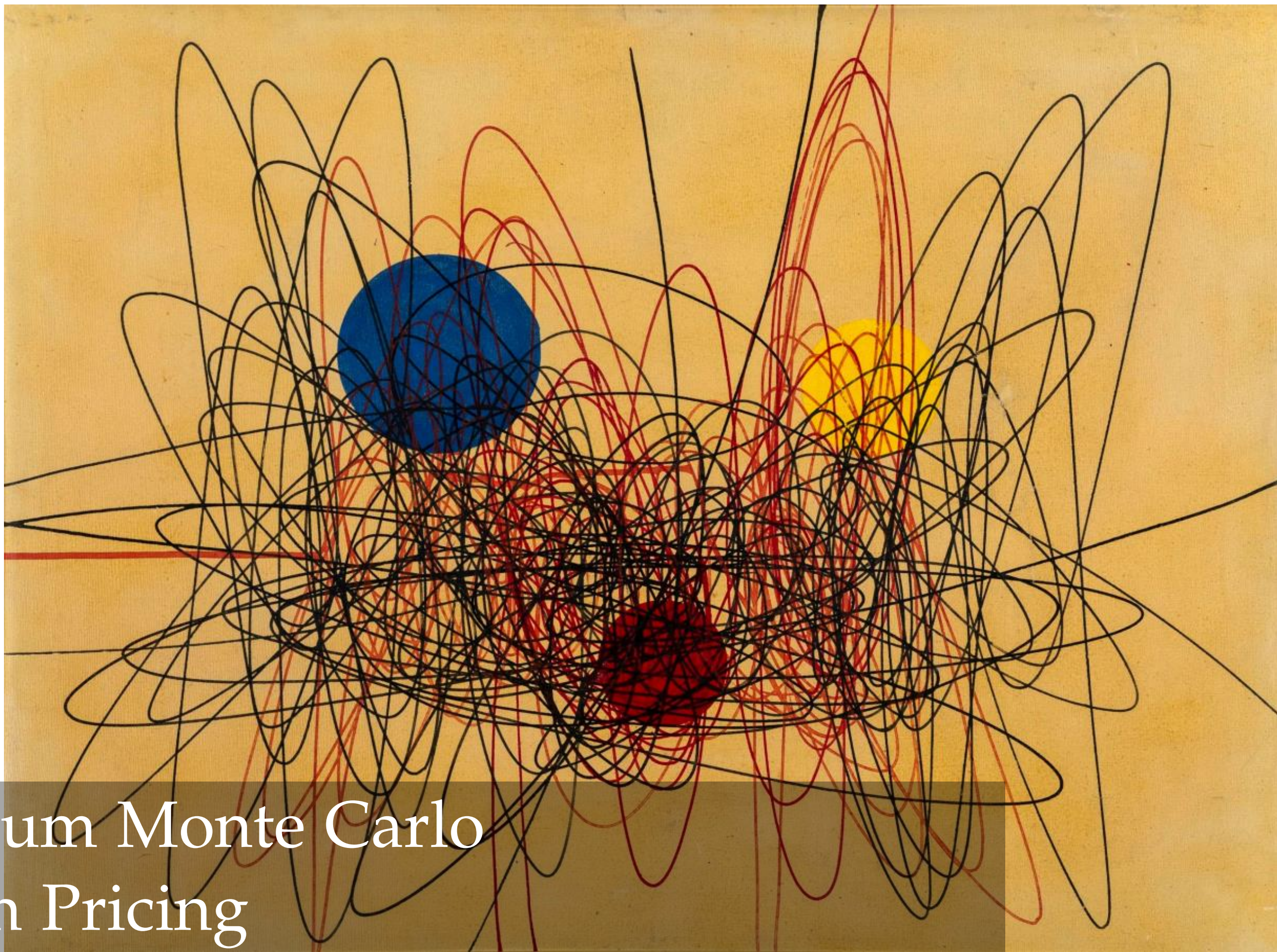
Research Paper Series



Quantum Monte Carlo Option Pricing

May 2023

iason ESSENTIAL SERVICES FOR FINANCIAL INSTITUTIONS

TECHNESTHAI

iason

Research Paper Series

Year 2023 - Issue Number 56

Last published issues are available online:
http://www.iasonltd.com/research

Front Cover: **Roberto Crippa**, *Spirali*, 1951.

Executive Summary

At the end of 20th century, many scientists begin to imagine the possibility to introduce some mechanisms of quantum mechanics into computers. It was proved, at least theoretically, that such mix between quantum mechanics and informatics could take computing power on such levels that problems with an exponential complexity for that time could be easily solved.
It did not take much time that this new field of quantum computation involved other fields such as finance, where many features of quantum computing would change significantly the amount of time and resource used for financial institutions for their business.
Nowadays, quantum computers are still at prototype level but it is already possible to test with simple experiment what a quantum computer can do.
Here, the author have tested a method that is the quantum version of Monte Carlo method, that is a best practice for pricing derivatives instruments and it is shown how it is possible for a Quantum Monte Carlo to obtain same results with a quadratic speed-up respect its classical version.

Table of Content

Quantum Monte Carlo Option Pricing

Alessandro Oriente *

One of the main activities for a bank consists to trade and manage OTC derivatives. Most banks possess huge portfolios of derivatives and they have to price them daily (or even multiple times in a day) to know the effective value of the portfolio at a given date.

Pricing procedure could be very expensive in terms of time and resource, so finding new improved methods for pricing derivatives is fundamental for the next future.

Nowadays international major banks [3] are exploring the new world of quantum computing, a technology that gives properties typical of quantum particles to the well known "bit". This new computational unit is called "*qubit*" (namely quantum bit) and computers that work with qubit are called "*quantum*", to distinguish them from the classical ones. A quantum computer has a remarkably high performance respect to a classical computer, in fact a problem that has an exponential complexity for a classical computer could be solved by a quantum one with a reasonable complexity, say logarithmic (see Shor's algorithm), reducing significantly the time of execution as never seen before.

