

Cours complet : Pricing de dérivés et risque (Monte Carlo)

et accélération quantique (QAE / Quantum Monte Carlo)

Notes de travail (Projet Quant)

17 janvier 2026

Table des matières

1 Objectif du document et feuille de route	5
1.1 But : tout comprendre, de zéro à pipeline complet	5
1.2 Roadmap typique (alignée projet)	5
I Finance classique : pricing par mesure risque-neutre	5
2 Instruments : dérivés, payoffs, notions minimales	5
2.1 Actif sous-jacent	5
2.2 Options européennes : payoffs	5
2.3 Options dépendant de plusieurs actifs	6
3 Modèles : Black–Scholes et lognormalité	6
3.1 SDE de Black–Scholes	6
3.2 Solution explicite et loi de S_T	6
3.3 Multi-actifs et corrélations	6
4 Pricing : mesure risque-neutre, martingale, actualisation	6
4.1 Idée fondamentale	6
4.2 Dynamics sous \mathbb{Q}	6
4.3 Formule de pricing (européen)	7
4.4 Lien PDE (Black–Scholes PDE)	7
5 Monte Carlo classique : estimateur, erreur, coût	7
5.1 Estimateur MC	7
5.2 Convergence et loi des grands nombres	7
5.3 Erreur statistique et CLT	7
5.4 Intervalle de confiance (pratique)	7
5.5 Variance reduction (indispensable en pratique)	8
5.6 Pseudocode : MC call européen BS	8
II Risque : VaR / CVaR et pourquoi c'est dur	8

6 P&L, pertes, quantiles	8
6.1 Variable de perte	8
6.2 VaR	8
6.3 CVaR / Expected Shortfall	8
6.4 MC pour VaR/CVaR	8
III Bases quantiques minimales (tout ce qu'il faut, rien de magique)	9
7 Qubits, états, mesures	9
7.1 État d'un qubit	9
7.2 Registre n qubits	9
8 Portes, circuits, unitaires	9
8.1 Portes standards	9
8.2 Oracle (idée)	9
IV Quantum-Accelerated Monte Carlo : QAE / QAMC	9
9 Le problème cible : calculer une espérance	10
10 Pipeline QAMC (les 4 étapes)	10
11 Encodage amplitude : comment l'espérance devient une probabilité	10
11.1 Préparation de la distribution	10
11.2 Encodage de f par rotation contrôlée	10
11.3 Remarque critique : normalisation	11
12 Amplitude Amplification : intuition géométrique	11
12.1 Décomposition bon/mauvais	11
12.2 Opérateur de Grover et rotations	11
13 Amplitude Estimation (QAE) : récupérer a efficacement	11
13.1 Pourquoi QAE donne un avantage quadratique	11
13.2 QAE originale (avec QFT inverse)	11
13.3 QAE sans QFT : variantes NISQ-friendly	11
14 Coût réel : où se cache la difficulté	12
V Goulot #1 : State preparation (charger une distribution)	12
15 Le problème	12
16 Trois cas typiques en finance	12
16.1 Cas A : loi analytique (BS, gaussienne, lognormale)	12
16.2 Cas B : loi empirique (VaR sur données)	12
16.3 Cas C : loi générée par un processus (SDE simulée)	12

17 Méthodes pour préparer p	12
17.1 Préparation exacte générique (peu scalable)	12
17.2 Approche Grover/constructive par partitions	13
17.3 Variational circuits	13
17.4 Tensor networks	13
18 Discrétisation : erreur systématique	13
VI Goulot #2 : Payoff loading (encoder $f(x)$)	13
19 Pourquoi c'est non-trivial	13
20 Cas d'école : payoff européen 1D	13
21 Techniques usuelles	13
21.1 Comparateurs + arithmétique réversible	13
21.2 Fonctions approchées	14
VII Application complète : call européen BS via QAE	14
22 Objectif	14
23 Choix d'encodage	14
23.1 Étape 1 : variable gaussienne	14
23.2 Étape 2 : discrétisation	14
23.3 Étape 3 : préparer $\sum \sqrt{p(z_i)} i\rangle$	14
23.4 Étape 4 : calculer le payoff	14
23.5 Étape 5 : rotation auxiliaire	14
24 Estimation (QAE)	15
25 Benchmark honnête : que comparer ?	15
25.1 Comparaison statistique	15
25.2 Comparaison ressources	15
VIII VaR / CVaR quantiques (idée générale)	15
26 VaR comme inversion de CDF	15
27 Encodage indicatrice	15
28 CVaR	15
IX Limites pratiques et message important	15
29 Pourquoi l'avantage peut disparaître	16

