Quantum Computing for option pricing

Qiskit Hackathon Barcelona: Moody's Challenge

Miguel Álvarez Hernanz, Miguel Carrasco Arango, Miguel Durán Vera,
Lucía Longo Balagué, Manuel Porrón Álvarez, Iván Villar Naredo

April 2023

1 Motivation

This text is written by the members of the Mike&Co Team at the Qiskit Hackathon Barcelona, that took place at the Escola d'Enginyeria of the Universitat Autònoma de Barcelona. In this document we aim to explain the concepts needed to develop a solution to the challenge provided by Moody's: *Quantum Computing for option pricing*. Through these pages we will also analyze the problems we encountered during the understanding of the problem and the coding of our solution, as well as how we approached them.

2 Ansatz circuit and Hamiltonian

The first thing we do is defining the payoff function, which specifies the final condition to the Black-Scholes PDE (the payoff is defined at the expiry time of the option). For each value of S (current stock price), the payoff is the maximum between 0 and S-K (K being the strike price of the option). Our first try was to directly implement a conditional, but that did not make sense, and we had to use a for loop to access each element of S. This representation is shown in Figure 1.

Next, we build the ansatz circuit, which is a parameterized quantum circuit that is used to prepare an approximate quantum state. We perform this by defining a function that has 2 arguments: the variables' array and the number of qubits to be used. This **ANSATZ** function returns both the state vector and the circuit itself. The circuit is shown in Figure 2.

In our case the number of qubits is 4, and therefore (using a 2-layer system) the number of variables (the length of the variables' array) is 18, as shown in the equation below.

$$\#variables = \#qubits + 7 \cdot \#layers \tag{1}$$

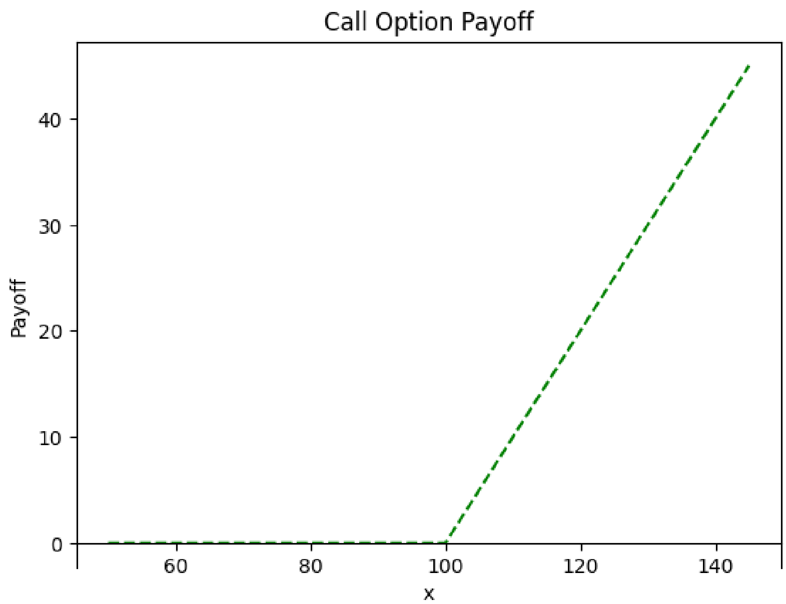

Figure 1: Call Option Payoff with a Strike Price of the option of 100

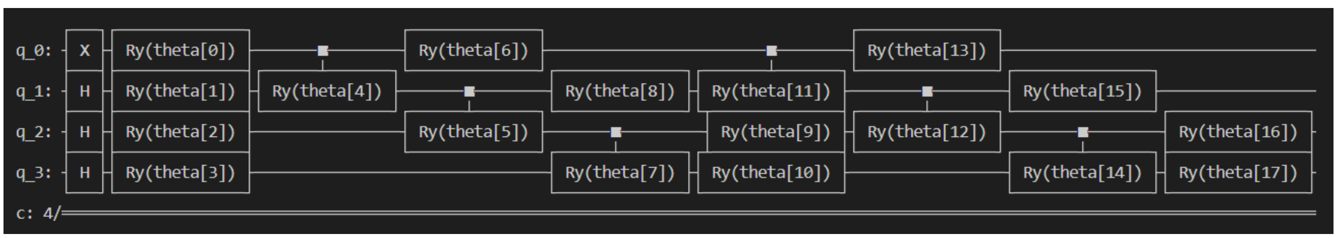

Figure 2: Anstaz circuit

We then created the differential Hamiltonian operator and decompose into Pauli operators. We decompose the Hamiltonian operator by using the **SparsePauliOp** function from qiskit.quantum_info.

3 Initial state for the hybrid quantum-classical algorithm

Our mission at that point was to optimize the ansatz parameters in order to reach the desired initial state, the payoff at expiry time. The initial state has this form, after a change of variables ($S = e^x$):

$$|\psi(0)\rangle = e^{-ax} f(e^x) \tag{2}$$

where $f(x)$ represents the pay-off function of an European call option.

