



PHOTONICS Research

Efficient option pricing with a unary-based photonic computing chip and generative adversarial learning

HUI ZHANG,^{1,2} LINGXIAO WAN,² SERGI RAMOS-CALDERER,^{3,4} YUANCHENG ZHAN,² WAI-KEONG MOK,⁵ HONG CAI,⁶ FENG GAO,⁷ XIANSHU LUO,⁷  GUO-QIANG LO,⁷ LEONG CHUAN KWEK,^{2,5,8} JOSÉ IGNACIO LATORRE,^{3,4,5} AND AI QUN LIU^{1,2,*} 

¹Institute of Quantum Technologies (IQT), The Hong Kong Polytechnic University, Hong Kong, China

²Quantum Science and Engineering Centre (QSec), Nanyang Technological University, Singapore, Singapore

³Departament de Física Quàntica i Astrofísica and Institut de Ciències del Cosmos (ICCUB), Universitat de Barcelona, Barcelona, Spain

⁴Quantum Research Centre, Technology Innovation Institute, Abu Dhabi, UAE

⁵Centre for Quantum Technologies, National University of Singapore, Singapore, Singapore

⁶Institute of Microelectronics, A*STAR (Agency for Science, Technology and Research), Singapore, Singapore

⁷Advanced Micro Foundry, Singapore, Singapore

⁸National Institute of Education, Nanyang Technological University, Singapore, Singapore

*Corresponding author: aiqun.liu@polyu.edu.hk

Received 25 April 2023; revised 16 July 2023; accepted 31 July 2023; posted 3 August 2023 (Doc. ID 493865); published 27 September 2023

In the modern financial industry system, the structure of products has become more and more complex, and the bottleneck constraint of classical computing power has already restricted the development of the financial industry. Here, we present a photonic chip that implements the unary approach to European option pricing, in combination with the quantum amplitude estimation algorithm, to achieve quadratic speedup compared to classical Monte Carlo methods. The circuit consists of three modules: one loading the distribution of asset prices, one computing the expected payoff, and a third performing the quantum amplitude estimation algorithm to introduce speedups. In the distribution module, a generative adversarial network is embedded for efficient learning and loading of asset distributions, which precisely captures market trends. This work is a step forward in the development of specialized photonic processors for applications in finance, with the potential to improve the efficiency and quality of financial services. © 2023 Chinese Laser Press

<https://doi.org/10.1364/PRJ.493865>

1. INTRODUCTION

The pricing of financial derivatives is a prominent problem that requires extensive computational resources, as the stochastic nature of the underlying assets requires precise modeling. One of the typical financial derivatives is the option, which is a contract that allows the holder to buy or sell assets at a pre-established price (strike) at or before a specified date (maturity date). The payoff of an option relies heavily on the stochastic evolution of asset price. The traditional option pricing model, Black–Scholes–Merton (BSM) [1], usually oversimplifies market dynamics, which limits its practical application to real-life scenarios. As such, numerical methods such as the Monte Carlo method are typically employed for handling more realistic stochastic fluctuations. However, the Monte Carlo method requires extensive computation resources and is slow to predict complicated options. Reducing the computational resources required for models and speeding up option pricing could have significant implications for the financial industry.

Recently, quantum algorithms have shown promise in facilitating computationally hard financial problems such as trading, portfolio optimization, and risk profiling [2,3], and specifically, quantum amplitude amplification can accelerate option pricing with quadratic speedups [4–10]. The unique advantages of quantum algorithms will make up for the shortcomings of classical algorithms to a certain extent, enabling massive high-speed data services in the financial industry. However, current experimental demonstrations using binary approaches and standard quantum circuit models on superconducting devices [11] require dense chip connections and high gate fidelity, making it difficult for practical applications in the near future without a universal quantum computer [12,13]. In addition, superconducting devices require bulky, energy-intensive, and expensive peripherals such as cooling systems, making industrial-scale applications have poor prospects.

For specialized application tasks such as option pricing, there is no need to use universal quantum computers. Photonic

circuits can provide fundamental functions that can be combined to implement specific algorithms [14–20], which would be practical and efficient for user-cased application scenarios. Moreover, the reduced energy costs of photonic computing have been a driving force behind works on dedicated photonic chips for machine learning and algebra [16,21–24]. Therefore, we demonstrate a unary (versus binary) approach in a photonic chip for option pricing. Compared to the binary approach, the unary approach [9] has a remarkably simplified structure and reduced depth of quantum circuits and is especially suitable for linear optical circuit realizations in photonic chips. The unary scheme also allows a post-selection strategy for error mitigation. Additionally, we demonstrate generative adversarial learning to upload the probability distribution implicitly given by data samples into the photonic chip. Generative adversarial learning has previously been demonstrated only in superconducting and optoelectronics devices [25–30]. Compared with traditional Monte Carlo methods, our approach shows high accuracy and significant speedup. It provides a promising avenue for interdisciplinary research in quantum machine learning and financial problems, paving the way for the development of practical photonic processors for quantitative financial applications. It can greatly improve the efficiency and quality of financial services, which is of great significance to the rapid and steady development of the financial industry.

2. CHIP DESIGN FOR UNARY OPTION PRICING

In this work, we focus on European option pricing, and the expected payoff of options is given by

$$C(S_T, K) = \int_K^\infty (S_T - K) dS_T, \quad (1)$$

where S_T is the asset price at time T , and K is the strike price; see European option pricing model in Appendix A. Figure 1 shows the overall scheme of our photonic-chip-based unary approach. The photonic chip [Fig. 1(a)] consists of a generative adversarial network (GAN) and an option pricing part that includes payoff computation and amplitude estimation. In contrast to the classical Monte Carlo approach [Fig. 1(b)] that requires huge computing power to simulate future asset prices to obtain an accurate solution, our approach is expected to show a speedup in the convergence of the standard error of estimated payoff [Fig. 1(c)], which is proved experimentally later in this section results.

The unary approach to option pricing encodes an asset price distribution into the unary basis of a quantum register, as shown in Fig. 2. A binning scheme is applied such that Monte Carlo paths that would belong to the same interval of asset prices end up in the same bin. Each bin is then mapped to an element of the unary basis, whose coefficient is the ratio of the number of Monte Carlo paths in that bin to the total number. The accuracy of unary encoding is bounded by the number of bins that can be stored in a quantum state, i.e., the usable dimension of the high-dimensional unary state. Based on the unary basis, Fig. 3(a) depicts the algorithmic model for unary option pricing, which consists of three modules: a distribution loading module \mathcal{D} that loads the asset price distribution into a quantum state, a payoff calculation module \mathcal{P} that computes the expected return, and a quantum amplitude estimation