Hence, quantum computer is promising as a new way to improve the efficiency in pricing options and, in this article, we will compare Monte Carlo method for pricing options, that is a standard in the industry, with its quantum analogue, exploring advantages and limits of the *Iterative Quantum Amplitude Estimation* algorithm [1].

1. A Glimpse of Monte Carlo Method

Monte Carlo is a well known method to generate numbers that represent the outcomes of an arbitrary probability distribution and to make estimation. There is a broad class of algorithms to implement Monte Carlo and the main fields of application are: *optimization*, *numerical integration* and *generating draws from a probability distribution*. This method has been proved to be very suitable to solve high dimensional problems, in fact the error we make from an estimation of some quantity, i.e. the result of an integral, does not depend on the degree of freedom of the problem.

1.1 Monte Carlo Estimation and Error

Generally, Monte Carlo is used to estimate the value of an element of interest, such as the result of an integral or the n-momentum of a distribution.

Suppose we need to estimate the mean value of a random variable $\mathbf{X}$, then we could generate a random sample of N elements $\{\mathbf{x}_i\}_{i=1}^N$ and compute the following:

$$\bar{\mathbf{x}} = \sum_{i=1}^{N} \frac{\mathbf{x}_i}{N}.$$

It can be proved that we can obtain an error from the exact value of order $\mathcal{O}(1/\sqrt{N})$ and it happens that this result does not depend on the specific problem but it is a general result [5].

At the time of the writing of this article, the authors were working for Iason Consulting.

1.2 Pricing Options with Monte Carlo

An option is a particular type of contract where the buyer has the right to receive a payoff at expiry of the contract, which value depends on an underlying, up front a fee that it has to be payed to the seller at the start of it. From the arbitrage-free valuation theory the up-front fee correspond to the discounted expected value of the payoff. Suppose $\Pi(S_T, T)$ is the payoff of the option at maturity T on the underlying S, then the price of the option at any time t is:

$$\Pi(S_t, t) = P(t, T)\mathbb{E}_t^{\mathbb{Q}_T}[\Pi(S_T, T)],$$

where $P(t, T)$ is the discount at time t and $\mathbb{Q}_T$ the T-forward measure. At the start of the contract, $t = 0$:

$$\Pi(S_0, 0) = P(0, T)\mathbb{E}_0^{\mathbb{Q}_T}[\Pi(S_T, T)].$$

As we can see, Monte Carlo can be used to evaluate the expectation value and then the option price, up to a constant factor, with an error $\sim \mathcal{O}(1/\sqrt{N})$ where N represents the number of possible value of S_T generated by the method.

2. Quantum Computation

Quantum computing has started to be notorious only in recent times but its history begins at the end of 20th century, where in 1980 the first idea of Quantum Computer appeared and first quantum algorithms were developed, at least theoretically. From the beginning of 21st century, industries are developing hardware and first prototypes of quantum computer. Though there are some technological issues to build an efficient computer, industries are making progress day by day.

2.1 Limits of Classical Computers

Nowadays, computers can process a massive amount of data through several algorithms to obtain what we require. During their evolution computers are becoming more efficient both in terms of calculation and size, but the underlying logic of a computer stay the same. Every computer, even the most powerful, have to translate data in a series of bits, where the fundamental unit is a bit, a character that can be 0 or 1. Then, bits are processed through logical operators, namely *gates*, to have the result required. Gates are grouped into transistors and as their number into a computer increase, as the computation efficiency increase too. *Moore's law* says that as the number of transistors grows so their size is decreasing, in fact we have experienced how computers become faster and smaller year by year, but we are reaching a physical limit for transistor's dimension which is the atomic size ($\sim 0.2nm$). We need to find new solutions for developing computers for the next future, such as quantum computers that can overcome this problem mixing quantum mechanics and informatics.

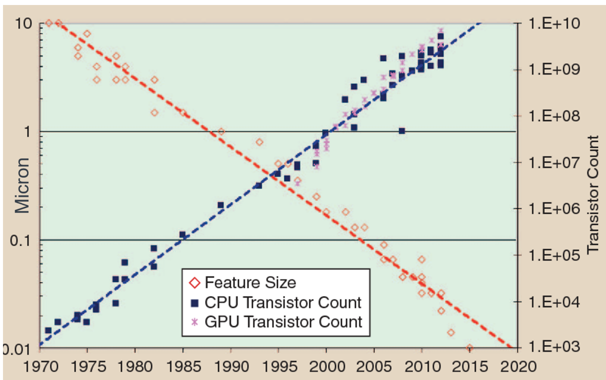


FIGURE 1: Moore's law

2.2 Quantum Computer and Main Properties

As we said before, the fundamental unit of a quantum computer is the qubit and it has properties typical of quantum particles: *superposition*, *entanglement*, *interference*. These properties are the foundation on which it is possible to overcome the limits of a classical computer and, in this section, we describe them briefly:

- Superposition is the most remarkable property of a qubit, in fact it regards the fundamental nature of the qubit itself. While a bit state could be "0" *or* "1", superposition can guarantee to a qubit a single state of "0" *and* "1" at the same time. Hence, a qubit can carry both the logical states. This property permits to carry a significant amount of information all at once.
- In quantum mechanics the entanglement describes a connection between two or more particles where one particle cannot be describe independently of the others, even if they are separated from a large distance. So, a change in one particle can influence the other ones without any further interaction. Qubits, as particles, can be entangled to each others and it permits to work indirectly on a group of qubits manipulating just one or a few of them.
- The fact that qubits are a superposition of "0" and "1" implies that we can have different series of binary code all at once, and for quantum mechanic principles, each of these has a probability to be observed when we want to see the final result of a quantum algorithm. Interference is the property that permits to alter the probability of the final outcome and obtain the desired result.

3. Basic Concepts of Quantum Mechanics

In the previous section, we have introduced some fundamental quantum properties of a qubit. Here we try to give a mathematical explanation of what it is a qubit; it will provide a better comprehension about how a quantum Monte Carlo works.