30 Checklist de compréhension (si tu sais tout ça, tu maîtrises)	16
X Annexes	16
A Exercices (très formateurs)	16

1 Objectif du document et feuille de route

1.1 But : tout comprendre, de zéro à pipeline complet

On veut être capable de :

- pricer un dérivé (ex : call européen) par approche **risque-neutre** ;
- comprendre pourquoi **Monte Carlo classique** converge en $O(1/\sqrt{M})$;
- comprendre et implémenter mentalement (puis en code) le pipeline **Quantum-Accelerated Monte Carlo** (QAMC) basé sur **Quantum Amplitude Estimation** (QAE) ;
- expliquer clairement **où est le gain quadratique** et **où se cachent les coûts** (chargement de distribution, profondeur, bruit NISQ, etc.) ;
- faire une comparaison benchmark : **temps/erreur** entre MC classique et QAE (au moins en simulation).

1.2 Roadmap typique (alignée projet)

- **S1–S2** : finance classique + MC + revue litté.
- **S3–S4** : briques quantiques (QPE/QAE).
- **S5–S7** : circuit complet de pricing quantique.
- **S8–S9** : benchmarks + analyse.
- **S10–S12** : rédaction + soutenance.

Méthode de lecture conseillée : lire dans l'ordre des Parties I → IV, puis revenir sur la Partie V (state preparation) et VI (payoff loading), qui sont les vrais goulots d'étranglement.

Première partie

Finance classique : pricing par mesure risque-neutre

2 Instruments : dérivés, payoffs, notions minimales

2.1 Actif sous-jacent

On note S_t le prix (spot) de l'actif à la date t . On suppose un horizon $[0, T]$.

2.2 Options européennes : payoffs

Définition 2.1 (Call et Put européens). Un **call européen** de strike K et maturité T a pour payoff :

$$h_{\text{call}}(S_T) = \max(S_T - K, 0).$$

Un **put européen** :

$$h_{\text{put}}(S_T) = \max(K - S_T, 0).$$

2.3 Options dépendant de plusieurs actifs

Si $S_t = (S_t^1, \dots, S_t^d)$:

- **basket call** : $h(S_T) = \max\left(\sum_{i=1}^d w_i S_T^i - K, 0\right)$;
- **best-of** : $h(S_T) = \max(\max_i S_T^i - K, 0)$;
- **spread** (2 actifs) : $h(S_T) = \max(S_T^1 - S_T^2 - K, 0)$.

En dimension élevée, les méthodes PDE souffrent (curse of dimensionality), alors que MC reste naturel.

3 Modèles : Black–Scholes et lognormalité

3.1 SDE de Black–Scholes

Sous une mesure physique (historique) \mathbb{P} :

$$dS_t = \mu S_t dt + \sigma S_t dW_t,$$

où μ est le drift, σ la volatilité, W_t un Brownien.

3.2 Solution explicite et loi de S_T

On applique Itô à $\ln S_t$:

$$d(\ln S_t) = \left(\mu - \frac{1}{2}\sigma^2\right) dt + \sigma dW_t.$$

Intégration sur $[0, T]$:

$$\ln S_T = \ln S_0 + \left(\mu - \frac{1}{2}\sigma^2\right) T + \sigma W_T,$$

donc $\ln S_T$ est normale et S_T est **lognormale**.

