INCLUSIVE PUBLISHING
TRUSTED SCIENCE

PAPER • OPEN ACCESS

Quantum advantage of Monte Carlo option pricing

To cite this article: Zoltán Udvarnoki *et al* 2023 *J. Phys. Commun.* **7** 055001

View the article online for updates and enhancements.

You may also like

Journal of Physics Communications

PAPER

Quantum advantage of Monte Carlo option pricing

Zoltán Udvarnoki[1] , Gábor Fáth[1] and Norbert Fogarasi[2]

1 Department of Physics of Complex Systems, Eötvös Loránd University, Budapest, Hungary
2 Department of Networked Systems and Services, Budapest University of Technology and Economics, Budapest, Hungary

E-mail: zoltan.andras.udvarnoki@ttk.elte.hu

Abstract

Quantum computers have the potential to provide quadratic speedup for Monte Carlo methods currently used in various classical applications. In this work, we examine the advantage of quantum computers for financial option pricing with the Monte Carlo method. Systematic and statistical errors are handled in a joint framework, and a relationship to quantum gate error is established. New metrics are introduced for the assessment of quantum advantage based on sample count and optimized error handling. We implement and analyze a Fourier series based approach and demonstrate its benefit over the more traditional rescaling method in function approximation. Our numerical calculations reveal the unpredictable nature of systematic errors, making consistent quantum advantage difficult with current quantum hardware. Our results indicate that very low noise levels, a two-qubit gate error rate below 10^{-6}, are necessary for the quantum method to outperform the classical one, but a low number of logical qubits (ca. 20) may be sufficient to see quantum advantage already.

1. Introduction

Over the last decades, many algorithms have been proposed for quantum computers that offer a speedup over the best-known classical counterpart [1]. One of them is the Monte Carlo method, for which quantum computers promise a quadratic speedup over the classical Monte Carlo method[2]. Although this speedup is less impressive than the speedup in some other quantum algorithms, the broad applicability of Monte Carlo in physics, mathematics, finance, and many other fields makes it of great interest [3–9]. Hereafter we will refer to Monte Carlo methods utilizing quantum computers to speed up the convergence as 'quantum-accelerated Monte Carlo' or QAMC methods.

At the core of the QAMC method lies the Quantum Amplitude Estimation (QAE) algorithm, which can estimate the amplitude of a quantum state with an error of $\mathcal{O}(1/N)$, where N denotes the number of samples [10]. Over the last years, many different realizations were proposed for QAE with decreasing quantum circuitry footprint [11–15].

Monte Carlo methods are widely used in the financial modeling industry for derivative pricing [5]. One of the simplest examples of a derivative is the European call option which has well-known properties, and an analytic solution under the Black-Scholes model [16]. Pricing options on quantum computers with Monte Carlo was proposed and implemented previously [5, 6].

Errors in the QAMC method were previously analyzed to some extent by Chakrabarti *et al* [17], but they only considered the number of logical qubits needed and did not model the noise in detail. In other papers [18–20] the impact of hardware noise on the statistical errors of the QAMC was analyzed with the assumption of perfect depolarizing noise, but systematic errors of the algorithm were not taken into account.

In this paper, we combine earlier approaches by considering all kinds of noise effects. We propose a novel general metric to quantify the advantage of a specific algorithm over another. In this metric, we connect systematic and statistical errors, so when optimizing for this metric, we can determine and set free parameters in the quantum algorithm to boost performance and create earlier quantum advantage.

In a recent paper, Herbert *et al* [21] came up with an improvement on the previous function approximation techniques using a Fourier series based approach. This Fourier method has been implemented below and was compared to the more traditional 'rescaling' approach. We can demonstrate that the Fourier method leads to smaller errors with clear advantages in option pricing algorithm design.

1.1. Pricing European call options

Our chosen example is the European call option pricing problem. A European call option is a financial contract that gives the holder the right (without obligation) to buy a certain asset at a specific future maturity time T at a predefined strike price K. Financial theory says that the option pricing problem is equivalent to an expected value calculation and, in the general case, can be approximated with Monte Carlo methods [22]. In the industry-standard Black-Scholes-Merton model [23, 24], the distribution comes out as log-normal, so there is a closed-form formula for the price, and this can be compared to any wider scale Monte Carlo implementation to test the efficiency of the latter.

The Black-Scholes log-normal probability distribution of the underlying stock price at time t takes the form:

$$P(S_t;\, S_0,\, r,\, \nu,\, t) = \frac{1}{\sqrt{2\pi}} \, \frac{1}{S_t \, \nu \sqrt{t}} \, \exp\left(-\frac{\left(\ln S_t - \ln S_0 - \left(r - \frac{\nu^2}{2} \right)t \right)^2}{2\nu^2 t} \right), \tag{1}$$

where ν is called the volatility of the underlying, r is the risk-free rate, S_0 is the current price.

The payoff function of the option is:

$$f(S) = \begin{cases} 0 & \text{for } S \leqslant K \\ S - K & \text{for } S > K \end{cases} \tag{2}$$

The option pricing task is to calculate the price of the option, which is just the expected value of the payoff at maturity discounted with the risk-free rate:

$$\mathbb{E}(f(S_T)) = \mathrm{e}^{-rT} \int_0^\infty P(S_T) f(S_T) dS_T \tag{3}$$

1.2. Quantum-accelerated Monte Carlo method

We will briefly discuss the main steps of the QAMC method to introduce the reader to our notation. For a more detailed exposition, see [6].

The QAMC algorithm consists of 3 parts: state preparation, function application, and amplitude estimation.
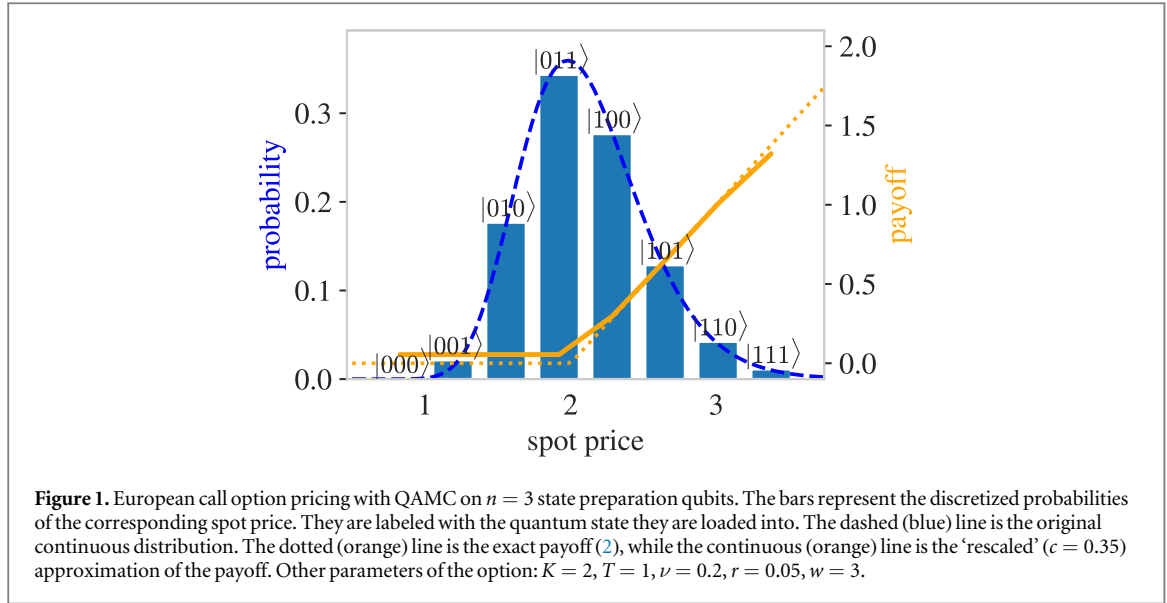
1.2.1. State preparation

The problem has to be discretized for the QAMC method, and we have to truncate the probability distribution to fit into a finite number of bins. The distribution is truncated to the range $[\mathbb{E}(S_T) - w\sqrt{\mathrm{Var}(S_T)}, \mathbb{E}(S_T) + w\sqrt{\mathrm{Var}(S_T)}]$ and discretized into 2^n bins, where w is a free parameter, n is the number of qubits. This discretization is one source of the systematic errors we will keep track of in the sequel.