3.1 Quantum State

We can define a quantum state such as the mathematical entity that provides a probability distribution for the outcomes of each possible measurement on a system. A generic form to represent an n–qubit state is:

$$|\psi\rangle_n = \sum_{j=0}^{2^n-1} e^{i\phi_j}\sqrt{p_j}|b_j\rangle,$$

where $|\cdot\rangle$ are called brackets and they are used to represent a quantum state, $|b_j\rangle$ represents one of the possible binary series with n qubit, p_j is the probability associated to that series, $e^{i\phi_j}$ is called phase and it is a complex number and $e^{i\phi_j}\sqrt{p_j}$ is called *probability amplitude*. For example for a 2-qubit state we can have:

$$|\psi\rangle_2 = \sqrt{p_0}|00\rangle + \sqrt{p_1}|01\rangle + \sqrt{p_2}|10\rangle + \sqrt{p_3}|11\rangle.$$

Quantum qubits can be manipulated through gates in order to modify probability amplitude.

3.2 Quantum Operator

A gate can be represented with a mathematical entity called *operator*. An n-operator can be seen as a linear function that can transform an initial state of n qubits to a final one of the same dimension:

$$|\psi_2\rangle_n = \mathcal{O}|\psi_1\rangle_n,$$

$$|\psi_2\rangle_n = \mathcal{O}(\sqrt{p_1}|\psi_1\rangle_n + \sqrt{1-p_1}|\psi_0\rangle_n) = \sqrt{p_1}\mathcal{O}|\psi_1\rangle_n + \sqrt{1-p_1}\mathcal{O}|\psi_0\rangle_n.$$

As we said before, gates (or equivalently operators) manipulate states in order to modify probability amplitude. To give a simple representation of how an operator works, we consider here a well-known single-qubit gate called *Hadamard* applied to a qubit $|\psi_1\rangle_1 = |0\rangle$:

$$|\psi_2\rangle_1 = \mathcal{H}|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

Hence, a qubit with probability 100% to be "0" can be transformed into a qubit with 50% of probability to be "0" and "1". For an initial qubit $|1\rangle$ we have:

$$|\psi_2\rangle_1 = \mathcal{H}|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

where the sign minus in the last term is given by the phase term. For any single qubit state, that is a combination of $|0\rangle$ and $|1\rangle$, we have:

$$|\psi_2\rangle_1 = \mathcal{H}(\sqrt{p_0}|0\rangle + \sqrt{1-p_0}|1\rangle) = \sqrt{p_0}\mathcal{H}|0\rangle + \sqrt{1-p_0}\mathcal{H}|1\rangle.$$

So, it is possible to transform any state just knowing how $|0\rangle$ and $|1\rangle$ transform and they are called a *basis* for single-qubit state space. In general it can be proved that, given a basis for n-qubit state space, any n-qubit state can be represented as a combination of this basis and transform for any operator as the combination of the results of the operator applied on the basis.

4. Quantum Monte Carlo Method

Following [1] and [4], we will try to show briefly the standard methodology behind the implementation of quantum Monte Carlo, or *QMC*, and the specific algorithm used in this paper: *Iterative Quantum Amplitude Estimation*, or *IQAE*.

4.1 Methodology

In this section we present the general methodology of a class of algorithms called *Quantum Amplitude Estimation*, or *QAE*, that can be used for a QMC problem. Suppose to have $n+1$ qubits in the state $|0\rangle_{n+1}$, where all the qubits are in the state "0" with probability 1. It is possible to define an operator $\mathcal{A}$ for the initial state:

$$\mathcal{A}|0\rangle_{n+1} = \sqrt{1-a}|\psi_0\rangle_n|0\rangle + \sqrt{a}|\psi_1\rangle_n|1\rangle,$$

where a is the probability that the last qubit has value "1", while $|\psi\rangle_n$ represents a generic state of n qubits. QAE methods allow an efficient estimation of a. Then, given an integer M and $M = 2^m$, we can define $k = 2^{i-1}$ and an operator $\mathcal{Q}^k$, for each integer $i \in [1, m]$, for which the following relation holds:

$$\mathcal{Q}^k\mathcal{A}|0\rangle_{n+1} = \cos((2k+1)\theta_a)|\psi_0\rangle_n|0\rangle + \sin((2k+1)\theta_a)|\psi_1\rangle_n|1\rangle, \quad (1)$$

where $a = \sin^2(\theta_a)$. Here, there are different methods to use the equation (1) to estimate θ_a, hence a. IQAE, that is the algorithm used in this paper, try to find a value of k for which it is possible to construct a confidence interval $[\theta_l, \theta_u]$ that holds the following equations:

$$\mathbb{P}(\theta_a \notin [\theta_l, \theta_u]) \leq \alpha,$$

$$\frac{\theta_u - \theta_l}{2} \leq \epsilon,$$

where $1-\alpha$ is the chosen confidence level while ϵ is the error desired for the estimation. Then, it computes the estimation of a:

$$\tilde{a} = \frac{a_u - a_l}{2},$$

$$\mathbb{P}(a \notin [a_l, a_u]) \leq \alpha,$$

$$|a - \tilde{a}| \leq \epsilon.$$

In [1] it is proved that the number of $Q - applications$, that correspond to the number of classical Monte Carlo samples, is inversely proportional to the maximum error ϵ. Hence we have reached the principal advantage of a QMC algorithm respect to the classical one: the maximum error is $\epsilon \sim \mathcal{O}(1/M)$ and it is quadratic faster than a classical algorithm for which we have $\epsilon \sim \mathcal{O}(1/\sqrt{M})$.