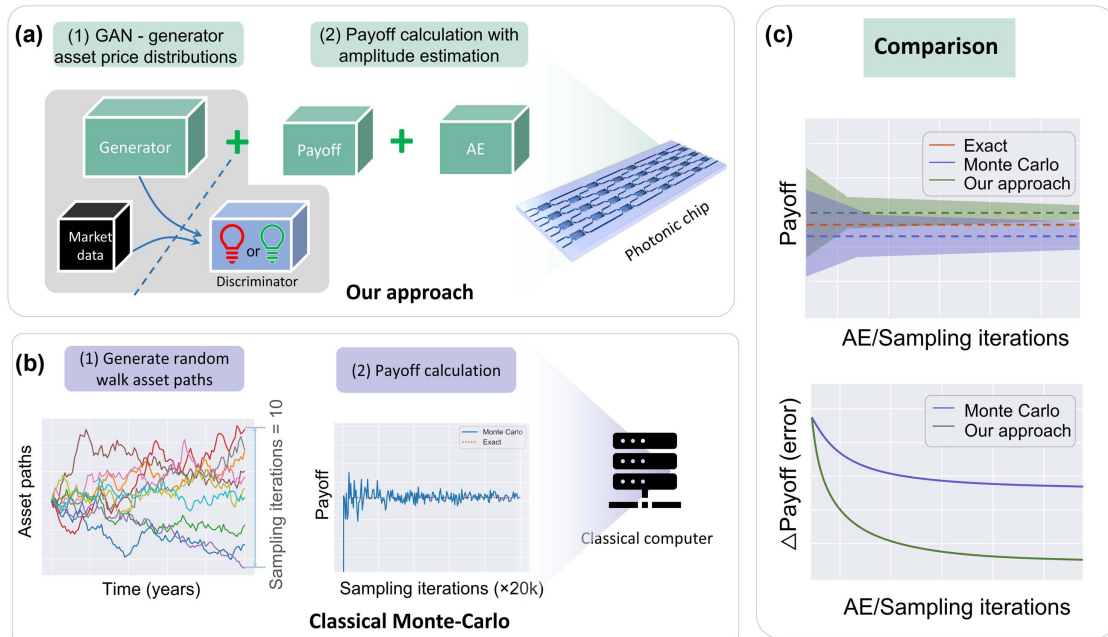


Fig. 1. Schematic of the unary approach to option pricing, compared to the classical Monte Carlo method. (a) Integrated photonic chip with the unary algorithm, consisting of a generator of the generative adversarial network (GAN), payoff calculation, and quantum amplitude estimation for acceleration. (b) Monte Carlo simulation on a classical computer, which first generates the future asset price paths based on random variables, and then calculates the payoff. The accuracy relies on extensive simulations of random walk asset paths. (c) Expected acceleration of the convergence of payoff errors, compared to classical Monte Carlo simulations. Shaded areas in the top inset indicate statistical uncertainty.

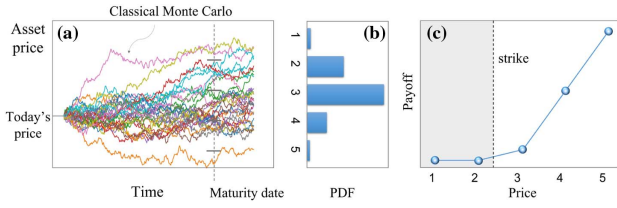


Fig. 2. Mapping of asset prices to unary basis. (a) Classical Monte Carlo paths partitioned into different unary bases. (b) Probability density function (PDF) according to the defined unary basis. (c) Payoff value calculated according to the PDF and asset prices.

module Q to gain quadratic speedup over classical sampling to reach a target accuracy.

Figure 3(b) depicts the optical circuit model, whereby each module of the unary algorithm is mapped to a linear optical operator. We represent the high-dimensional state by path encoding a single photon using n optical waveguides. The superposition of a single photon traveling through different waveguides directly encodes the unary basis. This high-dimensional state can be written as $|\psi\rangle = \sum_{i=0}^{n-1} \sqrt{p_i} |i\rangle$, where p_i represents the probability of observing a photon in the waveguide mode $|i\rangle$, and these probabilities conform to $\sum_{i=0}^{n-1} p_i = 1$. The payoff calculation requires an ancilla qubit to store the expected