Then the discrete probability distribution has to be loaded into the amplitudes of the quantum states as visualized in figure 1. In our work, this was done in an exact way with a state preparation circuit for arbitrary states [25].

$$\mathcal{A} \, |0\rangle_n |0\rangle = \sum_{i=0}^{2^n - 1} \sqrt{p_i} \, |i\rangle_n |0\rangle, \tag{4}$$

where $|i\rangle_n$ denotes the n-qubit basis states.

Note that in a typical classical Monte Carlo application in finance, the probability distribution is not known *a priori*, but a stochastic process is posited to evolve the price from time 0 to time T. Usually, this is the hard part of Monte Carlo pricing classically. Here, in our discussion of the QAMC method, we assume that the distribution is known and available in functional form. A full-scale general QAMC implementation should include a fast quantum circuit to generate this distribution and create proper state preparation, but this is a preceding step that we do not consider here.

Figure 1. European call option pricing with QAMC on $n = 3$ state preparation qubits. The bars represent the discretized probabilities of the corresponding spot price. They are labeled with the quantum state they are loaded into. The dashed (blue) line is the original continuous distribution. The dotted (orange) line is the exact payoff (2), while the continuous (orange) line is the 'rescaled' ($c = 0.35$) approximation of the payoff. Other parameters of the option: $K = 2$, $T = 1$, $\nu = 0.2$, $r = 0.05$, $w = 3$.

1.2.2. Function application

Apply the desired function $f(i)$ on the probability distribution:

$$\mathcal{A}\,|0\rangle_n|0\rangle = \sum_{i=0}^{2^n-1} [\sqrt{(1 - f(i))p_i}\,|i\rangle_n|0\rangle + \sqrt{f(i)p_i}\,|i\rangle_n|1\rangle]. \tag{5}$$

The probability of measuring the last (payoff) qubit in the $|1\rangle$ state will be equal to the expectation value of the function on the discrete distribution:

$$P(|1\rangle) = \sum_{i=0}^{2^n-1} f(i)p(i) = \mathbb{E}(f(i)). \tag{6}$$

The piece-wise linear behavior of the payoff function (2) is achieved with a comparator circuit on the quantum computer that sets one ancilla qubit to state $|1\rangle$ only if the state qubits are in state $|i\rangle$, where $S_i > K$. Our implementation of the comparator circuit [6] requires n ancilla qubits.

After the comparator, only the identity function needs to be applied to the payoff qubit. Because exact quantum arithmetic is unfeasible on NISQ devices, some approximation is needed. The 'rescaling technique' for the approximation of the linear part of the payoff function was proposed [6] based on the general method of Woerner *et al* [3]. This technique is based on the fact that the $\sin^2(x)$ function can be applied with a linear number of rotation gates. If we rescale and shift the support of the distribution so that all function values are close to $\frac{1}{2}$, the first term of the Taylor series of $\sin^2(x)$ can be used to generate a linear function:

$$\sin^2\left[x + \frac{\pi}{4}\right] \approx \frac{1}{2} + x, \tag{7}$$

for small x. Rescaling is parametrized by a number c, such that only x values in the range $[-c, c]$ are used.

With repeated measurements of the payoff qubit, we could approximate the expectation value using the quantum circuit in (5). This is the point where we have to use the Quantum Amplitude Estimation algorithm to achieve (quadratic) speedup, as we will see shortly.

We would like to point out that, strictly speaking, the QAMC algorithm solves a different sampling task than its classical counterpart. The expectation value of the function on the discretized probability distribution is transformed into a Bernoulli distribution with the same expectation value, and this latter is the distribution sampled.

1.2.3. Quantum amplitude estimation

The circuit (5) gave us the desired probability on its last qubit. With the QAE algorithm, we can find this probability with the estimation error scaling $O(1/N)$, i.e., quadratically faster than with traditional sampling.

The Grover operator is defined as:

$$\mathcal{Q} = \mathcal{A}\mathcal{S}_0\mathcal{A}^\dagger\mathcal{S}_{\psi_0}, \tag{8}$$

where $\mathcal{S}_0 = 2\,|0\rangle^{\otimes(n+1)}\langle 0|^{\otimes(n+1)} - \mathbb{I}_{n+1}$ and $\mathcal{S}_{\psi_0} = 2|\psi_0\rangle\langle\psi_0| \otimes |0\rangle\langle 0| - \mathbb{I}_{n+1}$.

Application of the m_kth power of the Grover operator on the original state gives:

$$\mathcal{Q}^{m_k}\mathcal{A}\,|0\rangle_n|0\rangle = \cos([2m_k + 1]\theta_a)|\psi_0\rangle|0\rangle + \sin([2m_k + 1]\theta_a)|\psi_1\rangle|1\rangle \tag{9}$$

where $|\psi_0\rangle$ and $|\psi_1\rangle$ are n-qubit normalized quantum states coming from (4), and $\sin^2(\theta_a) = \sum_{i=0}^{2^n-1} f(i)p_i$. The Grover operator is applied in a sequence of $\{m_k\}$ that is called the 'schedule' of amplitude estimation.

The measurement probabilities of the payoff qubit in the computational basis $P_k(|\psi_1\rangle|1\rangle) = \sin^2([2m_k + 1]\theta_a)$ are then sampled, and θ_a is estimated with a method like quantum phase estimation or maximum likelihood.

As mentioned previously, we use n qubits for the state preparation, n qubits for the comparator circuit, and an additional qubit as the payoff qubit. Therefore, the total number of qubits necessary is $2n + 1$.

If the number of quantum samples (shots) is N_{shot} for every Grover step, the operators $\mathcal{A}$ and $\mathcal{A}^\dagger$ corresponding to the problem get evaluated $N = N_{\text{shot}}N_q$ times in total, where $N_q = \sum_{k=0}^{M}(2m_k + 1)$ is the number of queries to the operator $\mathcal{A}$ (or $\mathcal{A}^\dagger$) per shot. The number N is interpreted as the number of samples for the quantum algorithm.

The two most costly and problematic parts are the qubit initialization into the desired discretized distribution and the application of the target function. The first is problematic because its computational cost grows exponentially with the number of qubits. The second one has to be done approximately. Since we use a finite number of qubits, these contribute to systematic errors beyond the usual statistical errors of Monte Carlo.

2. Quantum advantage metric

We can compare the relative efficiency of the quantum and the classical Monte Carlo methods by calculating the ratio of the pricing errors at a fixed sample size. Distinguishing statistical errors from systematic errors, we can write

$$Q^2(N) = \frac{\epsilon_c^2(N)}{\epsilon_q^2(N)} = \frac{\sigma_c^2(N)}{\sigma_q^2(N) + \epsilon_{\text{syst}}^2}, \tag{10}$$

where ϵ_c and ϵ_q denote the full error (statistical and systematic) of the classical and the quantum method, respectively; σ_c and σ_q denote the statistical error only (standard deviation around the expected value of the measurements), while ϵ_{syst} is the systematic error in the quantum method. We can reasonably assume that the classical method is free of systematic errors.

Although later on, we always calculate the above quantity numerically, the behavior of the above metric can be studied analytically.