4.2 Estimation of Payoff through QAE

Following [4], we can outline the procedure to estimate a generic expected payoff of an option on underlying S: $\mathbb{E}[f(S)]$. The first step is "to load" the distribution probability of the asset. As we have seen, a $n-$qubit state is composed by a group of binary series and their probability to be the final measurement of the algorithm. We can define $|S\rangle_n$ as a binary series of n elements that define one of the possible value of the underlying at maturity, namely S, and p the associated probability, then we can write:

$$|\psi\rangle_n = \sum_{i=0}^{2^n-1} \sqrt{p_i}|S_i\rangle_n. \tag{2}$$

Now the idea is to compute the payoff information inside the amplitude probability. Consider to add a $(n+1)-$qubit initialized at $|0\rangle$ to the state and to apply an operator such that:

$$\sum_{i=0}^{2^n-1} \sqrt{p_i}|S_i\rangle_n|0\rangle \rightarrow \sum_{i=0}^{2^n-1} \sqrt{p_i}\cos\left(c\tilde{f}(S_i) + \frac{\pi}{4}\right)|S_i\rangle_n|0\rangle + \sum_{i=0}^{2^n-1} \sqrt{p_i}\sin\left(c\tilde{f}(S_i) + \frac{\pi}{4}\right)|S_i\rangle_n|1\rangle, \tag{3}$$

where the scaling factor $c \in [0,1]$ and $\tilde{f}(\cdot)$ is the "scaled" payoff given by the function $f(\cdot)$:

$$\tilde{f}(S_i) = 2\frac{f(S_i) - f_{min}}{f_{max} - f_{min}} - 1,$$

where f_{max} and f_{min} are respectively the maximum and the minimum computable value of the payoff. The probability to measure that the last qubit is in the state $|1\rangle$ is:

$$P_1 = \sum_{i=0}^{2^n-1} p_i \sin^2\left(c\tilde{f}(S_i) + \frac{\pi}{4}\right).$$

That is well approximated by:

$$P_1 \simeq \sum_{i=0}^{2^n-1} p_i\left(c\tilde{f}(S_i) + \frac{1}{2}\right) = 2c\frac{\mathbb{E}[f(S)] - f_{min}}{f_{max} - f_{min}} - c + \frac{1}{2},$$

for small values of $c\tilde{f}(S_i)$. Finally, we can estimate the expected value of the payoff through the estimation of P_1 obtained by a QAE algorithm, such as $IQAE$:

$$\mathbb{E}[f(S)] = \left(P_1 + c - \frac{1}{2}\right) \cdot \frac{f_{max} - f_{min}}{2c} + f_{min}.$$

5. Estimation of Basket and Asian Payoff through IQAE

In this section we will show original results obtained by testing the IQAE algorithm on two types of payoff: *Asian* and *Basket* options. To estimate these two options, we made a script with python taking as an example the code in the module *qiskit-finance* on the website *IBMQ* [2] and using an open-source software development kit for working with quantum computers called *Qiskit*. Qiskit can provide a wide set of quantum computers from IBM to compute quantum algorithms remotely. Unfortunately quantum computers are still prototypes and cannot handle a sufficient number of qubits necessary for our experiment (at least for free access computers), so we have managed that by using a *quantum simulator*, that is a classical computer that *simulates* a quantum computer, also provided by Qiskit[2]. Though a quantum simulator has a number of "simulated" qubits greater than a quantum computer has and it can be used freely, without any fee, they cannot be taken in consideration in terms of speed by their classical nature. Hence, in this experiment we try to validate the theoretical result of the order of convergence $\mathcal{O}(1/M)$ without giving any consideration on the speed.

5.1 Computation of Payoff

In the previous section we have shown the general method to compute the payoff in the amplitude probability of the state and then to estimate it with IQAE. Though the procedure was general and consistent with the real procedure used in this experiment, we have omitted a step in the middle that involves a path-dependent or multi-asset payoff. Following [4], consider a basket payoff:

$$f(S_{basket}) = (S_{basket} - K)^+ = (\vec{w} \cdot \vec{S} - K)^+, \tag{4}$$

where $\vec{S}$ is the vector of underlying assets at maturity, $\vec{w}$ is the vector of their weights in basket and K is the strike price. In this case we can recover the equations (2) and (3), in fact in this case we have:

$$|\phi\rangle_n = \sum_{i_1,\ldots i_d} \sqrt{p_{i_1}\cdots p_{i_d}} |S_{i_1}\rangle_{n_1} \otimes \cdots \otimes |S_{i_d}\rangle_{n_d}, \tag{5}$$

where n_j represents the number of qubits to represent the i_j value of the $jth-$asset in the basket ($i_j \in [0, 2^{n_j} - 1]$ is an integer index) and $n = n_1 + \cdots + n_d$ is the total number of qubits used to represent $|\phi\rangle_n$, that is the state for the joint probability of the assets in the basket. To recover (2) we need to add a gate that transform the state as:

$$|S_{i_1}\rangle_{n_1} \otimes \cdots \otimes |S_{i_d}\rangle_{n_d} |0\rangle_{n'} \rightarrow |S_{i_1}\rangle_{n_1} \otimes \cdots \otimes |S_{i_d}\rangle_{n_d} |\vec{w} \cdot \vec{S}\rangle_{n'}.$$

Now, we can recover the general form of (2) for a multi-asset option:

$$|\psi\rangle_{n+n'} = \sum_{i_1,\ldots i_d} \sqrt{p_{i_1}\cdots p_{i_d}} |S_{i_1}\rangle_{n_1} \otimes \cdots \otimes |S_{i_d}\rangle_{n_d} |\vec{w} \cdot \vec{S}\rangle_{n'}. \tag{6}$$

Finally, we can use a relationship similar to (3) to compute the payoff from the last n' qubits in the amplitude probability and then we can estimate it with IQAE. As we can see from (6), this equation can hold even for an asian option, in fact for this type of option we have:

$$f(S_{asian}) = (S_{asian} - K)^+ = (\overline{S} - K)^+, \tag{7}$$

where $\overline{S}$ is the mean value of $S(t)$ from the start to the maturity of the option. For a real option, it is considered a discrete set of observation values for $S(t)$ and, assuming d observation values, we can identify equation (4) in (7) with $\vec{w}$ a $d-$dimensional vector with each element equal to $1/d$. Moreover we do not make any mistake to use (6) for $|S_{i_j}\rangle_{n_j}$ as a set of states for the same asset but at the $j-$observation time. Hence, we can use a similar algorithm for both options and the only difference will be the joint probability.