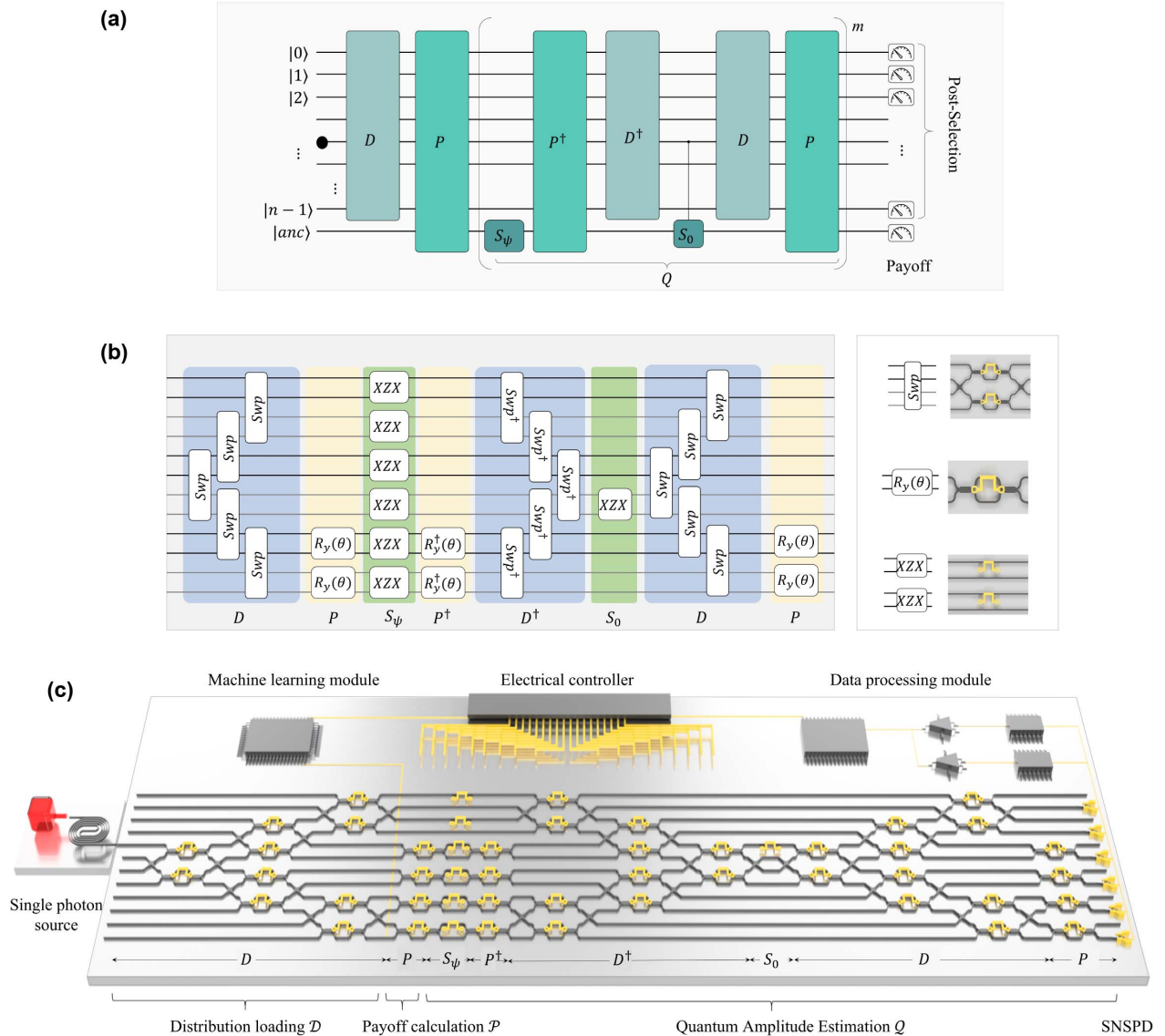


Fig. 3. Photonic chip design for the unary option pricing algorithm. (a) Algorithmic model of unary option pricing. The input state consists of an n -dimensional qudit and a two-dimensional ancilla. The following modules are contained: D , distribution loading; P , payoff calculation; Q , quantum operator for amplitude estimation. The amplification module Q is performed sequentially by $S_\psi \rightarrow P^\dagger \rightarrow D^\dagger \rightarrow S_0 \rightarrow D \rightarrow P$. The expected payoff is obtained by measuring the ancilla. (b) Optical circuit model by transforming the algorithmic model to linear optical operators. Each element of the unary basis is represented by two waveguides, extending the n -bin unary basis to $2n$ -dimensional Hilbert space. Relevant linear optical operators swp, $R_y(\theta)$, and XZX are listed with their waveguide structures. (c) Photonic chip design and architecture. The chip is designed by transforming the optical path model into waveguide structures and realizes the distribution loading, payoff calculation, and amplitude estimation sequentially. The distribution loading is trained as a GAN embedded in the machine learning module.

return for each asset price, expanding the Hilbert space of the algorithm to $2n$. To avoid non-local controlled gates in the photonic chip implementation, we instead add an ancillary waveguide to each of the n unary waveguide modes to represent the effect of the ancilla qubit. Each element of the unary basis is now represented by two waveguides. This way, the controlled operations of the original algorithm are converted to linear transformations on the optical circuit. The architecture of the photonic processor with the detailed chip design is shown in Fig. 3(c), which replaces each linear optical operator with the corresponding waveguide structure. The entire chip is reconfigurable via wire bonds and integrated thermo-optic phase shifters; see the experimental setup in Appendix B.

In the distribution loading module \mathcal{D} , a single photon is incident into the chip from a waveguide in the middle of the circuit, which encodes the ancilla in its $|0\rangle$ state, e.g., for a three-asset case, the initial input state can be written as the tensor product of the middle unary qudit and the ancilla qubit as $[0, 1, 0] \otimes [1, 0] = [0, 0, 1, 0, 0, 0]$. The distribution of asset prices is then uploaded to the different waveguides using a linear depth circuit. This distribution loading circuit spreads the superposition to the neighboring basis using swp operators:

$$\text{swp} = \begin{pmatrix} I & & \\ & \sqrt{p} & \sqrt{1-p} \\ & \sqrt{1-p} & -\sqrt{p} \\ & & & I \end{pmatrix} \otimes I, \quad (2)$$

where p depends on the target distribution. The procedure is repeated until the edge of the circuit is reached. The distribution loading module can be reconfigured to obtain any target probability distribution in the unary representation. Precisely, given n assets, the depth of the circuit is always $\lfloor (n+1)/2 \rfloor$, and the loading of any known probability distribution onto the unary basis depends on $(n-1)$ splitting parameters p . The generator of a GAN is embedded in this module. The GAN is employed to capture the probability distribution underlying given market data. The details are presented in the next section.

The payoff calculation module \mathcal{P} encodes the expected payoff as the probability of measuring the photon in the waveguides encoding the ancilla in state $|1\rangle$, using rotation operations between the two waveguides of each element of the unary basis. The rotations encode the expected return for each asset price in the distribution. This action, labeled \mathcal{P} , can be written as

$$P = \begin{pmatrix} M_0 & & & \\ & M_1 & & \\ & & \ddots & \\ & & & M_{n-1} \end{pmatrix}, \quad M_i = \begin{pmatrix} \cos \theta_i & -\sin \theta_i \\ \sin \theta_i & \cos \theta_i \end{pmatrix} \quad (3)$$

for a $2n$ -waveguide, n -bin example.

A quantum amplitude estimation module \mathcal{Q} is applied to achieve quantum speedups. Various amplitude estimation techniques have been presented that are friendly to NISQ devices [31–33]. Here, we implement an amplitude estimation algorithm without quantum phase estimation in the photonic circuit, following the technique used in Ref. [9]. Increasing steps of amplitude amplification are applied to estimate the

relevant amplitudes with up to a square root advantage oversampling from the original distribution. This amplification module \mathcal{Q} is performed by applying the following operators. First, S_ψ identifies the amplitudes that encode the expected payoff and reverses their signs. Explicitly, for the three-asset example at hand, such an operation is $S_\psi = \text{diag}(1, -1, 1, -1, 1, -1)$, and is realized experimentally by applying a phase shift of π on the second waveguide of each element of the unary basis. Then, the original operations are reversed, that is, the inverse of the payoff calculator \mathcal{P}^\dagger and the distribution loading \mathcal{D}^\dagger are applied. An operator S_0 follows, which reverses the sign of the initial state of the computation. Experimentally, it is applied by introducing a phase shift of π to the waveguide where the photon was introduced. The last step is to repeat the distribution loading \mathcal{D} and the payoff calculator \mathcal{P} modules. The amplitude amplification operator $\mathcal{Q} = \mathcal{P} \cdot \mathcal{D} \cdot S_0 \cdot \mathcal{D}^\dagger \cdot \mathcal{P}^\dagger \cdot S_\psi$ is repeated a different number of times, and the results are processed to estimate the expected payoff. This technique provides up to quadratic speedups over ordinary sampling in the number of calls to the \mathcal{D} and \mathcal{P} operators to reach the same confidence level; see theoretical derivations in Appendix C.

3. GAN FOR DISTRIBUTION UPLOADING

A GAN is implemented in the distribution loading module with on-chip training for real-time noise perception. The goal of the GAN is to obtain an intelligent generator at the chip parameter level that captures the probability distribution behind the given market data without simulating enormous random paths, accumulating data statistics, and then fitting them into the chip architecture. With the GAN, we can efficiently load the classical data, i.e., the probability distribution underlying market data, into quantum states and obtain more precise payoff calculations with the presented unary option pricing methods.

GANs train a generator (G) to synthesize semantically meaningful data from standard signal distributions, as well as a discriminator (D) to distinguish real samples in the training dataset from fake ones produced by the generator [34], as depicted in Fig. 4(a). As its adversary, the generator aims at deceiving the discriminator by producing more realistic samples. Training GANs involve the search for a Nash equilibrium of a two-player game between a generative and a discriminative network, which can be formulated as

$$\min_G \max_D \mathbb{E}_{x \sim p_{\text{real}}} (\log(D_\phi(x))) + \mathbb{E}_{z \sim p_z} (\log(1 - D_\phi(G_\theta(z)))), \quad (4)$$

where the generative network G_θ takes noisy samples z from a normal or uniform distribution p_z as input, and x comes from the real distribution p_{real} . The discriminative network D_ϕ tries to distinguish the generated (fake) sample $G_\theta(z)$ and real sample x by projecting their output to $\{0,1\}$. θ and ϕ are free parameters that construct the generator and discriminator. The training procedure is complete when the generator wins the adversarial game, that is, the discriminator cannot make a better decision than random guesses on the validity of a sample.