Consider a method that promises an algebraic (power ζ) speedup over another one at the expense of some systematic error. Then the above-defined metric 10 can be written in the following way:

$$Q^2(N) = \frac{\epsilon_c^2(N)}{\epsilon_q^2(N)} \approx \left(\frac{\tilde{\sigma}_q^2}{\tilde{\sigma}_c^2}\frac{1}{N_q^{\zeta-1}} + \frac{\epsilon_{\text{syst}}^2}{\tilde{\sigma}_c^2}N_qN_{\text{shot}} \right)^{-1}, \tag{11}$$

where N_{shot} is the sample count for the quantum circuit in every Grover step, N_q is the number of queries per shot, $\tilde{\sigma}_c$ and $\tilde{\sigma}_q$ are the standard deviations of the classical and quantum distributions being sampled.

In the presence of depolarizing noise, there is an upper threshold for the number of Grover steps (m_k), where the first term in the previous formula (11) remains constant for increasing N_q [18]. One can easily see that fewer quantum shots (N_{shot}) lead to a greater quantum advantage Q.

If we want to generalize this quantum advantage metric and make it more useful by selecting the same runtime instead of the number of samples, with known classical and quantum runtime per sample τ_c and τ_q, the modified formula reads:
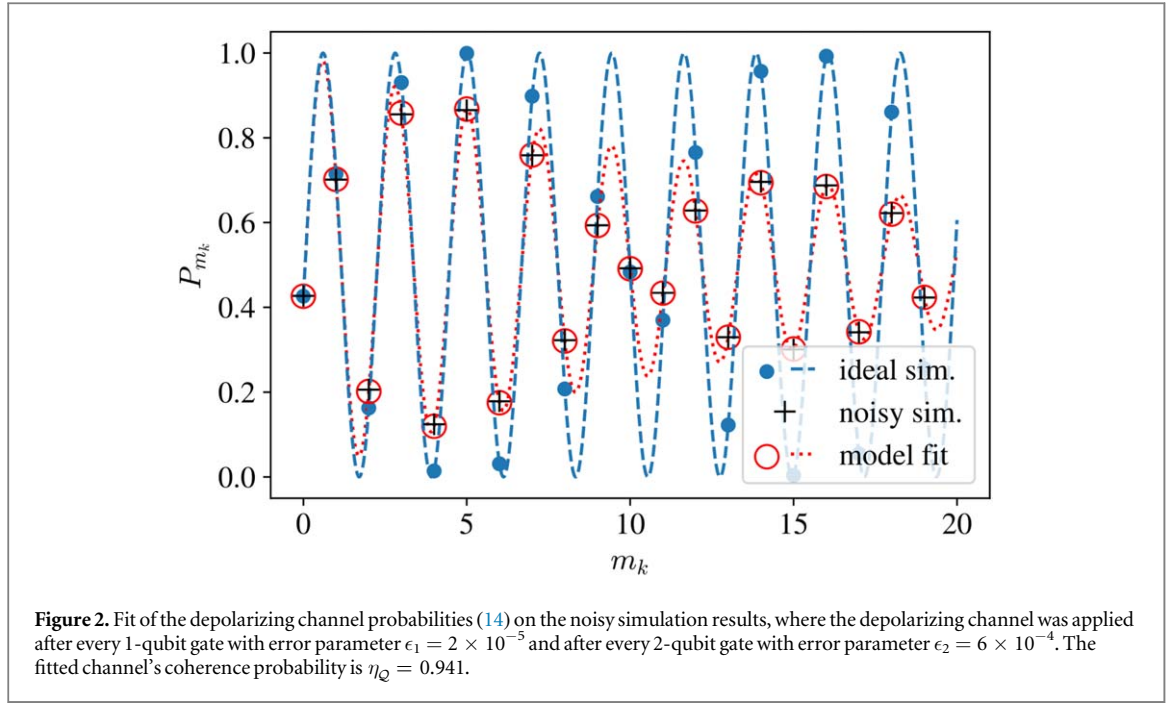
$$Q_T^2 \approx \frac{\tau_c}{\tau_q}Q^2 \tag{12}$$

Where we omitted the time of the measurement and reset operations τ_r, since generally $\tau_r \ll \tau_q$ for practical problems. In the case of the QAE, it also means that the two projection operations in each step are not taken into account since, carefully engineered, they do not contribute much to the overall depth of the quantum circuit.

3. Methods

3.1. Depolarizing noise

The depolarizing noise on n qubits with 'coherence probability' η, or alternatively with depolarizing 'error parameter' $\epsilon = 1 - \eta$, is defined as the following quantum error channel [26]:

Figure 2. Fit of the depolarizing channel probabilities (14) on the noisy simulation results, where the depolarizing channel was applied after every 1-qubit gate with error parameter $\epsilon_1 = 2 \times 10^{-5}$ and after every 2-qubit gate with error parameter $\epsilon_2 = 6 \times 10^{-4}$. The fitted channel's coherence probability is $\eta_Q = 0.941$.

$$\mathcal{D}(\rho) = \eta\rho + (1 - \eta)\frac{I}{2^n}, \tag{13}$$

where ρ is the theoretically exact density matrix of the n perfect qubits after the gate operation in question. This noise channel acting on the readout qubit causes the amplitude amplification operator of m_k iterations ($\mathcal{Q}^{m_k}\mathcal{A}$) to result in a noisy amplified probability

$$\tilde{P}_k = \frac{1}{2} - \frac{\eta_Q^{m_k}}{2}\cos(2[2m_k + 1]\theta), \tag{14}$$

where η_Q is the (effective) coherence probability of the Grover operator. With a bit of approximation, we neglected the small additional error of the $\mathcal{A}$ operation preceding the Grover operator $\mathcal{Q}$. Following the hardware noise through every elementary gate operation would be more realistic. However, as shown in [20], this assumption leads to a very similar overall effect characterized by an effective η_Q. This is demonstrated in figure 2, where we simulated the QAE noise on the gate level and then fitted the result with an effective η_Q on the Grover level with convincing accuracy.

The one and two-qubit gate error parameters, ϵ_1 and ϵ_2, in the depolarizing channel, can be related to IBM's randomized benchmarking measurements by some constant multiplicative factors. For instance, the reported 'Pauli-X error' on their platform is equal to $\frac{1}{2}\epsilon_1$ and the 'CNOT error' is $\frac{3}{4}\epsilon_2$ [27].

3.2. Fourier series based function application approach

A Fourier series based function approximation technique was recently introduced by Herbert *et al* [21]. The technique uses the easily calculable $\sin^2(ax - \theta)$ function to calculate the $\sin(ax)$ and $\cos(ax)$ functions and constructs the payoff function (weighted with the distribution) as a Fourier series from them.

We adapted this technique to European call option pricing. The extensions include taking the effect of the comparator into account and making the linear part of the payoff smoothly periodic.