3.3 Multi-actifs et corrélations

Pour d actifs :

$$\frac{dS_t^i}{S_t^i} = \mu_i dt + \sigma_i dW_t^i, \quad dW_t^i dW_t^j = \rho_{ij} dt.$$

En simulation : on génère un vecteur gaussien corrélé via Cholesky de ρ .

4 Pricing : mesure risque-neutre, martingale, actualisation

4.1 Idée fondamentale

En absence d'arbitrage et sous hypothèses standard (marché complet BS), il existe une mesure \mathbb{Q} telle que le prix actualisé est une martingale.

4.2 Dynamics sous \mathbb{Q}

Sous \mathbb{Q} :

$$dS_t = rS_t dt + \sigma S_t dW_t^{\mathbb{Q}}.$$

C'est le drift r (taux sans risque). On peut alors pricer par espérance.

4.3 Formule de pricing (européen)

Définition 4.1 (Prix risque-neutre). Le prix à $t = 0$ d'un payoff $h(S_T)$:

$$V_0 = e^{-rT} \mathbb{E}^{\mathbb{Q}}[h(S_T)].$$

Plus généralement,

$$V_t = e^{-r(T-t)} \mathbb{E}^{\mathbb{Q}}[h(S_T) | \mathcal{F}_t].$$

4.4 Lien PDE (Black–Scholes PDE)

La valeur $V(t, S)$ satisfait :

$$V_t + rSV_S + \frac{1}{2}\sigma^2 S^2 V_{SS} - rV = 0, \quad V(T, S) = h(S).$$

Cette équivalence est cruciale : on a **deux familles** numériques (PDE vs MC).

5 Monte Carlo classique : estimateur, erreur, coût

5.1 Estimateur MC

On simule M scénarios $S_T^{(m)}$ i.i.d. sous \mathbb{Q} et on calcule :

$$\hat{V}_0 = e^{-rT} \frac{1}{M} \sum_{m=1}^M h(S_T^{(m)}).$$

5.2 Convergence et loi des grands nombres

Théorème 5.1 (LLN). *Si $\mathbb{E}[|h(S_T)|] < \infty$, alors*

$$\hat{V}_0 \xrightarrow[M \rightarrow \infty]{} V_0 \quad p.s.$$

5.3 Erreur statistique et CLT

Théorème 5.2 (CLT). *Si $\text{Var}(h(S_T)) < \infty$, alors*

$$\sqrt{M} \left(\frac{1}{M} \sum_{m=1}^M h(S_T^{(m)}) - \mathbb{E}[h(S_T)] \right) \xrightarrow{d} \mathcal{N}(0, \text{Var}(h(S_T))).$$

Donc l'erreur typique est $O(1/\sqrt{M})$.

5.4 Intervalle de confiance (pratique)

$$\hat{V}_0 \pm z_{1-\alpha/2} \cdot e^{-rT} \sqrt{\frac{\widehat{\text{Var}}(h)}{M}}.$$

On voit : **gagner 1 chiffre de précision** coûte **environ $\times 100$** en M .

5.5 Variance reduction (indispensable en pratique)

- Antithetic variates : Z et $-Z$.
- Control variates : utiliser un payoff proche à prix connu.
- Importance sampling : échantillonner plus la queue.
- Stratification / Quasi-MC : séquences de Sobol (attention dimension effective).

5.6 Pseudocode : MC call européen BS

Algorithm 1 Monte Carlo BS (call européen)

Require: S_0, K, r, σ, T, M

```

1: sum  $\leftarrow 0$ 
2: for  $m = 1$  to  $M$  do
3:   Tirer  $Z \sim \mathcal{N}(0, 1)$ 
4:    $S_T \leftarrow S_0 \exp\left((r - \frac{1}{2}\sigma^2)T + \sigma\sqrt{T}Z\right)$ 
5:   sum  $\leftarrow$  sum + max( $S_T - K, 0$ )
6: end for
7: return  $e^{-rT} \cdot \text{sum}/M$ 

```

Deuxième partie

Risque : VaR / CVaR et pourquoi c'est dur

6 P&L, pertes, quantiles

6.1 Variable de perte

On note L une variable aléatoire de **perte** (positive = perte). Dans la pratique, L est un P&L transformé.

