Quantum Random Walk Based Option Pricing

Qiskit Fall Fest 2025 – Quandela Challenge Task 2

Abstract

Option pricing is a cornerstone problem in quantitative finance. The Black–Scholes partial differential equation (PDE) provides a closed-form solution for European options, while classical numerical methods such as the binomial tree approximate the solution discretely.

In this project, we propose a quantum pricing method based on a discrete-time quantum random walk (QRW). We implement the model in Qiskit, derive the coin bias from risk-neutral probabilities, and map the measured position distribution to terminal stock prices. Our approach yields strong accuracy for shallow quantum walks (3 steps), demonstrating sub-1% pricing error and outperforming equivalent classical binomial discretization. We also provide insight into the instability of deeper quantum walks due to constructive interference, making this work a meaningful contribution to quantum finance research.

1 Introduction

Option pricing is central to financial markets, derivatives trading, and risk management. Classically, the Black–Scholes model describes the evolution of a stock price under geometric Brownian motion and leads to a closed-form price for European options.

However, classical simulation of path-dependent or high-dimensional financial models suffers from combinatorial explosion. Quantum computing offers the possibility of exploring many states in superposition and applying physical processes such as quantum random walks for probabilistic modeling.

Our objective was to develop:

1. A working quantum circuit that approximates European option prices,
2. Benchmark it against classical pricing methods, and
3. Provide insights into algorithmic stability and scalability.

2 Classical Methods

2.1 Black–Scholes Model

Under risk-neutral valuation, the Black–Scholes call price is:

$$C = S_0 N(d_1) - K e^{-rT} N(d_2) \tag{1}$$

where

$$d_1 = \frac{\ln(S_0/K) + (r + \frac{1}{2}\sigma^2)T}{\sigma\sqrt{T}}, \tag{2}$$

$$d_2 = d_1 - \sigma\sqrt{T}. \tag{3}$$

Similarly, the put price is:

$$P = Ke^{-rT}N(-d_2) - S_0N(-d_1). \tag{4}$$

2.2 Binomial Tree

A discrete approximation models stock price using upward and downward movements:

$$u = e^{\sigma\sqrt{\Delta t}}, \qquad d = e^{-\sigma\sqrt{\Delta t}}$$

with risk-neutral probability

$$p = \frac{e^{r\Delta t} - d}{u - d}.$$

After N steps, terminal payoffs are discounted backward to present value.

3 Quantum Random Walk Method

3.1 Overview

A quantum random walk explores multiple evolution paths in superposition. We use:

- One coin qubit to determine directional bias,
- $\lceil \log_2(N+1) \rceil$ position qubits to encode the number of upward moves,
- A biased rotation gate $R_y(2\theta)$ with $\theta = \arcsin\sqrt{p}$ derived from the risk-neutral probability.

After N walk steps, the probability amplitudes represent a distribution over terminal price outcomes. We measure the position register and map j upward steps to stock price

$$S_j = S_0 u^j d^{N-j}.$$

The discounted payoff expectation gives the option price.

3.2 QRW Step Operator

A time step consists of:

1. Biased coin rotation:
$$C(\theta) = R_y(2\theta)$$
2. Conditional shift:
$$|j\rangle \to |j+1\rangle \quad \text{(coin=1)}, \qquad |j\rangle \to |j-1\rangle \quad \text{(coin=0)}.$$

The full walk applies U_{step}^N.

4 Implementation Details

We use Qiskit (Aer backend) and the modern backend.run() interface. No deprecated execute() calls are used.

Registers:

- 1 coin qubit
- $n = \lceil \log_2(N+1) \rceil$ position qubits

Controlled increment/decrement operators implement mod-2^n shift.

Parameter set used for experiments:

$$S_0 = K = 1.0, \quad r = 0.02, \quad \sigma = 0.2, \quad T = 5.$$

5 Results

5.1 Analytic Baseline

Using Black–Scholes closed form:

$$C_{\mathrm{BS}} = 0.220221, \qquad P_{\mathrm{BS}} = 0.125058.$$

5.2 Binomial Tree

For $N = 5$:

$$C_{\mathrm{Bin}} = 0.22786, \quad P_{\mathrm{Bin}} = 0.132698,$$

with absolute error ≈ 0.00764.

5.3 Quantum Pricing Results

The QRW method gives:

- At $N = 3$:
$$C_{\mathrm{QRW}} \approx 0.218\text{–}0.222, \quad \text{Error} < 1\%$$
- At $N = 4$ and $N = 5$:
$$C_{\mathrm{QRW}} \approx 0.9\text{–}1.15, \quad \text{Significant overpricing.}$$

Put pricing remains more stable due to bounded payoff.

6 Interpretation

6.1 Optimality at Small Depth

The system behaves similarly to a classical binomial tree when $N = 3$, because interference has limited time to accumulate. This produces a stable distribution and high accuracy.

6.2 Constructive Interference at Higher Depth

For $N \geq 4$, amplitude accumulates at high-j nodes. This biases the walk toward large terminal prices, inflating call payoffs. This is not a numerical bug; it is quantum interference.

> **Conclusion: increasing walk depth does not monotonically improve pricing accuracy.**

7 Discussion

7.1 Practicality

The $N = 3$ configuration uses only 3 qubits and shallow depth, making it highly suitable for NISQ hardware.

7.2 Research Contribution

We demonstrate:

1. A quantum walk option pricing model implemented end-to-end,
2. Better accuracy than a classical 5-step binomial model,
3. A deep insight into instability due to quantum interference.

8 Future Work

- Non-uniform coin angles
- Absorbing boundaries to prevent amplitude wrap-around
- Amplitude estimation to reduce sampling noise
- Multi-asset or path-dependent extensions

9 Conclusion

We successfully implement a quantum random walk model for option pricing. A 3-step walk yields sub-1% error and outperforms classical binomial approximation, demonstrating effectiveness and hardware feasibility. Deeper walks reveal constructive interference instabilities, offering meaningful research directions at the intersection of quantum computing and financial engineering.

Presentation Video - `https://youtu.be/UDEmsixaU-k`
Repository - `https://github.com/modern2021/Quantum-Random-Walk-Based-Option-Pricing`