The main steps of the method are to construct distinct quantum circuits $\mathcal{A}_l$ for the *l*th term in the Fourier series:

$$\mathcal{A}_l |0\rangle_n|0\rangle = \sum_{i=0}^{2^{n-1}} \Theta(x_i - K)\sqrt{p(x_i)} \ [\ \cos\left(\frac{l\pi(x_i - x_0)}{W} - \theta_0\right)|i\rangle_n|0\rangle$$
$$+ \sin\left(\frac{l\pi(x_i - x_0)}{W} - \theta_0\right)|i\rangle_n|1\rangle], \tag{15}$$

where Θ is the Heaviside step function. This can be done easily with a linear number of rotation gates, giving us the probabilities of measuring the payoff qubit in the $|1\rangle$ state:

$$P_l(x_0, \theta_0) = \sum_{x_i > K} p(x_i)\sin^2\left(\frac{l\pi(x_i - x_0)}{W} - \theta_0\right). \tag{16}$$

The method's feasibility relies on the fact that the Fourier coefficients decay as $1/n^3$. This is only true for functions that are continuous and have continuous first derivatives, with their second and third derivatives being piece-wise continuous and bounded. That is why we have to make the linear part of the payoff function continuously periodic and calculate the Fourier coefficients that satisfy the following equations (for the coefficients, see A6):

$$\sum_{l=1}^{\infty}\left[a_l\cos\left(l\frac{2\pi}{W}(x-x_0)\right)+b_l\sin\left(l\frac{2\pi}{W}(x-x_0)\right)\right]$$

$$=\begin{cases} x-x_0-x_{\mathrm{u}}/2 & \text{for } 0<x-x_0<x_{\mathrm{u}} \\ a(x-x_0)^3+b(x-x_0)^2 \\ +c(x-x_0)+d-x_{\mathrm{u}}/2 & \text{for } x_{\mathrm{u}}<x-x_0<W, \end{cases} \tag{17}$$

where x_{u} is the upper limit of the linear part, a, b, c, and d are the coefficients of the third-order polynomial that ensures the sufficiently smooth periodic continuation of the linear function, and W is the period of the function that should be strictly larger than x_{u}.

Following a short calculation (see section appendix A.1), we end up with the following formula of the expectation value for the option price with the above-mentioned Fourier coefficients:

$$\sum_{l=1}^{\infty}[a_l(P_0(0,\pi/2)-2P_l(x_0,0))+b_l(P_0(0,\pi/2)-2P_l(x_0,\pi/4))]$$

$$=\sum_{x_i>K}p(x_i)\left(x_i-x_0-\frac{x_{\mathrm{u}}}{2}\right) \tag{18}$$

From the error analysis of the Fourier technique, it will be apparent that choosing the smallest possible x_{u} will result in the smallest statistical error. By setting the shift $x_0=K$ and $x_{\mathrm{u}}=\max(x_i)-K$, we minimize x_{u} and ensure that only the linear part of the function gets evaluated.

Obviously, we have to choose a threshold l_{max} where we truncate the Fourier series. This will be treated as a free parameter influencing one source of systematic error.

3.3. Statistical error

The noise-dependent variance of the estimates is calculated with the help of the Cramér-Rao error bound for the QAE described in [18]. In a short summary, they create a likelihood function from the measurement probabilities (14) and calculate its Fisher information matrix regarding the parameters η and θ. The Cramér-Rao inequality provides a lower bound on the variance of the estimates:

$$\sigma_{\mathrm{QAE}}^2(\theta,\eta,\{m_k\},N)\geqslant[\mathcal{I}(\theta,\eta,\{m_k\},N)^{-1}]_{1,1}, \tag{19}$$

where $\sin^2(\theta)$ is the estimated probability, $\mathcal{I}$ is the Fisher information matrix, η is the parameter of the depolarizing channel, and $\{m_k\}$ is the schedule of the amplitude estimation.

The noise model is best interpretable in terms of qubit operation error parameters, and these are the parameters in our simulations too. We used the method described in section 3.1 to fit the coherence probability η_Q of the amplitude amplification step (Grover operator) on the simulated results with linear Grover schedule $m_k=\{0,1,2,3,\ldots,20\}$.

When we use the Cramér-Rao bound in place of the variance, we implicitly assume that the estimate is efficient. This is only true in the $N\to\infty$ limit. To take finite sample sizes into account, we made maximum likelihood simulations for a simple 2-qubit case with different numbers of samples. To calculate the standard deviation, the simulations were repeated 1000 times each. We could fit an exponential function on the ratio of the realized standard deviation and the lower bound that will be used later for correction (figure 3).
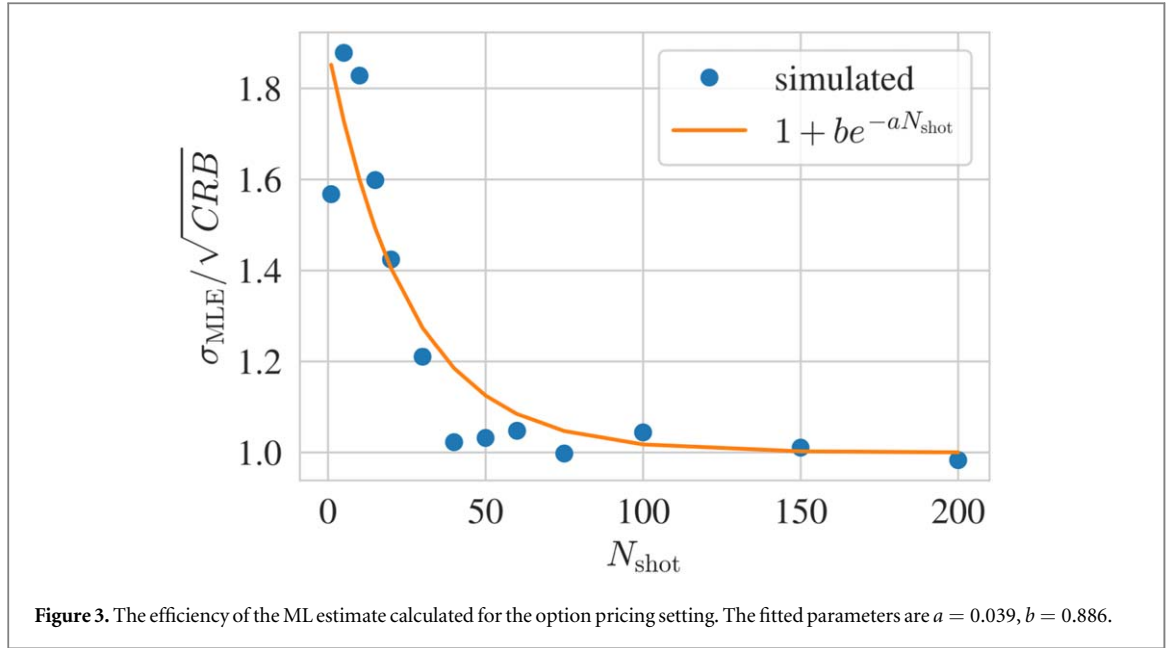
3.3.1. Rescaling technique

In the case of the rescaling technique (as its name suggests), the result of the QAE is rescaled, so the variance is amplified in most cases. If σ_{QAE}^2 denotes the variance of the estimated rescaled probability, the variance of the option price is:

$$\sigma_{\mathrm{R}}^2=\left(\frac{x_{\mathrm{max}}-K}{c}\sigma_{\mathrm{QAE}}\right)^2, \tag{20}$$

where x_{max} is the highest value of the domain after truncation controlled by w, c is the rescaling parameter.

3.3.2. Fourier series based approach

In the case of the Fourier function application technique, we perform multiple amplitude estimation tasks for the $P_0(0,0)$, $P_l(K,0)$ and $P_l(K,\pi/4)$ with standard deviations of the estimated probability σ_0, $\sigma_l^{(0)}$ and $\sigma_l^{(\pi/4)}$,

Figure 3. The efficiency of the ML estimate calculated for the option pricing setting. The fitted parameters are $a = 0.039$, $b = 0.886$.

respectively. The variance of the option price following the previous derivation 18 is:

$$\tilde{\sigma}_{F}^{2} = \sum_{l=1}^{l_{max}} \{a_l^2 [\sigma_0^2 + 4(\sigma_l^{(0)})^2] + b_l^2 [\sigma_0^2 + 4(\sigma_l^{(\pi/4)})^2]\} + \frac{x_u^2}{4}\sigma_0^2 \tag{21}$$

Since the Fourier technique demands additional applications of the circuit $\mathcal{A}$, the number of queries and so the sample size is increased, which is the same as multiplying the variance with $2l_{max} + 1$, i.e., the variance in our formulas will be:

$$\sigma_F^2 = \tilde{\sigma}_F^2 (2l_{max} + 1). \tag{22}$$

3.3.3. Classical case
In the classical case, the variance was calculated from the Black-Sholes model in a closed form (section appendix A.3). The variance of the Monte Carlo estimate can be calculated from the variance of the Black-Sholes option price:

$$\sigma_c^2 = \frac{\sigma_{BS}^2}{N}. \tag{23}$$

3.4. Systematic errors
As systematic errors are exempt from statistical uncertainty, they are easily calculated numerically. To mitigate the effect of large independent systematic errors canceling each other out randomly, we calculated the systematic errors coming from different sources separately and considered the quadratic sum.