6.2 VaR

Définition 6.1 (VaR). Au niveau $\alpha \in (0, 1)$:

$$\text{VaR}_\alpha(L) = \inf\{\ell \in \mathbb{R} : \mathbb{P}(L \leq \ell) \geq \alpha\}.$$

6.3 CVaR / Expected Shortfall

Définition 6.2 (CVaR).

$$\text{CVaR}_\alpha(L) = \mathbb{E}[L \mid L \geq \text{VaR}_\alpha(L)].$$

La CVaR est souvent préférée car plus sensible à la queue.

6.4 MC pour VaR/CVaR

1. Simuler $L^{(1)}, \dots, L^{(M)}$.
2. Trier : $L_{(1)} \leq \dots \leq L_{(M)}$.
3. $\widehat{\text{VaR}}_\alpha \approx L_{(\lceil \alpha M \rceil)}$.

$$4. \widehat{\text{CVaR}}_{\alpha} \approx \frac{1}{M - \lceil \alpha M \rceil + 1} \sum_{k=\lceil \alpha M \rceil}^M L_{(k)}.$$

Aux niveaux α très élevés, il faut énormément d'échantillons pour stabiliser la queue.

Troisième partie

Bases quantiques minimales (tout ce qu'il faut, rien de magique)

7 Qubits, états, mesures

7.1 État d'un qubit

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1.$$

Mesure dans la base computationnelle : 0 avec proba $|\alpha|^2$ et 1 avec proba $|\beta|^2$.

7.2 Registre n qubits

$$|\psi\rangle = \sum_{x=0}^{2^n-1} c_x |x\rangle, \quad \sum_x |c_x|^2 = 1.$$

L'espace de Hilbert a dimension 2^n .

8 Portes, circuits, unitaires

8.1 Portes standards

- X : NOT, $X|0\rangle = |1\rangle$.
- H : Hadamard, $H|0\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$.
- $R_y(\theta)$: rotation, $R_y(\theta)|0\rangle = \cos(\theta/2)|0\rangle + \sin(\theta/2)|1\rangle$.
- CNOT : contrôle sur un qubit, flip la cible.

8.2 Oracle (idée)

Un oracle est une unité (ou sous-circuit) qui encode une information sur les états. En QAMC on parle souvent de deux oracles :

- P : prépare un état dont les amplitudes encodent une distribution $p(x)$;
- R : encode une fonction $f(x)$ via une rotation sur un qubit auxiliaire.

Quatrième partie

Quantum-Accelerated Monte Carlo : QAE / QAMC

9 Le problème cible : calculer une espérance

On veut estimer

$$\mu := \mathbb{E}_{X \sim p}[f(X)] = \sum_x p(x)f(x).$$

En finance : f est un payoff (pricing) ou une indicatrice (VaR/CVaR).

10 Pipeline QAMC (les 4 étapes)

On se base sur le schéma standard :

1. charger $p(x)$ dans le registre quantique ;
2. charger $f(x)$;
3. amplifier (type Grover) ;
4. estimer l'amplitude (QAE, différentes variantes).

11 Encodage amplitude : comment l'espérance devient une probabilité

11.1 Préparation de la distribution

On vise un état

$$P|0\rangle^{\otimes n} = \sum_{x=0}^{2^n-1} \sqrt{p(x)} |x\rangle.$$

11.2 Encodage de f par rotation contrôlée

On ajoute un qubit auxiliaire initialisé à $|0\rangle$. On applique R tel que :

$$R : |x\rangle |0\rangle \mapsto |x\rangle \left(\sqrt{1-f(x)} |0\rangle + \sqrt{f(x)} |1\rangle \right),$$

avec $f(x) \in [0, 1]$ (sinon on normalise). L'état global devient :

$$\sum_x \sqrt{p(x)} |x\rangle \left(\sqrt{1-f(x)} |0\rangle + \sqrt{f(x)} |1\rangle \right).$$

La probabilité de mesurer l'auxiliaire à 1 vaut :

$$a^2 = \sum_x p(x)f(x) = \mu.$$

Donc **estimer une espérance** revient à **estimer une amplitude**.