We develop a hybrid GAN implementation that consists of a generator network in the photonic chip, a classical discriminator network, and a control system that communicates

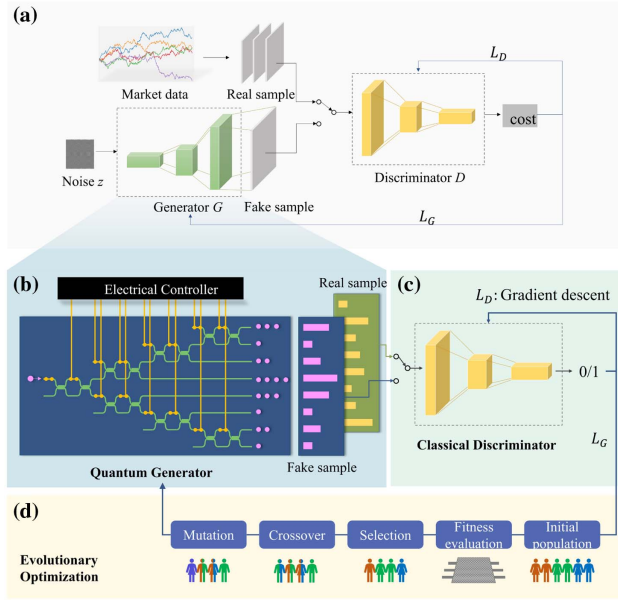


Fig. 4. GAN on the photonic chip for precise asset distribution uploading. (a) Algorithm of GAN, composed of a generator and a discriminator. (b) Generator implemented by a variational photonic circuit, which is trained on-chip in real time. The probability distributions accumulated on the waveguide paths are used as fake samples. Real samples are the training targets taken from market data in real applications. (c) Classical discriminator consisting of sequential convolutional layers and trained by a gradient descent algorithm. The discriminator aims to distinguish the source of the input sample, from the generator or a real distribution. The cost function is calculated from the discriminator output and used to train the discriminator itself and the generator. (d) The generator is trained by an evolutionary optimization procedure where populations (e.g., different configurations of the generator ansatz) are generated, evaluated, and iterated. The evaluation is accomplished using the scores granted by the discriminator. New generations are produced via the operators of selection, crossover, and mutation of current populations.

between the classical computer and photonic chip, all depicted in Figs. 4(b)–4(d). The generator is parameterized by the angles on the phase shifters that are reconfigurable through the thermo-optic effect, induced by applying tiny electrical power to the integrated heaters. Instead of a noise distribution as input, we utilize the uncertainty of photons appearing at different waveguide modes to achieve the equivalent randomness for the generator. The fake samples are the probability distribution of the photons at different waveguide modes. The real samples are drawn from the desired probability distribution, a log-normal or normal distribution for the examples presented in Fig. 5. Fake and real samples sequentially enter the classical discriminator to achieve the classification results. The discriminator is a classical neural network implemented with TensorFlow. We then explicitly discuss the training of the GAN with data samples drawn from the log-normal distribution and the normal distribution.

The training process of the GAN in a photonic chip introduces two challenges, the difficulty of obtaining gradients due to the stochastic nature of measurements, and the phenomenon

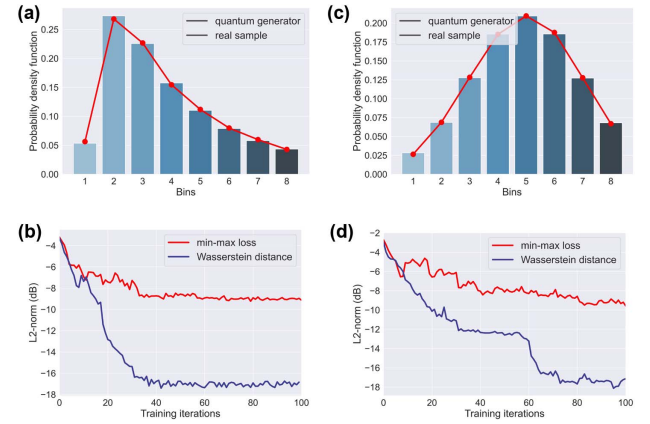


Fig. 5. Experimental training performance of the GAN under Wasserstein distance. (a), (c) Comparison between the probability distributions obtained experimentally from the generator (solid line with data points) and the target distribution (histogram). (b), (d) Evolution of the ℓ_2 norm between the fake and real samples with increasing training iterations. (a), (b) Log-normal distribution; (c), (d) normal distribution.

that the discriminator easily overpowers the generator. To circumvent these problems, we propose a hybrid training strategy, where the generator is optimized under a gradient-free evolutionary algorithm, while the classical discriminator uses a gradient descent optimizer. Additionally, the Wasserstein distance [35,36] is used to train the GAN, which changes the dynamic between the generator and the discriminator. In this new GAN scheme, the discriminator acts as a critic, instead of classifying. It aims to give a high score to real instances over fake ones, effectively alleviating the problem of unstable GAN training.

4. RESULTS AND DISCUSSION

A. GAN Results

Our chip can accommodate the entire option pricing process of distribution loading, payoff calculation, and amplitude estimation for three option assets. However, training three bins is too trivial to demonstrate the ability to implement GANs on a photonic chip. Here, to demonstrate the GAN, we employ a chip that supports up to eight bins to demonstrate the generation of probability distribution. Figure 5(a) shows the probability distribution of the generator output compared to the real log-normal distribution. Figure 5(b) shows the convergence of the ℓ_2 norm between the fake and real samples, of 100 training iterations. For generator output g and real distribution x , the ℓ_2 norm is defined as

$$\ell_2 = \sqrt{\sum_{i=1}^m (x_i - g_i)^2}. \quad (5)$$

The results for a target normal distribution are shown in Figs. 5(c) and 5(d). For both examples of log-normal and normal distributions, the final ℓ_2 norm between the generator output and the real distribution stabilizes at -18 dB.

By training this generative model directly on the photonic chip, we bypass the need to solve the BSM equations while capturing the nuances that the simplified method overlooks.

Concurrently, it incorporates environmental elements that are hard to model, such as cross talk and chip imperfections into the GAN training. Another feature of using GANs for the amplitude distribution step is that we can tailor the variational ansatz to construct short-depth circuits for a given degree of accuracy, even in a more general case with multiple photons.

B. Unary Option Pricing Results

As a proof of principle, the fabricated photonic chip supports an option pricing problem with three asset values, whose schematic diagram is shown in Fig. 6(a). The chip has six waveguide inputs: each pair represents one element of the unary basis and the ancilla qubit state. The chip is divided into distribution loading, payoff calculation, and then m runs of amplitude amplification. To stay within the depth constraints of this proof of concept, the circuit's unitary matrix is multiplied and uploaded into the photonic chip at a constant depth. The single-photon measurement is performed at the waveguide modes that represent ancilla state $|1\rangle$ for asset prices larger than the strike value. The comparison between the theoretical payoff and estimation achieved experimentally is shown in Fig. 6(b), with increasing iterations of amplitude estimation. The performance of the amplitude estimation is shown in Figs. 6(c) and 6(d). In Fig. 6(c), the dotted line represents the theoretical payoff

expectation, the solid line with data points represents the experimental results, and the shaded area represents the standard deviation (STD) of 50 measurements performed in each step of amplitude estimation. The progression of m from 0 to 50 ($m = 0$ being a classical sampling of payoff calculation) demonstrates the convergence of the STD. Similarly, in Fig. 6(d), we visualize the convergence of the payoff error with more amplitude estimation runs. The amplitude estimation improves the accuracy of the expected payoff for a certain number of circuit runs.