The discretization and truncation errors are not easily separated numerically. However, we can obtain an analytic formula for the truncation error (see section appendix A.4) and calculate the discretization error with it.

The total systematic error is calculated as follows:

$$\epsilon_{syst} = \sqrt{\epsilon_{trunc}^2 + \epsilon_{disc}^2 + \epsilon_{func}^2} \tag{24}$$

Where ϵ_{trunc}, ϵ_{disc} are the systematic errors coming from truncating and discretizing the probability distribution, and ϵ_{func} is the systematic error coming from the function approximation.

The above-defined systematic error is then added to the variance (20 or 21) to obtain the total error of the QAMC algorithm:

$$\epsilon_q = \sqrt{\sigma_{R/F}^2 + \epsilon_{syst}^2} \tag{25}$$

3.5. Simulation
All simulations were done with the IBM Qiskit SDK [28]. In noisy simulations, the depolarizing channel was used on one, and two-qubit quantum gates with all qubits treated equally. This noise model, although a

simplification, has a good resemblance to the noise on real devices [18, 29]. The effective coherence probability (η_Q) was fitted to the simulated probabilities as discussed in section 3.1. Where possible, Qiskit's density matrix method was used to obtain exact expectation values even under noise.

In our simulations, we used Qiskit's built-in transpilation with maximum optimization to transpile the circuits into a basis gate set of {CX, X, SX, RZ} defined by Qiskit. These are the standard native gates on IBM quantum computers. The depolarizing noise channel was applied after every gate operation except the RZ gate, similar to real quantum hardware. After one-qubit operations, a noise channel with depolarizing error parameter ϵ_1, and after two-qubit operations (CX gates) with parameter ϵ_2 was applied.

Throughout the simulations, we fixed the ratio between one and two-qubit gate errors to $\epsilon_2/\epsilon_1 = 30$, and the coupling map to all-to-all coupling. We checked different settings, but it had no substantial effect on the fitted noise model effective parameter (coherence probability) η_Q (see figure A1).

Due to increasing computational cost, we found it infeasible to simulate quantum circuits beyond 13 qubits. However, a higher number of qubits can be easily necessary for optimal quantum advantage. To overcome this issue, we used a simple formula (26) to extrapolate to more qubits. If we assume that every application of a CX gate with error parameter ϵ_2 introduces the same effective error to the Grover operator, we can approximate the effective coherence probability with:

$$\eta_Q(n) \approx (1 - \epsilon_2)^{a^n + bn + c}, \tag{26}$$

where n is the number of state preparation qubits. The exponent is the total number of *CX* gates in the Grover operator Q (a^n coming from state preparation, $b_n + c$ from the other parts).

The implementation of the option pricing circuit followed [6] with slight modifications. In the Grover operator, the S_0 projection onto the all-zero state is a multi-controlled Toffoli (MCX) gate that can be very costly if implemented without ancilla qubits: its CX count scales exponentially with the number of qubits [30]. With ancilla qubits, linear scaling is achievable. Luckily we already have enough ancilla qubits because of the comparator, and with the 'v-chain-dirty' implementation of MCX, the dirty ancilla qubits can be reused [31]. Our code is available on GitHub (https://github.com/udvzol/option_pricing).

4. Results

We show results that were made with the exponentially increasing Grover schedule m_k with a base of two, i.e., $m_0 = 0, m_k = 2^{k-1}, \forall k > 0$. The rationale behind this is that this schedule provides the largest quantum speedup, and its fast increase makes numerical optimizations easier. We have to note that other schedules can be more robust against noise [18], but from our metrics point of view, they performed similarly. The maturity of the options was chosen to be $T = 1$ for simplicity. This does not constrain the analysis since by changing the parameters ν and r, the same behavior can be achieved. Also, for simplicity, we consider at-the-money options, i.e., where the current spot price and the strike are the same, which is usually the most relevant setup in finance. In particular, we use $S_0 = K = 2$.
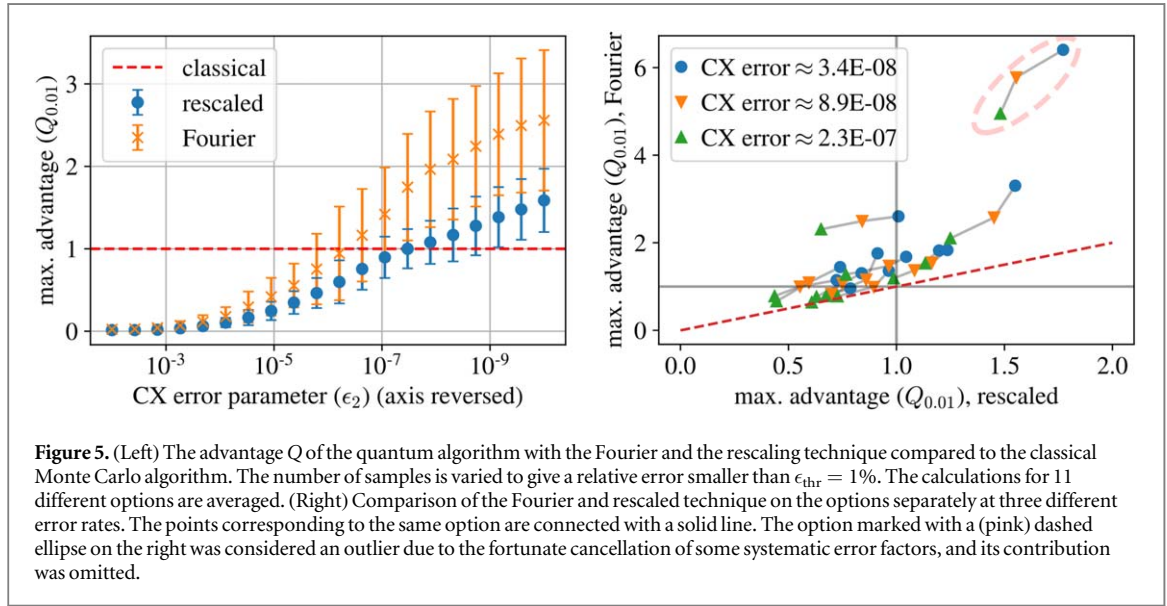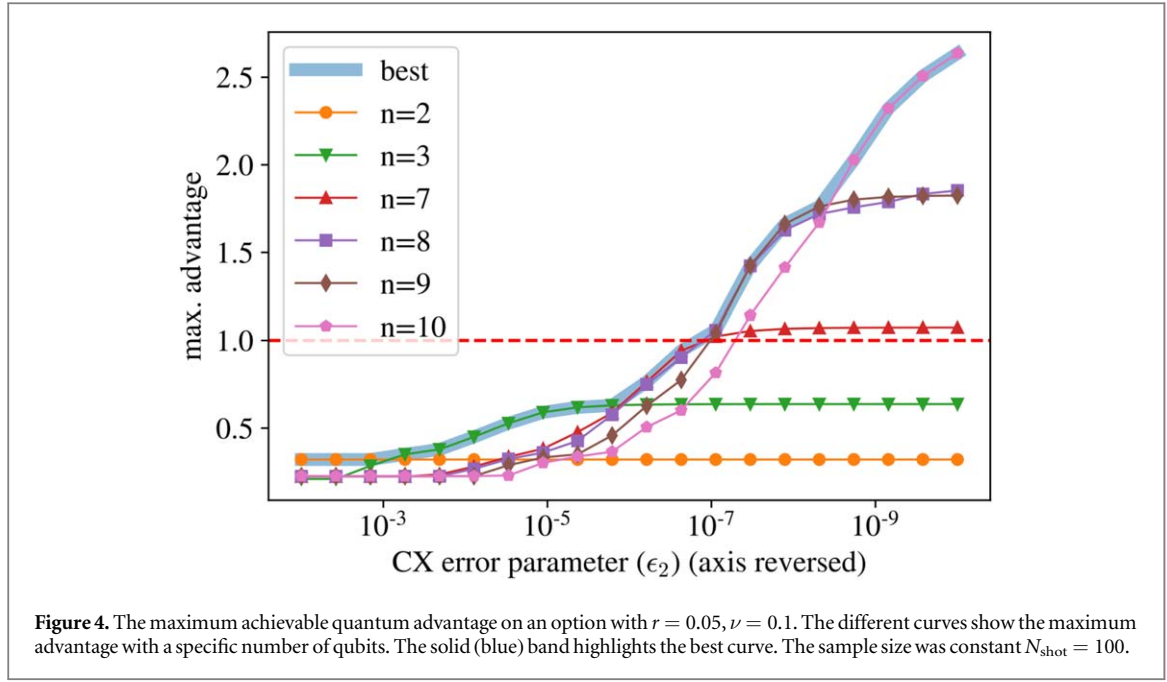
4.1. Numerical optimization of the quantum advantage