5.2 Experimental Results

In this section we will show the original results obtained for pricing Asian and Basket options. The first step of the experiment was to compare the estimated value with the exact value[3] for different

[2] Qiskit can provide an API service that connect remotely to quantum computers or simulators from IBM. IBM provides a wide range of them but only a few quantum computers can be used without paying a fee and they are less efficient with a few number of qubits. On the contrary, a quantum simulator access is free and it can be used for testing quantum algorithms though it cannot be compared to a real quantum computer in terms of speed.

[3] the exact value was computed using the discrete formula for the expected value of a random variable following a multivariate lognormal distribution

values of N_{sample}. From figures (2-3-4-5) we observe that as the number of sample increases as the estimation improves, moreover we observed that the error from the exact value is always below the confidence interval obtained from IQAE, that is the maximum error from the algorithm. From (3) and (5) we can observe our most important result: the maximum discrepancy of the estimation has order $\mathcal{O}(1/N_{sample})$ proving that a quadratic speed up respect to a classical Monte Carlo is possible for a quantum computer. The second part of the experiment consisted to test the algorithm for different strike values. As we have seen, for the first part we considered near "at the money" options, hence we will consider an estimation for a range of strikes from "out of the money" to "in the money". As we can see from figures (6-7-8-9), the algorithm works well even for strikes different from ATM, giving in most cases low relative error (about $\sim 2\%$) and the exact values fall into the area generated from confidence intervals.

In figure (8) we can observe that estimation gives high relative discrepancies for out of the money options, while for figure (6) this fact is present but less evident. The reason behind this fact is that IQAE works on a desired *absolute* error, so relative error would be greater for options that have values near zero, just like the ones out of the money. The fact that the effect is more evident for basket options could be a combination of the possibility to choose significant different parameters to specify the dynamic of assets and a poor discretization of probability distribution (we used just 3 qubits for each dimension of the multivariate distribution that correspond of just 9 outcomes for each dimension of the distribution). However this is just a hypothesis and we did not go further on this as it is beyond the scope of the article.

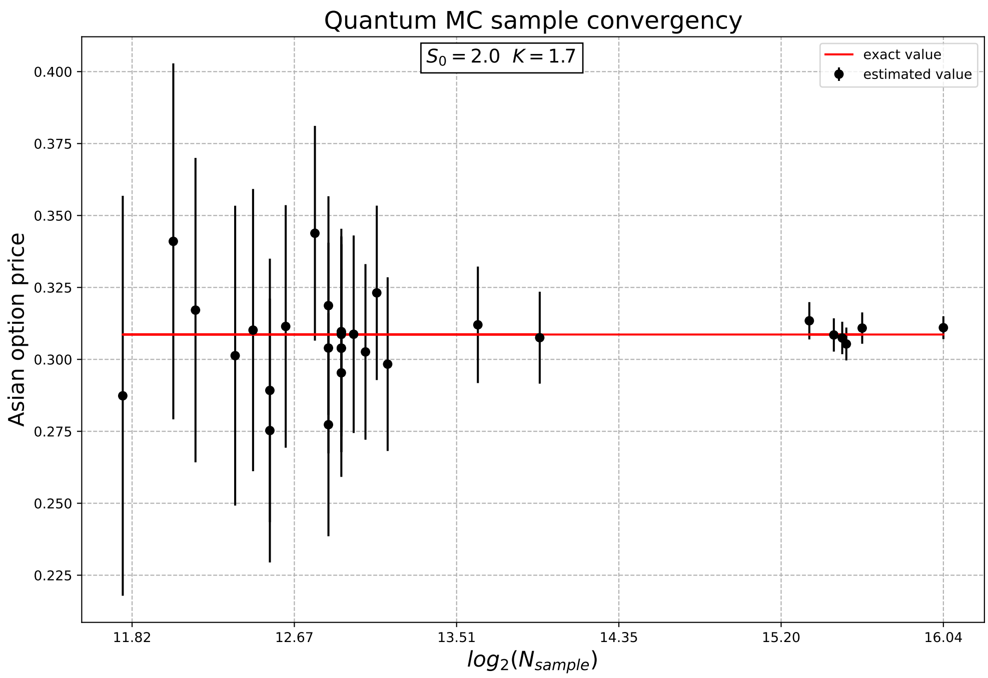


FIGURE 2: Convergence of the estimated value at the exact value. Black lines represent the confidence intervals.

