

# Discrétisation d'équations différentielles stochastiques

Ouvrez un navigateur internet à l'adresse

<http://cermics.enpc.fr/~jourdain/MC/MonteCarlo.html>

L'objectif de ce TP est d'illustrer numériquement dans le cadre du modèle de Black-Scholes les résultats du cours sur les schémas d'Euler et de Milstein. On note  $T$  l'horizon de la simulation,  $\Delta t = T/N$  (où  $N \in \mathbb{N}^*$ ) le pas de discrétisation et pour  $k \in \{0, \dots, N\}$ ,  $t_k = k\Delta t$  le  $k$ -ème instant de discrétisation.

## 1 Convergence forte

L'intérêt de se placer dans le modèle de Black-Scholes

$$dS_t = \sigma S_t dW_t + r S_t dt$$

est la possibilité de simuler exactement le sous-jacent en utilisant les mêmes accroissements browniens que ceux qui servent à générer les processus discrétisés.

1. Exprimer en fonction de la valeur en  $t_k$  du sous-jacent  $S_{t_k}$  (resp. du schéma d'Euler  $S_{t_k}^e$ , resp. du schéma de Milstein  $S_{t_k}^m$ ) et de l'accroissement  $\Delta W_{k+1} = W_{t_{k+1}} - W_{t_k}$  la valeur en  $t_{k+1}$  du sous-jacent  $S_{t_{k+1}}$  (resp. du schéma d'Euler  $S_{t_{k+1}}^e$ , resp. du schéma de Milstein  $S_{t_{k+1}}^m$ ).
2. Implémenter ces formules dans les zones à compléter du programme *vitfort\_Q.py/vitfort\_Q.sce*.
3. Quel est le comportement théorique de  $\mathbb{E} \left( \max_{1 \leq k \leq N} (S_{\frac{kT}{N}} - S_{\frac{kT}{N}}^e)^2 \right)$  en fonction de  $N$ ? Et celui de  $\mathbb{E} \left( \max_{1 \leq k \leq N} (S_{\frac{kT}{N}} - S_{\frac{kT}{N}}^m)^2 \right)$ ?
4. Exécuter le programme pour constater la dépendance en  $1/N$  de  $\mathbb{E} \left( \max_{1 \leq k \leq N} (S_{\frac{kT}{N}} - S_{\frac{kT}{N}}^e)^2 \right)$  et en  $1/N^2$  de  $\mathbb{E} \left( \max_{1 \leq k \leq N} (S_{\frac{kT}{N}} - S_{\frac{kT}{N}}^m)^2 \right)$ .

## 2 Convergence faible

### 2.1 Vitesse faible

On souhaite maintenant étudier la vitesse faible des schémas d'Euler et de Milstein pour le calcul d'un Put européen d'échéance  $T$  dans le modèle de Black-Scholes, c'est-à-dire la dépendance en  $N$  des quantités  $\mathbb{E}(e^{-rT}(K - S_T^e)^+) - \mathbb{E}(e^{-rT}(K - S_T)^+)$  et  $\mathbb{E}(e^{-rT}(K - S_T^m)^+) - \mathbb{E}(e^{-rT}(K - S_T)^+)$ . Pour cela, on va évaluer ces quantités d'une part en calculant  $\mathbb{E}(e^{-rT}(K - S_T)^+)$  par la formule de Black-Scholes et d'autre part en approchant cette espérance par un calcul Monte-Carlo utilisant les mêmes accroissements browniens que

ceux qui servent à calculer  $\mathbb{E}(e^{-rT}(K - S_T^e)^+)$  et  $\mathbb{E}(e^{-rT}(K - S_T^m)^+)$ . La deuxième approche entre dans le cadre des techniques de variables de contrôle pour réduire la variance.

1. Compléter le programme *vitfaible\_Q.py/vitfaible\_Q.sce*.
2. Quel est le comportement théorique de  $\mathbb{E}(e^{-rT}(K - S_T^e)^+) - \mathbb{E}(e^{-rT}(K - S_T)^+)$  en fonction de  $N$  ?
3. Exécuter le programme pour constater la vitesse faible effective des schémas d'Euler et de Milshtein. La technique de variable de contrôle est-elle efficace ?

## 2.2 Extrapolation de Romberg

On souhaite maintenant étudier l'accélération de la convergence faible par la méthode d'extrapolation de Romberg. On note respectivement  $S^{e,N}$  et  $S^{e,2N}$  (resp.  $S^{m,N}$  et  $S^{m,2N}$ ) les schémas d'Euler (resp. Milshtein) pour  $N$  resp.  $2N$  pas de temps. On va évaluer

$$\mathbb{E} \left( e^{-rT} \left[ 2(K - S_T^{e,2N})^+ - (K - S_T^{e,N})^+ - (K - S_T)^+ \right] \right)$$

et la quantité analogue pour le schéma de Milshtein en utilisant les mêmes accroissements browniens pour chacun des termes dans l'espérance afin de réduire la variance.

1. Pour  $k \in \{0, \dots, N-1\}$  exprimer  $S_{\frac{(k+1)T}{N}}$  (resp.  $S_{\frac{(k+1)T}{N}}^{e,N}$ ,  $S_{\frac{(k+1)T}{N}}^{e,2N}$ ,  $S_{\frac{(k+1)T}{N}}^{m,N}$ ,  $S_{\frac{(k+1)T}{N}}^{m,2N}$ ) en fonction de  $S_{\frac{kT}{N}}$  (resp.  $S_{\frac{kT}{N}}^{e,N}$ ,  $S_{\frac{kT}{N}}^{e,2N}$ ,  $S_{\frac{kT}{N}}^{m,N}$ ,  $S_{\frac{kT}{N}}^{m,2N}$ ) et des accroissements  $W_{\frac{(2k+1)T}{2N}} - W_{\frac{kT}{N}}$  et  $W_{\frac{(k+1)T}{N}} - W_{\frac{(2k+1)T}{2N}}$ .
2. Implémenter ces formules dans le programme *romberg\_Q.py/romberg\_Q.sce* puis l'exécuter. Que constatez-vous ?