We aimed to identify the gate error threshold that makes QAMC advantageous over the classical one. The quantum method has several parameters such as the Grover schedule $\{m_k\}$, the number of state preparation qubits n, the truncation parameter w, and the number of Fourier coefficients l_{max} or, alternatively, the parameter c in the rescaling technique. Optimization for the quantum advantage metric Q made it possible to eliminate the dependence on many parameters and only retain the dependence on the gate error. Of course, there is a trade-off between systematic and statistical errors, but during the optimization, we treat them on equal footing.

In fact, if we were optimizing for the quantum advantage metric Q alone, which is a ratio of error bars, there would be no control over the absolute magnitudes of these errors. To circumvent this, we maximize Q while restricting the individual errors below a certain threshold ϵ_{thr}:

$$Q_{\epsilon_{thr}} = \max_{\epsilon_q < \epsilon_{thr}} (Q) \tag{27}$$
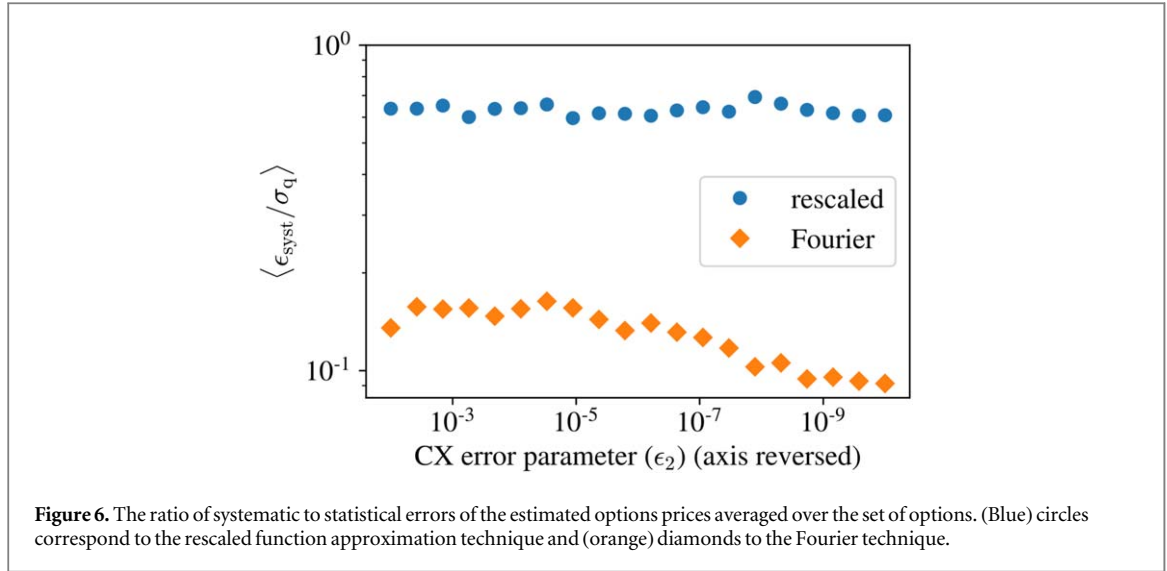
We investigated and optimized quantum advantage on the following parameter grid: n from 1 to 14 with step size 1; 30 evenly spaced values for w in [0.5, 10]; 30 logarithmically spaced values for c in [10^{-5}, 10^0]; l_{max} from 0 to 59 with step size 1, and the maximum of the exponential schedule went from 0 to 2^{23}. We worked with relatively small sample sizes, e.g., $N_{shot} = 100$, changing this parameter to achieve our ϵ_{thr} error constraint. Note, however, that due to systematic errors, the precision cannot be increased beyond a certain point by increasing N_{shot}. Therefore, our interest was rather how to get the maximum out of a small, optimized implementation with a reasonable (finite) number of quantum samples.

Figure 4. The maximum achievable quantum advantage on an option with $r = 0.05$, $\nu = 0.1$. The different curves show the maximum advantage with a specific number of qubits. The solid (blue) band highlights the best curve. The sample size was constant $N_{\text{shot}} = 100$.

Figure 5. (Left) The advantage Q of the quantum algorithm with the Fourier and the rescaling technique compared to the classical Monte Carlo algorithm. The number of samples is varied to give a relative error smaller than $\epsilon_{\text{thr}} = 1\%$. The calculations for 11 different options are averaged. (Right) Comparison of the Fourier and rescaled technique on the options separately at three different error rates. The points corresponding to the same option are connected with a solid line. The option marked with a (pink) dashed ellipse on the right was considered an outlier due to the fortunate cancellation of some systematic error factors, and its contribution was omitted.

The optimal number of state preparation qubits n changes with varying gate errors. In agreement with our expectations, the lower the noise is, the more qubits are favored since it decreases the discretization error. This behavior can be seen in figure 4. For large noise, there is no point in decreasing the discretization error because that would require more qubits and larger circuits and thus would increase the overall error level of the algorithm elsewhere.

To compare the Fourier and the rescaling methodologies and quantify the quantum advantage across several option parameters, we made a grid of options with $r = \{0.05, 0.1, 0.2\}$ and $\nu = \{0.1, 0.2, 0.3, 0.4\}$. The grid choice was inspired by real-world options. Our results show great variability in the achievable quantum advantage due to the unpredictable and erratic impact of systematic errors. Nevertheless, from figure 5 the following conclusions can be drawn:

- For most of the options tested, quantum advantage is achieved when the *CX* error rate goes down to 10^{-6} or 10^{-7}.

- The Fourier technique performs better on every option across a wide range of *CX* errors.

- The relative advantage of the Fourier approach is pretty stable and does not depend on the gate error.

Figure 6. The ratio of systematic to statistical errors of the estimated options prices averaged over the set of options. (Blue) circles correspond to the rescaled function approximation technique and (orange) diamonds to the Fourier technique.

Typical optimal parameters in the region where we see quantum advantage ($Q > 1$) are 5–10 for n, around 10^{-2} for c, 5–10 for w, 10–30 for l_{max}, greatly varying, and 10–20 for the length of the schedule.