11.3 Remarque critique : normalisation

Si ton payoff est $g(x) \geq 0$ non borné par 1, tu fais :

$$f(x) = \frac{g(x)}{C} \in [0, 1] \quad \Rightarrow \quad \mathbb{E}[g] = C \mathbb{E}[f].$$

Choisir C trop grand réduit l'amplitude et peut rendre l'estimation plus bruyante.

12 Amplitude Amplification : intuition géométrique

12.1 Décomposition bon/mauvais

On peut écrire l'état obtenu après $A := R \circ (P \otimes I)$:

$$A |0\rangle = a |\Psi_1\rangle + \sqrt{1 - a^2} |\Psi_0\rangle,$$

où $|\Psi_1\rangle$ correspond aux états “bons” (auxiliaire = 1).

12.2 Opérateur de Grover et rotations

On définit un opérateur Q qui agit comme une rotation dans le plan engendré par $|\Psi_1\rangle$ et $|\Psi_0\rangle$. En posant $a = \sin \theta$, l'application répétée :

$$Q^k A |0\rangle = \sin((2k+1)\theta) |\Psi_1\rangle + \cos((2k+1)\theta) |\Psi_0\rangle.$$

Choisir k permet d'amplifier la proba de succès.

13 Amplitude Estimation (QAE) : récupérer a efficacement

13.1 Pourquoi QAE donne un avantage quadratique

- **Sampling classique** : estimer une proba a^2 par fréquences \Rightarrow erreur $O(1/\sqrt{M})$.
- **QAE** : exploite amplitude amplification et estimation de phase \Rightarrow erreur idéale $O(1/M)$.

13.2 QAE originale (avec QFT inverse)

Approche type *phase estimation* :

- registre d'ancilla de taille m ;
- contrôles de Q^{2^j} ;
- QFT inverse pour extraire une estimation de θ puis $a = \sin \theta$.

Point clé : la QFT inverse est coûteuse en profondeur et en connectivité.

13.3 QAE sans QFT : variantes NISQ-friendly

Max Likelihood QAE On exécute des circuits avec différents k , on mesure, puis on fait un post-traitement classique (MLE) pour estimer θ .

Iterative QAE On évite QFT en adaptant k et les mesures pour réduire progressivement l'intervalle possible de θ .

Trade-off profondeur / nombre d'appels Certaines variantes (power-law, etc.) choisissent des familles de k pour minimiser la profondeur maximale tout en conservant un gain.

14 Coût réel : où se cache la difficulté

Même si QAE est “quadratiquement meilleure” en appels oracle, le coût total dépend fortement de :

- la complexité de P (préparer p) ;
- la complexité de R (encoder f) ;
- la profondeur totale (répéter Q de nombreuses fois) ;
- le bruit matériel (NISQ) et l'overhead de correction d'erreurs.

C'est pour ça qu'en pratique, le **chargement d'état** est souvent le goulot.

Cinquième partie

Goulot #1 : State preparation (charger une distribution)

15 Le problème

On veut préparer $\sum_x \sqrt{p(x)} |x\rangle$. Si p est une distribution “arbitraire” sur 2^n points, une préparation exacte peut coûter $O(2^n)$ (mauvais).

16 Trois cas typiques en finance

16.1 Cas A : loi analytique (BS, gaussienne, lognormale)

On peut exploiter une structure : S_T lognormal $\Rightarrow \ln S_T$ gaussien. On discrétise un intervalle $[x_{\min}, x_{\max}]$ et on définit $p(x)$ via une quadrature/normalisation.

16.2 Cas B : loi empirique (VaR sur données)

Il faut charger une distribution observée : c'est précisément le cas où le chargement peut dominer.

16.3 Cas C : loi générée par un processus (SDE simulée)

“Générer la distribution” revient à simuler une SDE : c'est dur aussi, mais différent de “charger un histogramme”.