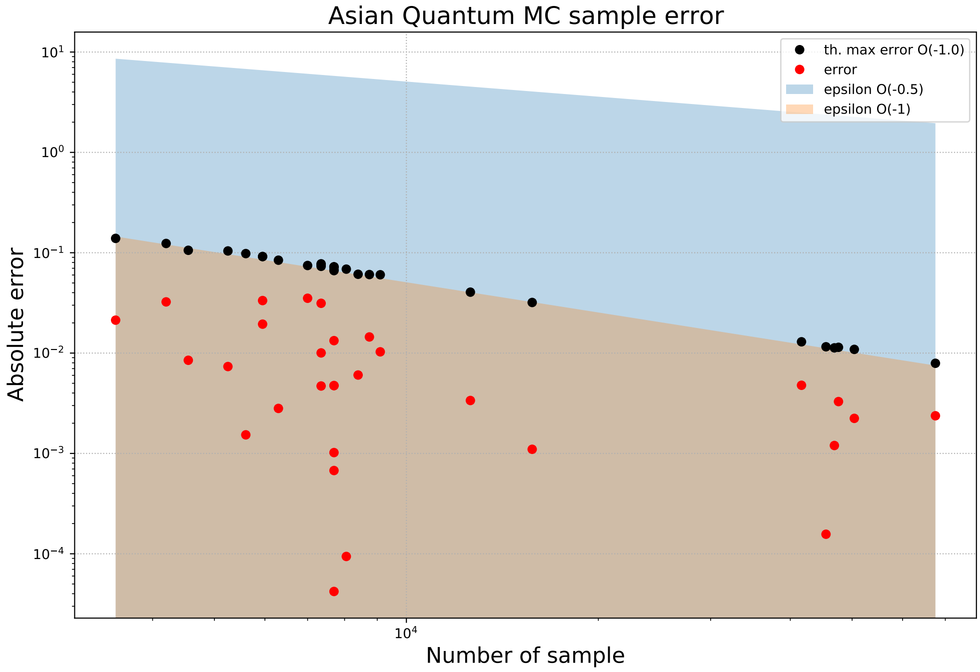


FIGURE 3: Error of the estimated value from the exact value (red dots), confidence interval (black dots).

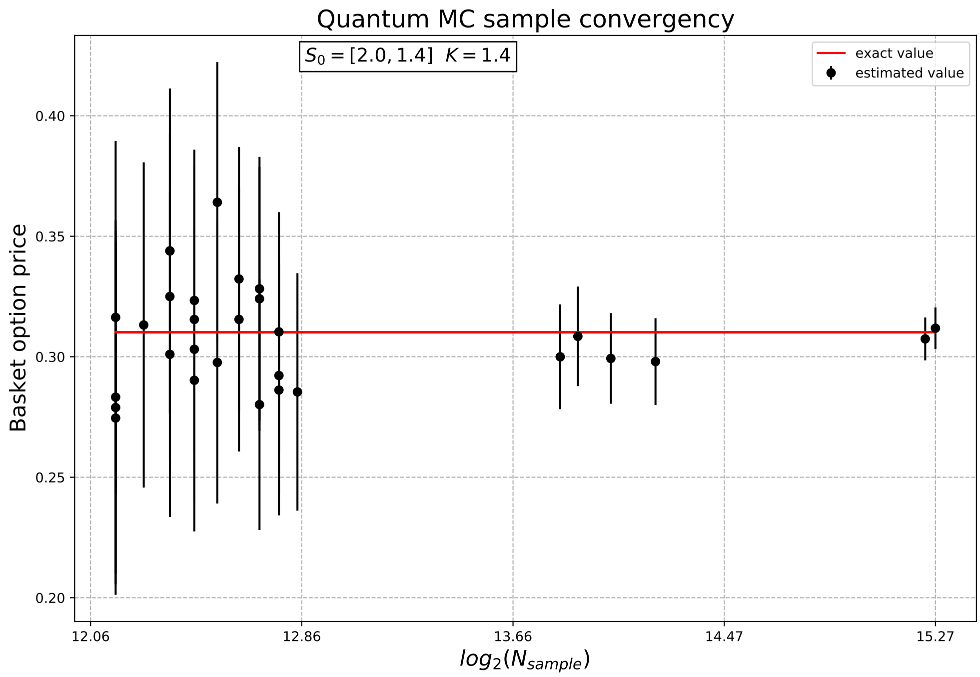


FIGURE 4: Convergence of the estimated value at the exact value. Black lines represent the confidence intervals.

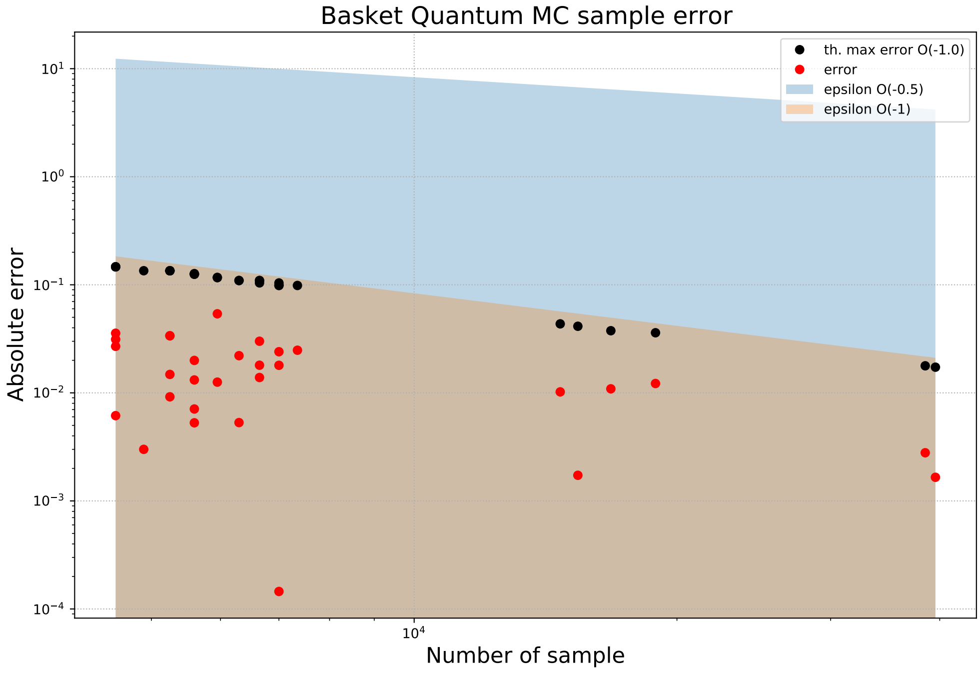


FIGURE 5: Error of the estimated value from the exact value (red dots), confidence interval (black dots).