4.2. Systematic errors

We found that systematic errors have high variability and can show unexpected behavior. To assess the importance of systematic errors, we calculated the ratio of the systematic to statistical errors in the optimum, see figure 6. The statistical error (standard deviation around the mean) is generally larger than the systematic error, but the ratio remains relatively high, around 0.7 for the rescaled function application technique. For the Fourier technique, the ratio is lower almost by one order of magnitude. The fact that it was possible to decrease the ratio to these levels highlights the benefit of the technique.

5. Conclusion

We examined the efficiency of the QAMC method in a realistic implementation of quantum options pricing. We found that systematic errors can produce large variations as parameters of the options or the quantum algorithm are changed, and this greatly influences the method's feasibility on a small noisy quantum computer.

We found clear evidence that the Fourier function approximation technique works better in the option pricing algorithm. This finding coincides with the limited comparison done by Herbert *et al* [21].

According to our calculations, a CX error level of around 10^{-6} will be needed for quantum advantage to show up in European option pricing. Current IBM computers have a CX error rate of around $10^{-2} - 10^{-3}$. Our target error rates can be achieved most probably with error correction only.

Our finding that about 20 qubits might be sufficient to achieve quantum advantage may contradict the 7500 qubits Chakrabarti *et al* [17] deemed necessary at first sight. However, given the much more complicated problem and objective they considered, the two results complement each other.

The state preparation step has a high impact on both the noise level and the errors from discretization and truncation. The problem can be tackled by quantum generative adversarial networks, which can reduce the computational cost of the step, but introduce another type of systematic error due to its approximate nature [32].

If quantum arithmetic becomes feasible, both the probability distribution evaluation and function evaluation could be done on the quantum computer eliminating most of the systematic errors at the expense of larger noise.

The quantum Fischer information could provide a lower error bound than the classical Fischer information calculated here. However, it is only obtainable through adaptive measurements. Equivalently, modified versions of the Grover operator could lead to higher classical Fischer information [33]. These types of modifications were out of scope for this paper; only the standard algorithm was considered.

It would be worth exploring other types of noise models, but unfortunately, they can be treated only numerically.

Data availability statement

The data that support the findings of this study are openly available at the following URL/DOI:https://github.com/udvzol/option_pricing.

Appendix

A.1. Fourier series based technique

Compared to the original approach, the kink in the options payoff function brings new terms into the Fourier expansion. The following equations (A1), (A2) connect the probabilities measured on the payoff qubit to the cos and sin terms of the Fourier series:

$$\sum_{x_i > K} p(x_i) \sin^2\left(\frac{l\pi x_i}{W}\right) = \sum_{x_i > K} p(x_i) \frac{1}{2}\left(1 - \cos\left(\frac{l2\pi x_i}{W}\right)\right)$$

$$= \frac{1}{2}\left(\sum_{x_i > K} p(x_i) - \sum_{x_i > K} p(x_i)\cos\left(\frac{l2\pi x_i}{W}\right)\right) \tag{A1}$$

$$\sum_{x_i > K} p(x_i) \sin^2\left(\frac{l\pi x_i}{2} - \frac{\pi}{4}\right) = \sum_{x_i > K} p(x_i) \frac{1}{2}\left(1 - \sin\left(\frac{l2\pi x_i}{W}\right)\right)$$

$$= \frac{1}{2}\left(\sum_{x_i > K} p(x_i) - \sum_{x_i > K} p(x_i)\sin\left(\frac{l2\pi x_i}{W}\right)\right) \tag{A2}$$

Inverting A1 and A2 and substituting x_i with $(x_i - x_0)$ gives us:

$$\sum_{x_i > K} p(x_i)\cos\left(\frac{l2\pi(x_i - x_0)}{W}\right) = \sum_{x_i > K} p(x_i) - 2\sum_{x_i > K} p(x_i)\sin^2\left(\frac{l\pi(x_i - x_0)}{W}\right)$$

$$= P_0(0, \pi/2) - 2P_l(x_0, 0) \tag{A3}$$

and

$$\sum_{x_i > K} p(x_i)\sin\left(\frac{l2\pi(x_i - x_0)}{W}\right) = \sum_{x_i > K} p(x_i) - 2\sum_{x_i > K} p(x_i)\sin^2\left(\frac{l\pi(x_i - x_0)}{W} - \frac{\pi}{4}\right) \tag{A4}$$

$$= P_0(0, \pi/2) - 2P_l(x_0; \pi/4) \tag{A5}$$

We use these formulas in the Fourier series to reach equation (18).

A.2. Coefficients of the third-order polynomial

The following coefficients satisfy the conditions required of the third-order polynomial 17:

$$a = \frac{2T}{T^3 - x_u^3 + 3(x_u^2 T - x_u T^2)}$$

$$b = -\frac{3}{2}a(x_u + T)$$

$$c = -3ax_u^2 - 2bx_u + 1$$

$$d = -aT^3 - bT^2 - cT \tag{A6}$$

A.3. Variance of the option return

The Black-Sholes formula for the option price in our notation:

$$\mathfrak{p}_{BS} = S_0 \text{CDF}\left(\frac{\ln(S_0/K) + t(r + \nu^2/2)}{\nu\sqrt{t}}\right)$$
$$- Ke^{-rt}\text{CDF}\left(\frac{\ln(S_0/K) + t(r + \nu^2/2) - \nu^2 t}{\nu\sqrt{t}}\right). \tag{A7}$$

The variance of the option price [34] can be calculated as follows:

$$\sigma^2_{\text{BS}} = S_0^2 e^{\nu^2 t}\text{CDF}(-d + 2\nu\sqrt{t}) - 2KS_0 e^{-rt}\text{CDF}(-d + \nu\sqrt{t})$$
$$+ K^2 e^{-2rt}\text{CDF}(-d) - p_{\text{BS}}^2, \tag{A8}$$

where CDF(x) is the cumulative distribution function of the log-normal distribution (A10) and

$$d = \frac{1}{\nu\sqrt{t}}\ln\left(Ke^{-rt/S_0}\right) + \frac{\nu\sqrt{t}}{2}. \tag{A9}$$

A.4. Truncation error

First, we would like to introduce a couple of useful expressions. In our notation, the expectation value and the standard deviation of the log-normal distribution are:

$$\tilde{r} = S_0 e^{rt}$$
$$\tilde{\nu} = \tilde{r}\sqrt{e^{\nu^2 t} - 1}$$

The maximum/minimum value where the truncation occurs, expressed in standard deviation units:

$$\tilde{x}_\pm = 1 \pm w\tilde{\nu}$$

We can express the strike in expectation value units too:

$$\tilde{K} = \frac{K}{\tilde{r}}.$$

The length of one bin after the discretization for the probability distribution:

$$\Delta = \frac{2w\tilde{\nu}}{2^n - 1}$$

The cumulative distribution function of the log-normal distribution can be expressed as:

$$\text{CDF}(x) = \frac{1}{2}\left(1 + \text{erf}\left(\frac{\nu^2 t + 2\ln x}{\nu\sqrt{8t}}\right)\right) \tag{A10}$$