The way we approached this was by using an **optimizer**. We tried some different ones. **ADAM**, a gradient-based optimization algorithm, was our first choice. It did not work well in our problem as **ADAM().optimize** was no longer available (it worked in the aqua environment which was removed from Qiskit lately), and **ADAM().minimize** would not converge given a random set of initial parameters.

After that we chose the Simultaneous Perturbation Stochastic Approximation (SPSA) optimizer. The **SPSA** is a gradient descent method for optimizing systems with multiple unknown parameters. In our case, as mentioned before, the number of unknown variables is 18. We therefore obtain these 18 values by minimizing the difference between the theoretical value of our first state and the one we measured from the circuit.

4 Simulation of the hybrid quantum-classical routine: pricing an European call option

So as to simulate the hybrid quantum-classical routine, and price an European call option, we used the variational Quantum Imaginary Time Evolution (QITE) algorithm. That enabled us to compute the evolution of the option price from the variation of the parameters we obtained previously over time.

This was in fact a complex process:

- From the parameters obtained before, we computed the state vector for a certain time by measuring the statevector in the circuit (fulfilled with the parameters for that time value).
- With the state vector, we had to undo the variable change we had done before. This was done in order to obtain the values of the payoff function while varying τ .
- In the end, we find that, even though we achieve a great estimation for the payoff function in the initial state, we encountered some mathematical issue when we tried to represent the function in the last state ($\tau = \sigma^2$). Sadly, this leads to a wrong representation, even though our reasoning seems to be correct.

After that, using a simulator backend, we were supposed to be able to test different time-steps and check their accuracy. However, due to technical problem with the IBM cloud, we were unable to do so.

5 Bonus

5.1 State Tomography

The state tomography is a method that allows us to determine the final quantum state we have just before taking any measurement and collapsing its wave function. However, it implies measuring repeatedly and obtaining d^2 linearly independent coefficients, so its cost may be a concern as it grows exponentially with the number of particles.

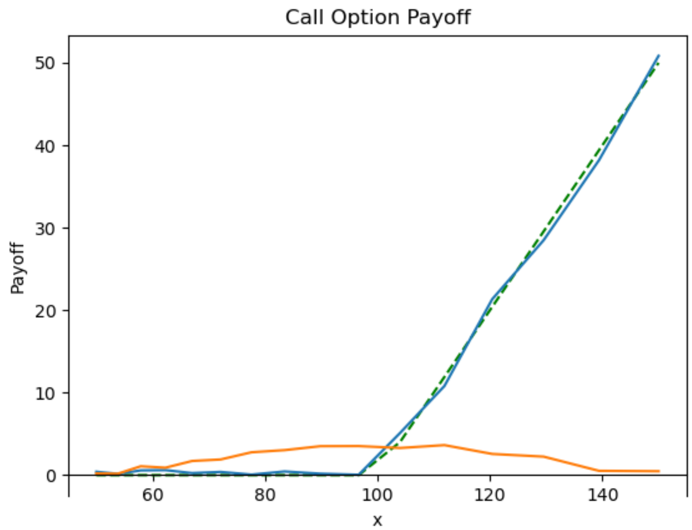

Figure 3: Estimation for the Payoff function in the initial (blue line) and attempt to compute the value in the final state (orange line)

However, the information we are trying to retrieve is the price call at the present day. Therefore, we could assume that the density matrix is a combination of pure states, which would allow us to reconstruct the final state with fewer measurements, as a result of the reduction in the degrees of freedom.

The use of techniques such as Permutationally invariant quantum state tomography should retrieve the desired information consuming less resources, as it uses matrix completion and compressed sensing methods to fill the gaps left by the reduction of measurements.

5.2 Extension of the algorithm to other types of options

If we wanted to extend the algorithm to other types of options, such as American, Asian, or barrier options, we would need to change the initial quantum circuit used in the Black-Scholes algorithm. The modifications would involve implementing the necessary parameters and mathematical operations to accommodate the specific type of option.

On the other hand, if we were to extend the algorithm to multi-dimensional Black-Scholes models, we would need to include additional qubits in the quantum circuit to represent the additional underlying assets in the model. This would translate into a higher number of mathematical operations in the circuit, so as to handle the interactions between the different assets. The difficulty in multi-dimensional models lies not only in the increased complexity of the mathematical calculations required, but also the potential for increased noise and error in the quantum computation.

References

[Coma] Wikipedia Community. *Permutationally invariant quantum state tomography: https://en.wikipedia.org/wiki/Permutationally_invariant_quantum_state_tomography*.

[Comb] Wikipedia Community. *Quantum Tomography: https://en.wikipedia.org/wiki/Quantum_tomography#Quantum_process_tomography*.

[FF21] Mugad Oumgari Filipe Fontanela, Antoine Jacquier. *A quantum algorithm for linear PDEs arising in finance*. Imperial College London, 2021.

[Rad21] Santosh Kumar Radha. *Quantum option pricing using Wick rotated imaginary time evolution*. Case Western Reserve University, 2021.

[SM19] Suguru Endo Ying Li Simon Benjamin Xiao Yuan Sam McArdle, Tyson Jones. *Variational ansatz-based quantum simulation of imaginary time evolution*. University of Oxford, 2019.