The structure of the unary algorithm allows a simple but efficient design of photonic chips, especially when loading probability distributions into quantum registers since only local interactions between neighboring waveguides are required. This, however, is inaccessible for the binary alternative, where high connectivity is required to offset the exponential Hilbert space available; see the unary and binary comparison in Appendix D. The optical circuit that implements the unary approach requires a linear number of waveguides scaling with the required precision, which coincides with the remarkable scalability of photonic chips. Instead, avoiding the use of controlled operations via ancilla waveguides bypasses one of the main bottlenecks of photonic chips in quantum computing: the obstacle of realizing photon interactions.

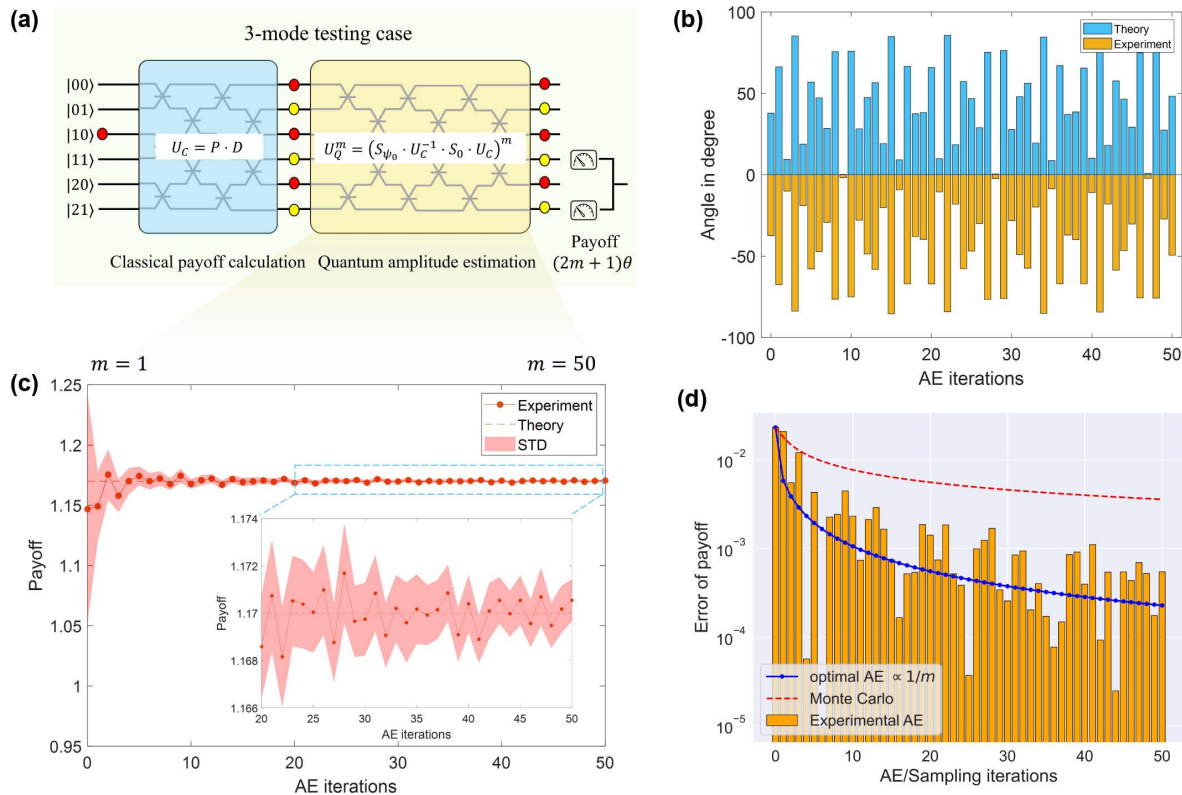


Fig. 6. Experimental results of option pricing with three asset values. (a) Illustration of the optical chip with payoff calculation and amplitude estimation module. Operator Q is repeated up to m ($m \leq 50$) times. The payoff is measured on waveguides that encode the ancilla in state $|1\rangle$ when the asset price is larger than the pre-defined strike value. (b) Comparison between theoretical expectations and experimental results of the payoff, represented in angles. The raw angles $(2m+1)\theta$ are shifted back to the original angles θ , and the differences from theoretical expectations are recorded as errors. (c) Standard deviation (STD) of the expected payoff with increasing iterations of the amplitude estimation module. The STD converges from the initial ~ 0.2 to less than 0.004. Iterations from 20 to 50 are zoomed in. (d) Error in payoff estimation between theoretical and experimental results, with increasing iterations of amplitude estimation. It shows a speedup in convergence compared to the Monte Carlo method.

Speedup is achieved in our work much akin to proposals for quantum search without entanglement [37,38], whereby polynomial speedup of unstructured search is achieved with a single photon at the cost of exponential resources. In particular, isomorphisms exist between a system of n qubits and a qudit residing in a 2^n Hilbert space (systems presented in this thought experiment and our implementation) [39]; thereby the unary implementation on a photonic chip displays entanglement in the path encoding of the single photon. Coherent light can achieve a similar effect with a high sampling rate, which is advantageous in near-term use case scenarios, with some trade-offs for the random behavior of single photons in the generator part of the GAN.

The presented avenue to achieve speedup in option pricing is scalable in the photonic chip. It transforms the unary algorithm's need for increasing qubits into a need for waveguide paths, which are highly scalable in photonic chips. For further scalability, in the presented experiment, the photon detectors placed in ancilla waveguides could be combined into a single one, as only the counts of photons in any ancilla qubit are needed, hence significantly reducing the resources needed to scale this approach to meaningful problems. The energy efficiency of photonic chips also promises a relevant advantage beyond a complexity separation between quantum and classical algorithms. Given an energy budget instead of a shot budget, the photonic implementation of the unary approach to option pricing can yield a significant advantage in the number of operations performed.

5. CONCLUSION

This work is the first demonstration of photonic chips for financial applications. As a proof of concept, we implement the unary option pricing algorithm in a photonic chip for European options, which includes the generation of amplitude distribution of the asset value, evaluation of expected return, and amplitude estimation. We prove the high accuracy in calculating the payoff function, as well as the effectiveness of amplitude estimation in reducing the number of evaluations to reach the same degree of accuracy when compared to classical sampling. The unary representation remarkably simplifies the structure and reduces the depth of quantum circuits in the linear optical circuit implementation. Such photonic devices could eventually be an eco-friendly alternative to electronic circuits. Furthermore, we demonstrate an on-chip training of a GAN that successfully captures important market dynamics in real-life scenarios, bypassing the simplified assumptions in the BSM model that limit its accuracy, as well as the computational burden in solving differential equations. The photonic chip could be potentially employed for other options pricing, paving the way for developing dedicated processors in finance applications.

