Option pricing on a quantum computer

Björn Löfdahl, SEB

Quantum Autumn School
December 2-4, 2024

Introduction to option pricing

- A European call (put) option on a stock:
  - Right to buy (sell) the stock S at a known price K at some pre-determined time T
  - K is the *strike*
  - T is the *expiry*
- Banks often sell options to their clients
  - At expiry, the client will receive the *payoff* $\Phi(S_T) = \max(S_T - K, 0)$
  - Need to be able to calculate a fair price!

Introduction to option pricing

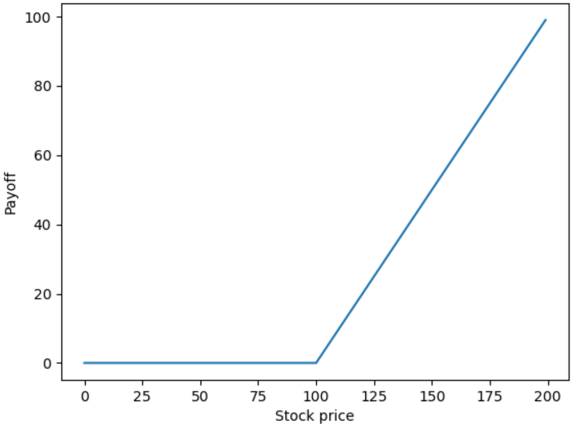

Figure: Payoff function for $K = 100$.

Introduction to option pricing

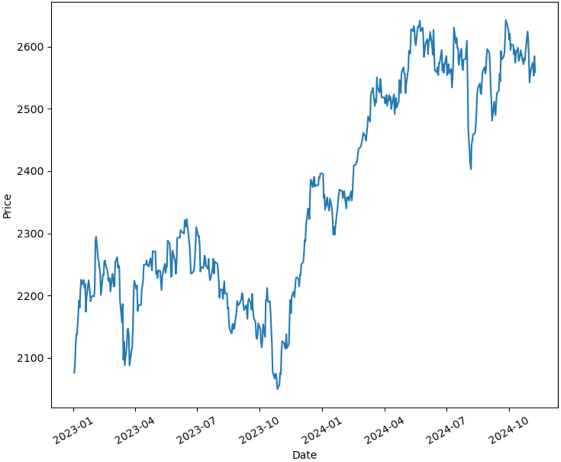

Figure: Recent OMX Index price movements.

Introduction to option pricing

- The fair price v of a European call option is determined by the *discounted expected payoff*

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

- Depends on the distribution of S_T, given the current price S_0!

- Black & Scholes model:,

$$\log\frac{S_T}{S_0} \sim \mathrm{N}((r - \frac{1}{2}\sigma^2)T, \sigma\sqrt{T})$$

  where σ is the *volatility*, and r is the interest rate.

- Monte Carlo can be used to estimate the expected value!

Monte Carlo pricing

- Monte Carlo recipe:
  - Simulate x_i from the log-normal distribution of S_T
  - Evaluate payoff $\Phi(x_i) = \max(x_i - K, 0)$
  - Repeat N times
  - Calculate average payoff!
- From the Law of Large numbers,

$$\frac{1}{N}\sum_{i=1}^{N}\Phi(x_i) \to \mathbb{E}[\max(S_T - K, 0)], \text{ as } N \to \infty$$

Option pricing on a quantum computer

- Let $\mu = \mathbb{E}[\max(S_T - K, 0)]$
- Imagine if we could encode μ in the state $|\Psi\rangle$ of a qubit q in a quantum circuit, e.g.

$$|\Psi\rangle = \sqrt{1-\mu}\,|0\rangle + \sqrt{\mu}\,|1\rangle$$

- The probability of measuring a 1 is $(\sqrt{\mu})^2 = \mu$
- Then, we could:
  - Run circuit with N shots and measure q
  - Calculate average of the measurements $m_1, m_2, \ldots, m_N$
- From the Law of Large numbers,

$$\frac{1}{N}\sum_{i=1}^{N} m_i \to \mu, \text{ as } N \to \infty$$

Encoding the probability distribution

- With n qubits, discretize distribution of stock price to 2^n grid points
- Let $p_i = \mathbb{P}(\text{measuring i})$
- Define operator $\mathcal{A}$ by

$$\mathcal{A}\left|0\right\rangle_n = \sum_{i=0}^{2^n-1} \sqrt{p_i}\left|i\right\rangle_n$$

- $\mathcal{A}$ encodes the probability distribution into a cirquit!

Encoding the payoff function and the expected value

- Consider random variable X on $\{0, 1, \ldots, 2^n - 1\}$ and function $f(X) \mapsto [0, 1]$
- Define operator $\mathcal{F}$ by

$$\mathcal{F} |i\rangle_n |0\rangle = \sqrt{1 - f(i)} |i\rangle_n |0\rangle + \sqrt{f(i)} |i\rangle_n |1\rangle$$

- Applying $\mathcal{F}$ to $\mathcal{A} |0\rangle_n |0\rangle$ yields

$$\mathcal{F}\mathcal{A} |0\rangle_n |0\rangle = \ldots |0\rangle + \sum_{i=0}^{2^n-1} \sqrt{f(i)} \sqrt{p_i} |i\rangle_n |1\rangle$$

- The probability of measuring $|1\rangle$ in the final qubit is

$$\sum_{i=0}^{2^n-1} f(i) p_i = \mathbb{E}[f(X)]$$

Encoding the payoff function and the expected value

- Problem: $f(x) = \max(x - K, 0)$ does not map to $[0, 1]$ interval
- Solution: we rescale it!
- Take $\hat{f}(x) = \frac{f(\phi(x))}{f(x_{\text{max}})}$, with $\phi(x) = x_{\text{min}} + (x_{\text{max}} - x_{\text{min}})\frac{x}{2^n - 1}$.
- Final note:
  - This is NOT quantum MC!
  - This is classical MC, implemented on a quantum computer
- Now ready to implement in Qiskit!

Selected references

Woerner, S. and Egger, D. (2019).
Quantum Risk Analysis
npj Quantum Information, 5(1), 15.

Stamatopoulos, N. et al. (2020).
Option pricing using quantum computers
Quantum, 4, 291.