17 Méthodes pour préparer p

17.1 Préparation exacte générique (peu scalable)

Méthodes générales de “state preparation” existent mais ne scalent pas bien pour de grands n .

17.2 Approche Grover/constructive par partitions

Idée : partitionner l'espace en 2^n cellules, construire l'état itérativement via des rotations contrôlées dont les angles codent des probabilités conditionnelles.

17.3 Variational circuits

On choisit un ansatz $U(\lambda)$ et on optimise λ pour que les probabilités de sortie approximent p . Avantage : potentiellement moins profond. Inconvénient : entraînement non convexe, risque de barren plateaus.

17.4 Tensor networks

Approximer p via une structure TN (MPS, TT) puis compiler en circuit. Souvent pertinent pour distributions à faible entanglement effectif.

18 Discréétisation : erreur systématique

Le quantique ne supprime pas les erreurs de modèle :

- erreur de troncature (x_{\min}, x_{\max}) ;
- erreur de discréétisation (pas de grille) ;
- erreur d'approximation (circuit de préparation).

Pour un benchmark honnête, tu dois isoler l'erreur **statistique** (MC/QAE) de l'erreur **systématique**.

Sixième partie

Goulot #2 : Payoff loading (encoder $f(x)$)

19 Pourquoi c'est non-trivial

Encoder $f(x)$ signifie construire un circuit qui, conditionné sur $|x\rangle$, applique une rotation $R_y(\theta_x)$ telle que $\sin^2(\theta_x/2) = f(x)$.

20 Cas d'école : payoff européen 1D

Si x représente (une discréétisation de) S_T :

$$f(x) = \frac{\max(S(x) - K, 0)}{C}.$$

Alors f est une fonction **par morceaux**, monotone après K .

21 Techniques usuelles

21.1 Comparateurs + arithmétique réversible

On construit :

- un comparateur $\mathbf{1}\{S > K\}$;

- une soustraction réversible $S - K$;
- une normalisation (division par C) ;
- une rotation contrôlée par la valeur calculée (approximation polynomiale, table lookup, CORDIC-like).

Tout ça coûte des qubits auxiliaires (work qubits) et de la profondeur.

21.2 Fonctions approchées

On approxime f par polynôme (Chebyshev) ou par segments linéaires, ce qui simplifie les rotations mais ajoute une erreur.

Septième partie

Application complète : call européen BS via QAE

22 Objectif

Estimer

$$V_0 = e^{-rT} \mathbb{E}^{\mathbb{Q}}[\max(S_T - K, 0)].$$

23 Choix d'encodage

23.1 Étape 1 : variable gaussienne

Écrire

$$S_T = S_0 \exp((r - \frac{1}{2}\sigma^2)T + \sigma\sqrt{T}Z), \quad Z \sim \mathcal{N}(0, 1).$$

On peut discréteriser Z (plus simple) puis calculer S_T .

23.2 Étape 2 : discréétisation

Choisir un intervalle $[z_{\min}, z_{\max}]$ (ex : ± 6) et 2^n points. Définir $p(z_i) \propto \varphi(z_i)$ (densité gaussienne) avec quadrature.

23.3 Étape 3 : préparer $\sum \sqrt{p(z_i)} |i\rangle$

C'est l'oracle P .

23.4 Étape 4 : calculer le payoff

Calculer $S(z_i)$ puis $g_i = \max(S(z_i) - K, 0)$, normaliser $f_i = g_i/C$.

23.5 Étape 5 : rotation auxiliaire

Appliquer R tel que $|i\rangle |0\rangle \mapsto |i\rangle (\sqrt{1-f_i} |0\rangle + \sqrt{f_i} |1\rangle)$.

24 Estimation (QAE)

Après construction de $A = R \circ (P \otimes I)$, on exécute une variante de QAE pour estimer $a^2 = \sum p_i f_i$.
On renvoie :

$$\widehat{V}_0 = e^{-rT} C \widehat{a^2}.$$

25 Benchmark honnête : que comparer ?

25.1 Comparaison statistique

Comparer la relation **erreur vs nombre d'appels oracle** :

- MC : erreur $\sim O(1/\sqrt{M})$ sur M samples ;
- QAE : erreur $\sim O(1/M)$ sur M appels structurés (idéal).