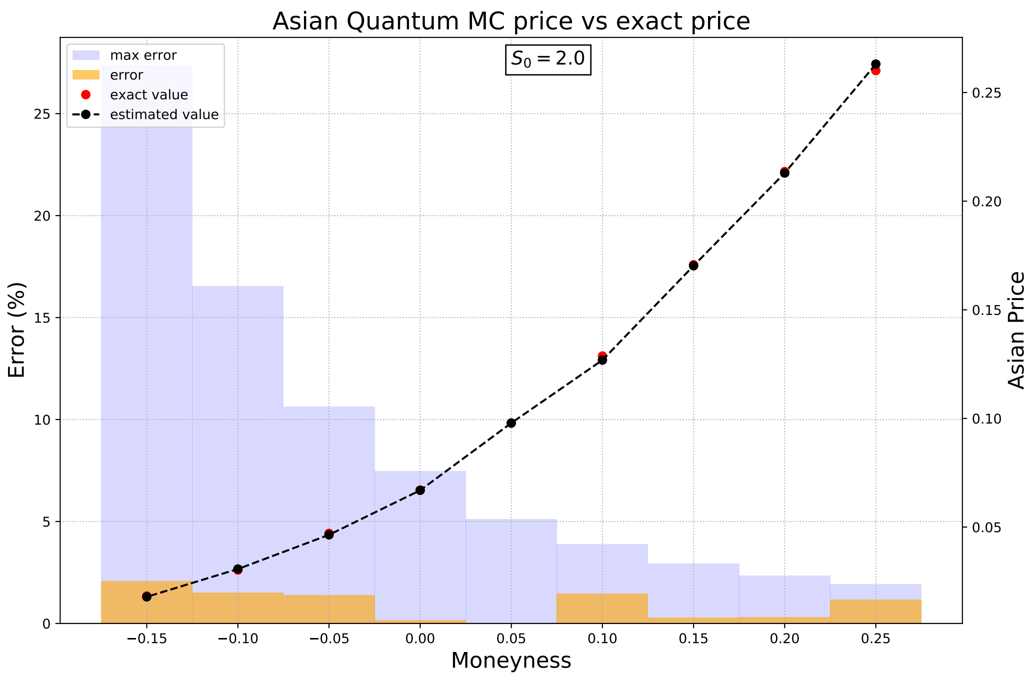


FIGURE 6: Exact price and estimated price (dots). Maximum and observed relative error (bars).

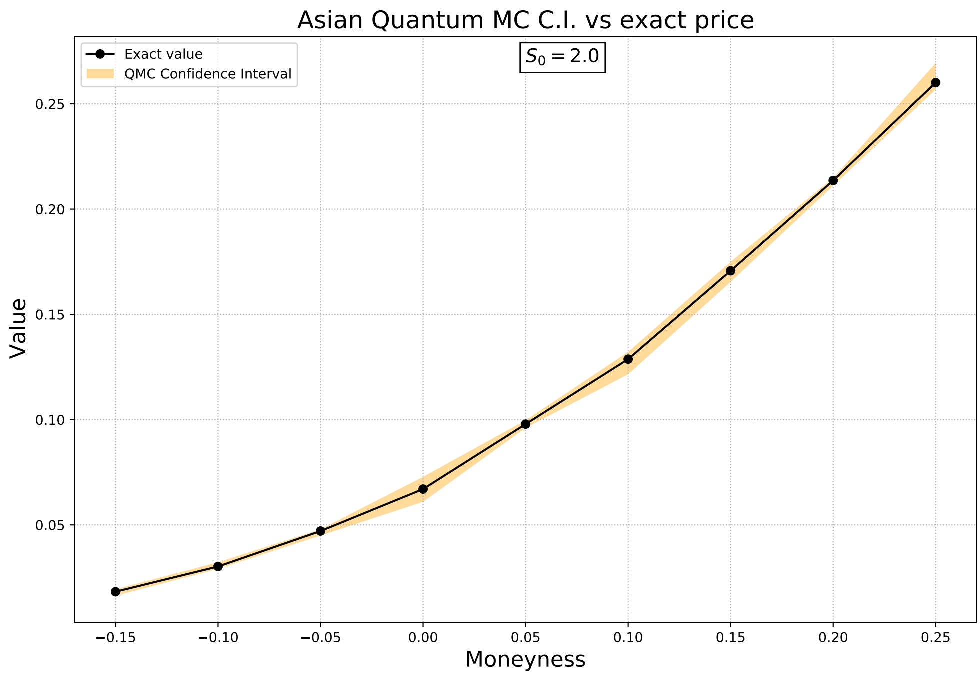


FIGURE 7: Exact price and CI from estimation with IQAE.

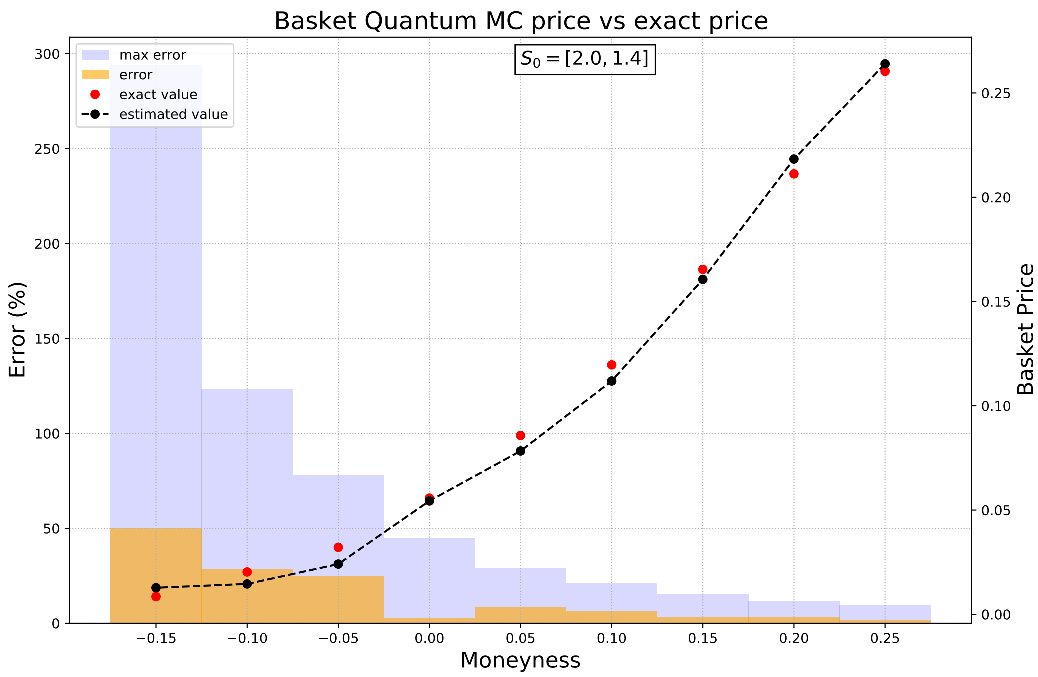


FIGURE 8: Exact price and estimated price (dots). Maximum and observed relative error (bars).