APPENDIX A: EUROPEAN OPTION PRICING MODEL

The Black–Scholes model is a typical economic model used to calculate the evolution of asset prices in financial markets, known as the European option pricing problem. In this model, the evolution of option price S_T at time T is decided by two

market properties: interest rate r and volatility σ , which are expressed by the stochastic differential equation

$$dS_T = S_T r dT + S_T \sigma dW_T, \quad (\text{A1})$$

where W_T describes a Brownian process, which is a continuous stochastic evolution starting at $W_0 = 0$ and consists of independent Gaussian increments. Specifically, let $\mathcal{N}(\mu, \sigma_s)$ be a normal distribution with mean μ and STD σ_s ; then the increment of two steps of the Brownian processes is $W_T - W_S \sim \mathcal{N}(0, T - S)$, for $T > S$. The stochastic differential equation can be approximately resolved to first order, and the solution is

$$S_T = S_0 e^{(r - \frac{\sigma^2}{2})T} e^{\sigma W_T} \sim e^{\mathcal{N}((r - \frac{\sigma^2}{2})T, \sigma\sqrt{T})}, \quad (\text{A2})$$

which is a log-normal distribution. The process of solving the stochastic differential equation is valid for the simplified European option model, while for more practical cases, an analytical solution does not exist, and even numerical simulation is costly. To get the expected return, a payoff calculation block is integrated over the resulting probability distribution. The payoff function is given by

$$f(S_T, K) = \max(0, S_T - K), \quad (\text{A3})$$

producing the expected payoff

$$C(S_T, K) = \int_K^\infty (S_T - K) dS_T, \quad (\text{A4})$$

where K is the strike.

APPENDIX B: EXPERIMENTAL SETUP AND SINGLE-PHOTON GENERATION

The entire packaged chip is shown in Fig. 7. Each phase shifter is independently controlled by an electronic current driver with 1-kHz frequency and 12-bit resolution. Output photons are filtered via wavelength division multiplexing (WDM) to remove the residual photons, and then detected by superconducting nanowire single-photon detectors (SNSPDs) (PhotonSpot, 100 Hz dark counts, 85% efficiency). Polarization controllers are placed before the SNSPDs as the detectors are polarization sensitive. A time tagger (Swabian Instrument) is used to count the single-photon events, which can support more than 40 million events per second. A temperature controller is used to

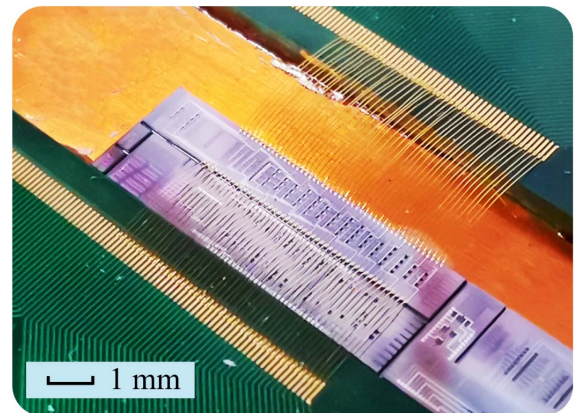


Fig. 7. Fabricated quantum photonic chip.

stabilize the chip temperature and reduce thermal fluctuations caused by possible cross talk.

A degenerated photon pair is used in our experiment. The pump laser is generated from the Ultrafast Optical Clocks device (PriTel) with a repetition rate of 500 MHz, central wavelength of 1550.116 nm, and bandwidth of 1.9 nm. A dual pump scheme is employed to generate pairs of identical photons on chip with a degenerated spontaneous four-wave mixing (SFWM) process. On the chip, the desired state $|\psi\rangle = |11\rangle$ is generated out of the two-photon N00N state $|\psi\rangle = \frac{1}{\sqrt{2}}(|20\rangle + |02\rangle)$, by configuring the phase value $\theta = \pi/2$ when interfering the two photons.

APPENDIX C: THEORY OF UNARY OPTION PRICING

By solving the aforementioned BSM model, the probability density function of the option price can be described by a log-normal distribution. We map this continuous price distribution into n discrete values, which are the amplitudes of n orthogonal quantum state basis, by using a probability loading operator D acting on an initial state $|\psi_{\text{ini}}\rangle$ as

$$D|\psi_{\text{ini}}\rangle = \sum_{i=0}^{n-1} \sqrt{p_i} |\psi_i\rangle_n, \quad (\text{C1})$$

where each state $|\psi_i\rangle$ represents a discrete option price value S_i , and p_i is the corresponding probability. These quantum state bases are orthogonal so that $\langle\psi_i|\psi_j\rangle = \delta_{ij}$. The payoff is obtained by accumulating the asset value under its corresponding probability. The payoff of the European option in this discrete scenario can be simplified as

$$C(S_T, K) = \sum_0^{n-1} p_i \cdot f(S_i, K) = \sum_{S_i > K}^{n-1} p_i \cdot (S_i - K), \quad (\text{C2})$$

where K is the strike price. The rotation angles after being normalized by the maximum asset price S_{max} are given by

$$\theta_i = \max\left(0, \arcsin\left(\sqrt{\frac{S_i - K}{S_{\text{max}} - K}}\right)\right). \quad (\text{C3})$$

This payoff calculation can be mapped to the quantum model by introducing an ancilla qubit into the original quantum state followed by a controlled rotation gate CR defined as

$$\text{CR} = \sum_{i=0}^{n-1} |\psi_i\rangle\langle\psi_i| \otimes R_y(2\theta_i). \quad (\text{C4})$$

Then, the expected payoff of the option price is related to the amplitude of the ancilla qubit in the form of

$$\begin{aligned} |\psi\rangle &= \text{CR} \cdot \sum_{i=0}^{n-1} \sqrt{p_i} |\psi_i\rangle \otimes |0\rangle \\ &= \sum_{i=0}^{n-1} \sqrt{p_i} \cos \theta_i |\psi_i\rangle |0\rangle + \sqrt{p_i} \sin \theta_i |\psi_i\rangle |1\rangle. \end{aligned} \quad (\text{C5})$$

By measuring the ancilla qubit under basis $|1\rangle$, we can achieve the result as

$$|\langle 1|\psi\rangle|^2 = \sum_{i=0}^{n-1} p_i \cdot \sin^2 \theta_i = \frac{C(S_T, K)}{S_{\text{max}} - K}. \quad (\text{C6})$$

Thus, the payoff of the option price can be directly read out from the measurement results of the ancilla qubit under basis $|1\rangle$. Then, we explain how the amplitude estimation works. The payoff calculation [Eq. (C5)] can be simplified as

$$\text{CR} \cdot D \cdot |\psi_{\text{ini}}\rangle |0\rangle = \cos \alpha |\psi_a\rangle |0\rangle + \sin \alpha |\psi_b\rangle |1\rangle, \quad (\text{C7})$$

where α is the normalized parameter, and $|\psi_a\rangle$ and $|\psi_b\rangle$ are the normalized states

$$|\psi_a\rangle = \sum_{i=1}^{n-1} \sqrt{p_i} \cos \theta_i |\psi_i\rangle, \quad |\psi_b\rangle = \sum_{i=1}^{n-1} \sqrt{p_i} \sin \theta_i |\psi_i\rangle. \quad (\text{C8})$$