25.2 Comparaison ressources

Comparer :

- profondeur (gates) ;
- nombre de qubits (registre + ancilla + work) ;
- sensibilité au bruit (ex : répétitions de Q) .

Huitième partie

VaR / CVaR quantiques (idée générale)

26 VaR comme inversion de CDF

Si $F_L(\ell) = \mathbb{P}(L \leq \ell)$, alors VaR_α est un quantile :

$$\text{VaR}_\alpha = \inf\{\ell : F_L(\ell) \geq \alpha\}.$$

On peut estimer $F_L(\ell)$ pour un ℓ donné (via indicatrice $\mathbf{1}\{L \leq \ell\}$), puis faire une recherche dichotomique.

27 Encodage indicatrice

On construit un circuit qui produit un qubit “flag” valant 1 si $L \leq \ell$. Alors l’amplitude correspond à $F_L(\ell)$.

28 CVaR

Une fois VaR_α trouvée, la CVaR s’écrit comme une espérance conditionnelle (ou intégrale de queue), et peut aussi être estimée via des combinaisons d’indicatrices et d’espérances.

Neuvième partie

Limites pratiques et message important

29 Pourquoi l'avantage peut disparaître

Même si QAE offre un gain quadratique sur l'estimation d'une espérance, le temps total dépend de T_P (préparation) et T_R (payoff), et la profondeur de circuit peut exploser. Sur matériel NISQ, c'est souvent le **bottleneck** principal.

30 Checklist de compréhension (si tu sais tout ça, tu maîtrises)

Tu dois pouvoir répondre (sans regarder tes notes) :

1. Pourquoi S_T est lognormal sous BS ?
2. Pourquoi le pricing est une espérance sous \mathbb{Q} ?
3. Pourquoi MC a une erreur $O(1/\sqrt{M})$?
4. Comment transformer $\mathbb{E}[f(X)]$ en amplitude mesurable ?
5. C'est quoi P et R exactement ?
6. Pourquoi la QFT inverse est un souci et quelles alternatives existent ?
7. Où est le vrai coût : préparation distribution, payoff loading, répétitions de Q , bruit ?

Dixième partie

Annexes

A Exercices (très formateurs)

Exercice A : MC et variance

Montrer (par calcul) que $\text{Var}(\frac{1}{M} \sum Y_m) = \text{Var}(Y)/M$.

Exercice B : normalisation payoff

Pour un call, proposer un C raisonnable. Discuter l'impact de C sur le signal a^2 .

Exercice C : discréétisation gaussienne

Choisir z_{\min}, z_{\max} et justifier via une borne sur $\mathbb{P}(|Z| > a)$.

Exercice D : VaR par dichotomie

Écrire un pseudo-code complet : pour ℓ dans un intervalle, estimer $F_L(\ell)$ (par sampling ou QAE), puis mettre à jour l'intervalle.

Références

- [1] F. Black and M. Scholes, The Pricing of Options and Corporate Liabilities, *Journal of Political Economy*, 1973.
- [2] R. Merton, Theory of Rational Option Pricing, *Bell Journal of Economics*, 1973.
- [3] S. Heston, A Closed-Form Solution for Options with Stochastic Volatility, *Review of Financial Studies*, 1993.
- [4] L. Grover, A Fast Quantum Mechanical Algorithm for Database Search, 1996.
- [5] G. Brassard, P. Høyer, M. Mosca, A. Tapp, Quantum Amplitude Amplification and Estimation, 2002.
- [6] A. Gómez et al., A Survey on Quantum Computational Finance for Derivatives Pricing and VaR, 2022.
- [7] R. Orús et al., Quantum Computing for Finance : Overview and Prospects, 2019.
- [8] D. Herman et al., A Survey of Quantum Computing for Finance, 2022.