnée par $N_l = \lceil 2\epsilon^{-2}\sqrt{V_l h_l}\sum_{l'=1}^{L}\sqrt{V_{l'}/h_{l'}x}\rceil$, h_l représentant le pas de discrétisation
4. Pour chaque l, si le nouveau N_l calculé est supérieur à l'ancien, ajouter la difference entre ces deux échantillons simulations
5. Si $L \geq 2$, tester la condition de convergence suivante : $\hat{Y}_L - \frac{1}{2}\hat{Y}_{L-1} \leq \frac{1}{\sqrt{2}}(2^2 - 1)\epsilon$
6. Si $L < 2$ ou la condition de convergence n'est pas respectée, $L = L + 1$ et retourner à l'étape 2.

Résultats obtenus en fixant $\epsilon = 10^{-3}$

$$\hat{Y} \approx 0.66 \approx \hat{C}$$

L varie entre 3 *et* 4

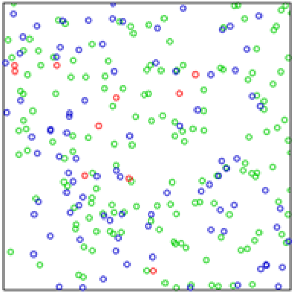


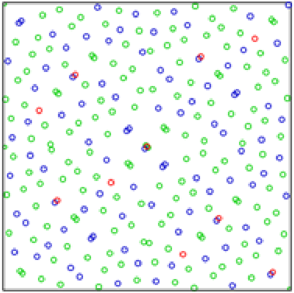
- **Principe de la méthode Quasi Monte Carlo**

Générer le CIR :

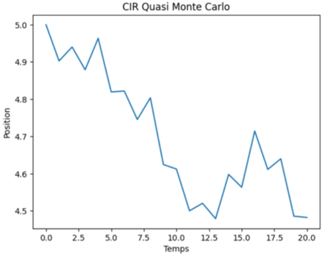
$$dS_t = \alpha(b - S_t)dt + \sigma\sqrt{S_t}dW_t \text{ avec } W_t \sim \mathcal{N}(0, dt)$$

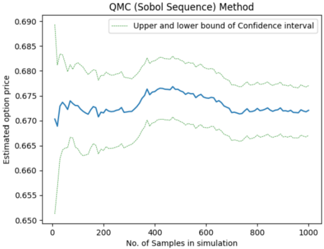
avec W_t simulé selon la suite de SOBOL et la méthode d'inversion.





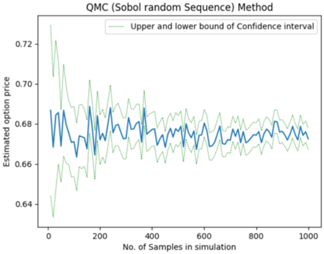
- **Résultats :**
  On obtient des résultats similaires à ceux de Monte Carlo classique



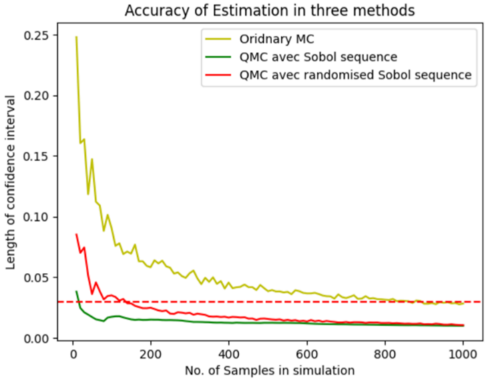


- **Randomised Monte Carlo :**
  On obtient des résultats similaires à ceux de Monte Carlo classique

$$\hat{C} \approx 0.66$$



- **Comparaison des intervalles de confiances**

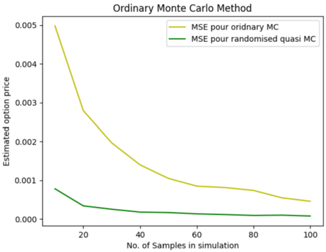


- **Comparaison des MSE**

$$MSE = \mathbb{E}[P_L] = \mathbb{E}[P_0] + \sum_{l=1}^{L} \mathbb{E}[P_l - P_{l-1}]$$

Nombre de simulations $\simeq 180000$

$$C \approx 0.66$$



	Valeurs des MSE
Ordinary	1.12*10(-6)
Randomised Quasi Monte Carlo	7.59*10(-6)
Multi-level Monte Carlo	3.13*10(-4)

- **Comparaison des CPU Times**

Nombre de simulations $\simeq 180000$

	Valeurs des CPU Times
Ordinary	36 secondes
Randomised Quasi Monte Carlo	84 secondes
Multi-level Monte Carlo	6.9 secondes