The ancilla qubit is functioning as an indicator to identify the useful state. The amplitude amplification step begins by applying an oracle operator S_ψ on state ψ with the form

$$S_\psi = I - 2 \sum_{i=0}^{n-1} |\psi_i\rangle\langle\psi_i| \otimes |0\rangle\langle 0| \quad (\text{C9})$$

to produce a sign change on the ancilla qubit state $|0\rangle$ that we want to perform the amplitude estimation. Then, we add an inversion operation of the previous payoff calculation CR and distribution loading operator D followed by another sign flip operation S_0 on the initial state as

$$S_0 = I - 2|\psi_{\text{ini}}\rangle\langle\psi_{\text{ini}}| \otimes |0\rangle\langle 0|, \quad (\text{C10})$$

and the last step is to apply D and CR again so that the amplitude estimation operator Q can be written as

$$Q = \text{CR} \cdot D \cdot S_0 \cdot D^\dagger \cdot \text{CR}^\dagger \cdot S_\psi. \quad (\text{C11})$$

By repeating the Q operator m times, the full amplitude estimation can be represented as

$$\begin{aligned} Q^m \cdot \text{CR} \cdot D \cdot |\psi_{\text{ini}}\rangle |0\rangle \\ = \cos(2m+1)\alpha |\psi_a\rangle |0\rangle + \sin(2m+1)\alpha |\psi_b\rangle |1\rangle. \end{aligned} \quad (\text{C12})$$

Therefore, the measurement of the ancilla qubit under basis $|1\rangle$ after repeated amplitude estimation would yield the results of $\sin^2(2m+1)\alpha$ for us to infer the payoff of the option price

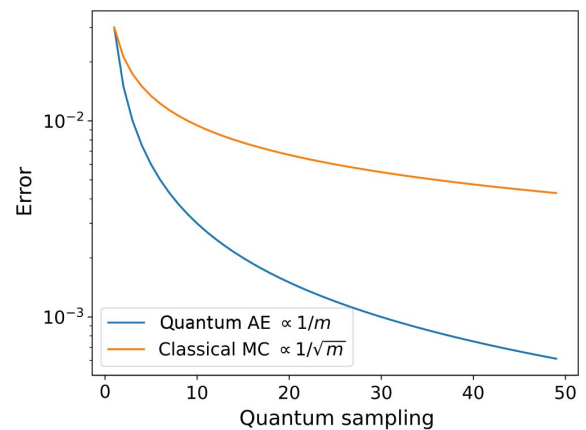


Fig. 8. Simulation of the scaling of quantum AE and classical MC.

Table 1. Unary and Binary Comparison

Aspect	Unary Approach	Binary Approach
Representation	Intuitive: single symbol repeated multiple times	Compact: base-2 system with 0 and 1 symbols
Chip architecture	Simple: first-nearest neighbor connectivity	Complex: full connectivity
Amplitude estimation	Without phase estimation: feasible in linear optical circuits	Phase estimation required: not feasible in linear optical circuits
Gate count	Linear (advantageous for near-term devices with <100 qubits)	Logarithmic, requiring Toffoli gate
Distribution loading error due to single-qubit error	KL divergence of 10^{-3} , one order of magnitude lower	KL divergence of 10^{-2}
Payoff deviation due to single-qubit error	~25%, 10% more robust	~35%

with improved accuracy. The amplitude estimation scheme we use here is an iterative approach [9]. This procedure is based on the theory of confidence intervals for binomial distributions [40] and uses samples of increasing amplitude amplification [4] steps to better estimate the value of the target amplitude. The quantum amplitude estimation algorithm achieves quadratic speedups when compared to classical Monte Carlo methods of option pricing:

$$\mathcal{O}\left(\frac{1}{\sqrt{m}}\right) \rightarrow \mathcal{O}\left(\frac{1}{m}\right), \quad (\text{C13})$$

where m is the number of quantum samples used. The comparison between quantum and classical scaling factors is depicted in Fig. 8, exhibiting a trend that aligns well with our experimental results shown in Fig. 6(d).

APPENDIX D: UNARY AND BINARY COMPARISON

The utilization of a unary approach [9], instead of the commonly adopted binary approach [6,11], distinguishes this work from other quantum approaches. The key advantage of the unary method is its ability to implement all the necessary quantum operations within the option pricing algorithm using a linear optical circuit. In contrast, the binary approach relies on two-qubit controlled operations, which cannot be deterministically achieved in a photonic chip. The comparison between unary and binary approaches is summarized in Table 1. The unary method has a simple chip architecture, no need for phase estimation, scalable gate count, accurate distributed loading, and robustness in payoff computation.

Funding. Hong Kong Polytechnic University (P0046236); National Research Foundation Singapore (NRF2022-QEP2-02-P16); Ministry of Education - Singapore (MOE2017-T3-1-001).

Author Contributions. H.Z., L.X.W., J.I.L., and A.Q.L. jointly conceived the idea. H.Z., L.X.W., and H.C. designed the chip and built the experimental setup. F.G., G.Q.L., and X.S.L. fabricated the silicon photonic chip. H.Z., L.X.W., and Y.C.Z. performed the experiments. W.K.M., S.R.C., L.C.K., and J.I.L. assisted with the theory. All authors contributed to the discussion of experimental results. L.C.K., J.I.L., and A.Q.L. co-supervised and coordinated all the work. H.Z., S.R.C.,

J.I.L., and A.Q.L. wrote the manuscript with contributions from all co-authors.

Disclosures. The authors declare no conflicts of interest.

Data Availability. Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

REFERENCES

1. F. Black and M. Scholes, "The pricing of options and corporate liabilities," in *Foundations of CCA and Equity Valuation*, World Scientific Reference on Contingent Claims Analysis in Corporate Finance (World Scientific, 2019), pp. 3–21.
2. R. Orus, S. Mugel, and E. Lizaso, "Quantum computing for finance: overview and prospects," *Rev. Phys.* **4**, 100028 (2019).
3. D. Herman, C. Googin, X. Liu, A. Galda, I. Safro, Y. Sun, M. Pistoia, and Y. Alexeev, "A survey of quantum computing for finance," *arXiv*, [arXiv:2201.02773](https://arxiv.org/abs/2201.02773) (2022).
4. G. Brassard, P. Hoyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," *arXiv*, [arXiv:quant-ph/0005055](https://arxiv.org/abs/quant-ph/0005055) (2002).
5. A. Montanaro, "Quantum speedup of Monte Carlo methods," *Proc. R. Soc. A* **471**, 20150301 (2015).
6. P. Rebentrost, B. Gupt, and T. R. Bromley, "Quantum computational finance: Monte Carlo pricing of financial derivatives," *Phys. Rev. A* **98**, 022321 (2018).
7. S. Woerner and D. J. Egger, "Quantum risk analysis," *npj Quantum Inf.* **5**, 15 (2019).
8. S. Focardi, F. J. Fabozzi, and D. Mazza, "Quantum option pricing and quantum finance," *J. Deriv.* **28**, 79–98 (2020).
9. S. Ramos-Calderer, A. Pérez-Salinas, D. Garca-Martín, C. Bravo-Prieto, J. Cortada, J. Planaguma, and J. I. Latorre, "Quantum unary approach to option pricing," *Phys. Rev. A* **103**, 032414 (2021).
10. F. Bova, A. Goldfarb, and R. G. Melko, "Commercial applications of quantum computing," *EPJ Quantum Technol.* **8**, 2 (2021).
11. N. Stamatopoulos, D. J. Egger, Y. Sun, C. Zoufal, R. Iten, N. Shen, and S. Woerner, "Option pricing using quantum computers," *Quantum* **4**, 291 (2020).
12. J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum* **2**, 79 (2018).
13. K. Bharti, A. Cervera-Lierta, T. H. Kyaw, T. Haug, S. Alperin-Lea, A. Anand, M. Degroote, H. Heimonen, J. S. Kottmann, T. Menke, W.-K. Mok, S. Sim, L.-C. Kwek, and A. Aspuru-Guzik, "Noisy intermediate-scale quantum algorithms," *Rev. Mod. Phys.* **94**, 015004 (2022).
14. H. Zhang, L. Wan, T. Haug, W.-K. Mok, S. Paesani, Y. Shi, H. Cai, L. K. Chin, M. F. Karim, L. Xiao, X. Luo, F. Gao, B. Dong, S. Assad, M. S. Kim, A. Laing, L. C. Kwek, and A. Q. Liu, "Resource-efficient high-dimensional subspace teleportation with a quantum autoencoder," *Sci. Adv.* **8**, eabn9783 (2022).
15. T. Wang, M. M. Sohoni, L. G. Wright, M. M. Stein, S.-Y. Ma, T. Onodera, M. G. Anderson, and P. L. McMahon, "Image sensing with multilayer nonlinear optical neural networks," *Nat. Photonics* **17**, 408–415 (2023).