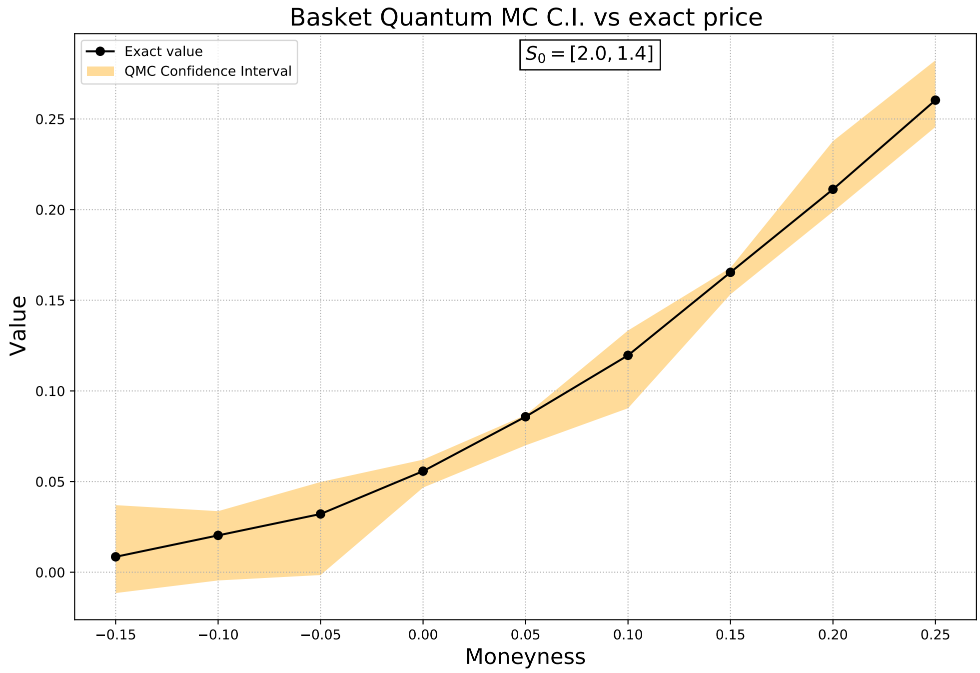


FIGURE 9: Exact price and CI from estimation with IQAE.

Type	Drift	Volatility	Time Obs.	Weights	Correlation (Y/N)
Asian	Equal	Equal	Different	Equal	N
Basket	Different	Different	Equal	Different	Y

TABLE 1: Parameters for asian and basket payoffs

6. Conclusion

In this article we have given the meaning of what a quantum computer is and why many institutions are investing in its development. A quantum computer appears to be both theoretically and practically efficient respect to a classical one in a significant way, in fact it is able to solve high complexity problems, that could be impossible to solve for a classical computer. Though quantum computers are still a prototype facing many problems and limitations, it is already possible to test some quantum algorithm on these devices remotely. In finance big institutions are investing in research on quantum computer for several topics such as portfolio optimization or pricing derivatives. In this article we have shown through our original results how it is possible to implement a quantum Monte Carlo that has a quadratic speed-up respect to the classical one for pricing Asian and basket options. Then we have tested QMC for different strikes and we have observed that expected values fall into the estimated error bound, though we have observed some discrepancy in basket options. In conclusion we have proved that, even if we are far from testing a complete and functional quantum computer, early experiment on quantum Monte Carlo already shows significant improvement in estimation error from the fair value of an option respect to a classical Monte Carlo and we can expect that, as far the research goes on, these improvement could be determinant for pricing derivatives in the next future.

References

[1] **D. Grinko, J. Gacon, C. Zoufal, and S. Woerner** *Iterative Quantum Amplitude Estimation*. IBM Quantum, IBM Research - Zurich, ETH Zurich, University of Geneva, September 2019.

[2] **IBM Quantum** *https://quantum-computing.ibm.com/*.

[3] **McKinsey Company.** *How quantum computing could change financial services*. December 2020.

[4] **N. Stamatopoulos, D. J. Egger, Y. Sun, C. Zoufal, R. Iten, N. Shen, and S. Woerner** *Option Pricing using Quantum Computers*. JPMorgan Chase Co., IBM Quantum, IBM Research - Zurich, ETH Zurich, July 2020.

[5] **Paul Glasserman** *Monte Carlo Methods in Financial Engineering*. Stochastic Modelling and Applied Probability, 53, 2003rd Edition.

iason is an international firm that consults Financial Institutions on Risk Management. iason is a leader in quantitative analysis and advanced risk methodology, offering a unique mix of know-how and expertise on the pricing of complex financial products and the management of financial, credit and liquidity risks. In addition iason provides a suite of essential solutions to meet the fundamental needs of Financial Institutions.

Techneshtai is a technology start-up specialising in providing Tech solutions to the financial sector.
Technesthai supports companies to improve their businesses with bespoken technology ecosystems: the solutions range from Artificial Intelligence and Machine Learning researches to the implementation of custom web-based applications.
Techneshtai is the ideal technology partner to meet the major challenges that digitisation is imposing on the financial sector.

iason ESSENTIAL SERVICES FOR FINANCIAL INSTITUTIONS

TECHNESTHAI