Another helpful function is what we call here the 'cumulative expectation function' (CEF):

$$CEF(x) = \int_0^x x'p(x')dx' = \frac{1}{2}\left(1 + \text{erf}\left(\frac{2\ln x - \nu^2 t}{\nu\sqrt{8t}}\right)\right) \tag{A11}$$
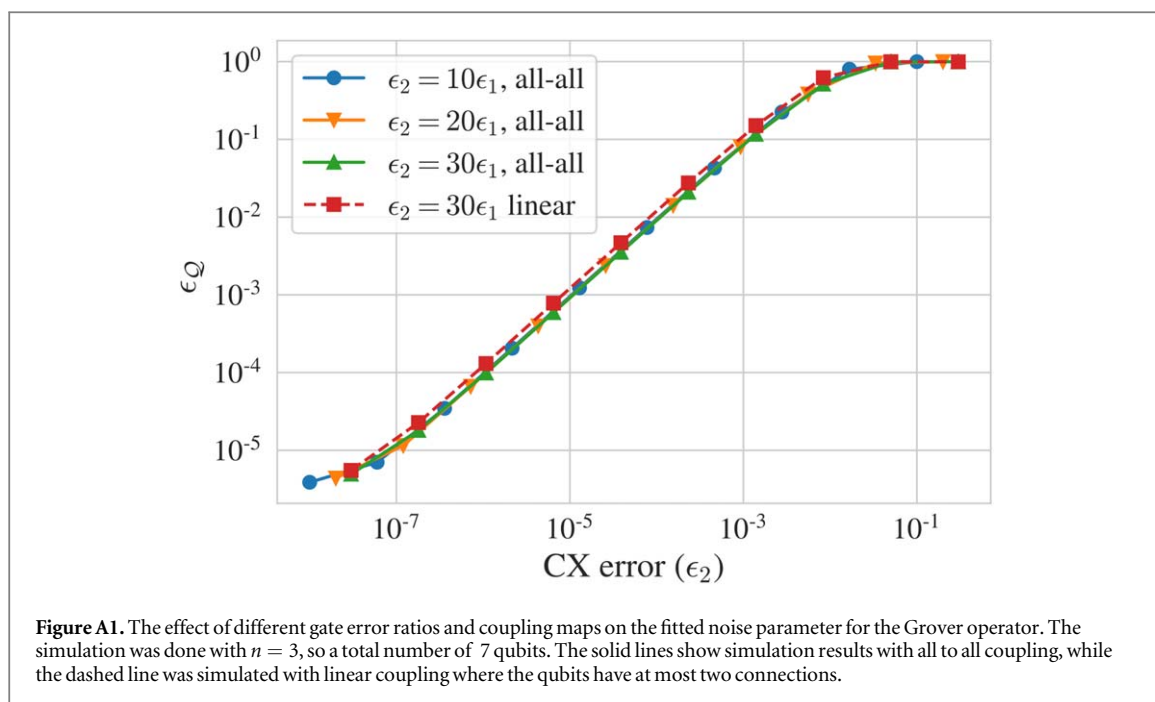
The normalization factor:

$$\mathcal{N} = \text{CDF}\left(\tilde{x}_+ + \frac{\Delta}{2\tilde{r}}\right) - \text{CDF}\left(\tilde{x}_- - \frac{\Delta}{2\tilde{r}}\right) \tag{A12}$$

Using the expressions above the truncation error can be calculated as:

$$\epsilon_{\text{trunc}} = K(\text{CDF}(\tilde{K}) - 1) + K\frac{(\text{CDF}(\tilde{x}_+) - \text{CDF}(\tilde{K}))}{\mathcal{N}}$$
$$+ \tilde{r}\frac{1 - CEF(\tilde{K}) - CEF(\tilde{x}_+) - CEF(\tilde{K})}{\mathcal{N}} \tag{A13}$$

A.5. Different noise models

We found that the noise level of the Grover operator was largely independent of the noise model used on qubit operations, except for the *CX* error parameter (figure A1). This made the *CX* error parameter ideal to describe the noise models. Other parameters were fixed.

Figure A1. The effect of different gate error ratios and coupling maps on the fitted noise parameter for the Grover operator. The simulation was done with $n = 3$, so a total number of 7 qubits. The solid lines show simulation results with all to all coupling, while the dashed line was simulated with linear coupling where the qubits have at most two connections.

ORCID iDs

Zoltán Udvarnoki ⓘ https://orcid.org/0000-0001-7086-235X

References

[1] Montanaro A 2016 *npj Quantum Information* **2** 15023
[2] Montanaro A 2015 *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **471** 20150301
[3] Woerner S and Egger D J 2019 *npj Quantum Information* **5** 15
[4] Orús R, Mugel S and Lizaso E 2019 *Reviews in Physics* **4** 100028
[5] Rebentrost P, Gupt B and Bromley T R 2018 *Phys. Rev.* A **98** 022321
[6] Stamatopoulos N, Egger D J, Sun Y, Zoufal C, Iten R, Shen N and Woerner S 2020 *Quantum* **4** 291
[7] Temme K, Osborne T J, Vollbrecht K G, Poulin D and Verstraete F 2011 *Nature* **471** 87–90
[8] Lemieux J, Heim B, Poulin D, Svore K and Troyer M 2020 *Quantum* **4** 287
[9] Campos R, Casares P A M and Martin-Delgado M A 2022 arxiv:2207.06462
[10] Brassard G, Høyer P, Mosca M and Tapp A 2002 *Quantum Computation and Quantum Information* 305 ed S J Lomonaco Jr. (AMS Contemporary Mathematics) 53
[11] Giurgica-Tiron T, Kerenidis I, Labib F, Prakash A and Zeng W 2022 *Quantum* **6** 745
[12] Grinko D, Gacon J, Zoufal C and Woerner S 2021 *npj Quantum Information* **7** 52
[13] Plekhanov K, Rosenkranz M, Fiorentini M and Lubasch M 2022 *Quantum* **6** 670
[14] Suzuki Y, Uno S, Raymond R, Tanaka T, Onodera T and Yamamoto N 2020 *Quantum Inf. Process.* **19** 75
[15] Aaronson S and Rall P 2020 Quantum approximate counting, simplified *Symposium on Simplicity in Algorithms* (Philadelphia, PA: Society for Industrial and Applied Mathematics) 24–32
[16] Gay G D and Hull J 1990 *The Journal of Finance* **45** 312
[17] Chakrabarti S, Krishnakumar R, Mazzola G, Stamatopoulos N, Woerner S and Zeng W J 2021 *Quantum* **5** 463
[18] Tanaka T, Suzuki Y, Uno S, Raymond R, Onodera T and Yamamoto N 2021 *Quantum Inf. Process.* **20** (9) 293
[19] Herbert S, Guichard R and Ng D 2021 arXiv:2109.04840
[20] Brown E G, Goktas O and Tham W K 2020 1–14 arXiv:2006.14145
[21] Herbert S 2022 *Quantum* **6** 823
[22] Shreve S E 2004 *Stochastic Calculus for Finance I* 1st ed (New York, NY: Springer) Springer Finance
[23] Black F and Scholes M 1973 *J. Political Econ.* **81** 637–54
[24] Merton R C 1973 *The Bell Journal of Economics and Management Science* **4** 141
[25] Shende V V, Bullock S S and Markov I L 2006 *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **25** (6) 1000–10
[26] Nielsen M A, Chuang I and Grover L K 2002 *Am. J. Phys.* **70** 558
[27] Magesan E, Gambetta J M and Emerson J 2011 *Phys. Rev. Lett.* **106** 180504
[28] Qiskit 2020 https://qiskit.org/
[29] Georgopoulos K, Emary C and Zuliani P 2021 *Phys. Rev. A* **104** 062432
[30] Balauca S and Arusoaie A 2022 *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 179–94
[31] Barenco A, Bennett C H, Cleve R, DiVincenzo D P, Margolus N, Shor P, Sleator T, Smolin J A and Weinfurter H 1995 *Phys. Rev. A* **52** 3457–67
[32] Zoufal C, Lucchi A and Woerner S 2019 *npj Quantum Information* **5** 103
[33] Uno S, Suzuki Y, Hisanaga K, Raymond R, Tanaka T, Onodera T and Yamamoto N 2021 *New J. Phys.* **23** 083031
[34] Ben-Meir A and Schiff J 2012 arXiv:1204.3452