16. T. Fu, Y. Zang, Y. Huang, Z. Du, H. Huang, C. Hu, M. Chen, S. Yang, and H. Chen, "Photonic machine learning with on-chip diffractive optics," *Nat. Commun.* **14**, 70 (2023).
17. S. Xu, J. Wang, H. Shu, Z. Zhang, S. Yi, B. Bai, X. Wang, J. Liu, and W. Zou, "Optical coherent dot-product chip for sophisticated deep learning regression," *Light Sci. Appl.* **10**, 221 (2021).
18. K. Liao, Y. Chen, Z. Yu, X. Hu, X. Wang, C. Lu, H. Lin, Q. Du, J. Hu, and Q. Gong, "All-optical computing based on convolutional neural networks," *Opto-Electron. Adv.* **4**, 200060 (2021).
19. H. Zhou, J. Dong, J. Cheng, W. Dong, C. Huang, Y. Shen, Q. Zhang, M. Gu, C. Qian, H. Chen, Z. Ruan, and X. Zhang, "Photonic matrix multiplication lights up photonic accelerator and beyond," *Light Sci. Appl.* **11**, 30 (2022).
20. J. Romero and A. Aspuru-Guzik, "Variational quantum generators: generative adversarial quantum machine learning for continuous distributions," *Adv. Quantum Technol.* **4**, 2000003 (2021).
21. H. Zhang, M. Gu, X. Jiang, J. Thompson, H. Cai, S. Paesani, R. Santagati, A. Laing, Y. Zhang, M. Yung, Y. Z. Shi, F. K. Muhammad, G. Q. Lo, X. S. Luo, B. Dong, D. L. Kwong, L. C. Kwek, and A. Q. Liu, "An optical neural chip for implementing complex-valued neural network," *Nat. Commun.* **12**, 457 (2021).
22. Q. Zhang, H. Yu, M. Barbiero, B. Wang, and M. Gu, "Artificial neural networks enabled by nanophotonics," *Light Sci. Appl.* **8**, 42 (2019).
23. J. Spall, X. Guo, T. D. Barrett, and A. I. Lvovsky, "Fully reconfigurable coherent optical vector-matrix multiplication," *Opt. Lett.* **45**, 5752–5755 (2020).
24. S. Yan and J. Dong, "Metasurface for highly-efficient on-chip classical and quantum all-optical modulation," *Light Sci. Appl.* **11**, 238 (2022).
25. C. Zoufal, A. Lucchi, and S. Woerner, "Quantum generative adversarial networks for learning and loading random distributions," *npj Quantum Inf.* **5**, 103 (2019).
26. L. Hu, S.-H. Wu, W. Cai, Y. Ma, X. Mu, Y. Xu, H. Wang, Y. Song, D.-L. Deng, C.-L. Zou, and L. Sun, "Quantum generative adversarial learning in a superconducting quantum circuit," *Sci. Adv.* **5**, eaav2761 (2019).
27. H.-L. Huang, Y. Du, M. Gong, Y. Zhao, Y. Wu, C. Wang, S. Li, F. Liang, J. Lin, Y. Xu, R. Yang, T. Liu, M.-H. Hsieh, H. Deng, H. Rong, C.-Z. Peng, C.-Y. Lu, Y.-A. Chen, D. Tao, X. Zhu, and J.-W. Pan, "Experimental quantum generative adversarial networks for image generation," *Phys. Rev. Appl.* **16**, 024051 (2021).
28. K. Huang, Z.-A. Wang, C. Song, K. Xu, H. Li, Z. Wang, Q. Guo, Z. Song, Z.-B. Liu, D. Zheng, H. Wang, J.-G. Tian, and H. Fan, "Quantum generative adversarial networks with multiple superconducting qubits," *npj Quantum Inf.* **7**, 165 (2021).
29. C. Wu, X. Yang, H. Yu, R. Peng, I. Takeuchi, Y. Chen, and M. Li, "Harnessing optoelectronic noises in a photonic generative network," *Sci. Adv.* **8**, eabm2956 (2022).
30. S. Lloyd and C. Weedbrook, "Quantum generative adversarial learning," *Phys. Rev. Lett.* **121**, 040502 (2018).
31. Y. Suzuki, S. Uno, R. Raymond, T. Tanaka, T. Onodera, and N. Yamamoto, "Amplitude estimation without phase estimation," *Quantum Inf. Process.* **19**, 75 (2020).
32. S. Aaronson and P. Rall, "Quantum approximate counting, simplified," in *Symposium on Simplicity in Algorithms* (2020), pp. 24–32.
33. D. Grinko, J. Gacon, C. Zoufal, and S. Woerner, "Iterative quantum amplitude estimation," *npj Quantum Inf.* **7**, 52 (2021).
34. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *Advances in Neural Information Processing Systems* (2014), Vol. **27**, pp. 2672–2680.
35. M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning* (2017), pp. 214–223.
36. S. Chakrabarti, H. Yiming, T. Li, S. Feizi, and X. Wu, "Quantum Wasserstein generative adversarial networks," *arXiv*, arXiv:1911.00111 (2019).
37. S. Lloyd, "Quantum search without entanglement," *Phys. Rev. A* **61**, 010301 (1999).
38. D. A. Meyer, "Sophisticated quantum search without entanglement," *Phys. Rev. Lett.* **85**, 2014–2017 (2000).
39. A. Ekert and R. Jozsa, "Quantum algorithms: entanglement-enhanced information processing," *Philos. Trans. R. Soc.* **356**, 1769–1782 (1998).
40. S. Wallis, "Binomial confidence intervals and contingency tests: mathematical fundamentals and the evaluation of alternative methods," *J. Quant. Linguist.* **20**, 178–208 (2013).