

POSITION PAPER

Quantum computing for data-centric engineering and science

Steven Herbert^{1,2} 

¹Quantinuum (Cambridge Quantum), Cambridge CB2 1NL, United Kingdom

²Department of Computer Science and Technology, University of Cambridge, Cambridge CB3 0FD, United Kingdom

E-mail: Steven.Herbert@Quantinuum.com

Received: 07 July 2022; **Revised:** 28 September 2022; **Accepted:** 14 October 2022

Keywords: Data science; estimation; Monte Carlo integration; quantum algorithms; quantum computing

Abstract

In this perspective, I give my answer to the question of how quantum computing will impact on data-intensive applications in engineering and science. I focus on quantum Monte Carlo integration as a likely source of (relatively) near-term quantum advantage, but also discuss some other ideas that have garnered widespread interest.

Impact Statement

For the past several decades, it has been known that quantum mechanics gives rise to a computational paradigm that offers spectacular speed-ups for certain tasks relative to the best possible conventional, “classical” algorithms. In recent years, quantum computation has become a physical reality; and simultaneously there has been an explosion in the volume of data that must be processed by computers. Thus the tantalizing possibility of quantum computation aiding the processing of large classical datasets has gained widespread interest. This article takes a broad look at the prominent proposals for how quantum computation may impact on data-centric applications. The pros and cons of “quantum machine learning” algorithms are discussed, and quantum Monte Carlo integration is identified as a potential source of (relatively) near-term quantum advantage. Finally, some speculative predictions are given for when such quantum advantage may materialize.

1. Quantum Computing: A Very Brief Introduction

The conception of quantum computing is usually attributed to Richard Feynman, who in 1981 speculated that simulating the behavior of a quantum mechanical system would require a computer that was itself somehow quantum mechanical in nature (Feynman, 1982; Preskill, 2021). Manin (1980) and Benioff (1980) also espoused similar ideas at around the same time. It was David Deutsch who in 1985 then laid the groundwork for quantum computing as we now know it, by formalizing a quantum mechanical model of computation, and posing well-defined mathematical problems where quantum computing offers a clear computational advantage (Deutsch, 1985). This in turn spawned a great profusion of activity in the embryonic field of quantum computing in the late 1980s and early 1990s, leading to what remains to this day two of the crowning achievements of the field: in 1994, Peter Shor proposed a quantum algorithm for factoring in polynomial time (Shor, 1994) and in 1996, Lov Grover proposed an algorithm to search an unstructured database in time proportional to the square root of the database size (Grover, 1996).

Unstructured search (in this context) is the problem where we have some $N = 2^n$ elements, indexed $\{0, 1\}^n$, to search through, and a “function” f , such that for exactly one $x \in \{0, 1\}^n$, $f(x) = 1$, and $f(x) = 0$ otherwise. “Unstructured” means there is no algorithmic short-cut— f is a function in the technical sense only and does not imply it can be represented as some simple algebraic expression—and hence classically the best (only) strategy is exhaustive search, which requires $f(x)$ to be evaluated for all N elements at worst, and $N/2$ elements on average. Quantumly, we can prepare a superposition of all possible n -bitstrings, and hence “query” f for all possible x in a single step, *however, this does not imply that quantum unstructured search completes in $\mathcal{O}(1)$ operations*. In fact, as the answer is encoded in a quantum state it turns out that it takes at least $\mathcal{O}(\sqrt{N})$ operations to extract—a lower bound that Grover’s search algorithm achieves. This improvement from $\mathcal{O}(N)$ classical operations to $\mathcal{O}(\sqrt{N})$ quantum operations is commonly referred to as a “quadratic advantage.”

While the quadratic advantage is extremely valuable, the fact that quantum computing enables the simultaneous querying of f for an exponential number of x (that is, we say the problem size is “ n ” and we query f for all $N = 2^n$ possible n -bitstrings in superposition), dangles the tantalizing possibility of *exponential* computational advantages. To see such advantages, we must move on from unstructured search to problems with some specific structure that can be attacked by quantum, but not classical algorithms. The manner in which this structure is attackable by the quantum algorithm can be a little hard to grasp, but essentially amounts to the fact that the answer we are searching for is in some sense determined by all 2^n queries. But in such a way that a quantum mechanical “interference” step (for which there is no analog in classical computation) can efficiently extract the solution. This is indeed the case for Shor’s factoring algorithm, where the *Quantum Fourier Transform* (QFT) performs this interference step (in fact many of the most prominent proposals for super-polynomial quantum advantage use the QFT). Classically the best factoring algorithm is the number field sieve (Lenstra et al., 1993), which has complexity $\exp\left(\Theta\left(n^{1/3} \log^{2/3} n\right)\right)$, whereas Shor’s algorithm requires only $\mathcal{O}(n^2 \log n \log \log n)$ operations, where the problem size, n , is the number of bits required to express the number being factored.

For the past 25 years, Shor’s and Grover’s algorithms have been the mighty pillars upon which many other proposals for quantum algorithms have been built, and the computational complexity thereof continues to provide some insight into the sorts of advantage we should expect from quantum algorithms: if the algorithm is tackling a task with little structure, then we expect a quadratic (or other polynomials) advantage; whereas if there is a structure that can be exploited by a quantum interference step (such as the QFT) then we can get a super-polynomial speed-up. (Scott Aaronson recently posted a very nice and concise article about the role of structure in quantum speed-ups [Aaronson, 2022].)

Furthermore, an important point to note is that all of the “canonical” quantum algorithms presume an abstract model of quantum computation, which is innately noiseless (quantum noise occurs when the environment randomly perturbs the quantum state such that it departs from that predicted by the abstract model of quantum computation). It was therefore a substantial and highly important breakthrough when it was shown that real, noisy, and quantum hardware can efficiently simulate the noiseless model of quantum computation in principle owing to the celebrated *threshold theorem* (Shor, 1996; Knill et al., 1998; Kitaev, 2003; Aharonov and Ben-Or, 2008). However, in practice, this still requires a quantum error-correction overhead that takes the noiseless model out of reach of near-term quantum hardware. In the past several years, significant attention has been given to the question of whether useful quantum advantage can be obtained by computing with noisy qubits, that is, without quantum error correction. For this setting, John Preskill coined the term “NISQ” (*noisy intermediate-scale quantum [computer]*) (Preskill, 2018), and it has become commonplace to speak of the “NISQ-era” (computing with noisy qubits) which will eventually give way to the “full-scale era” (when quantum error correction will mean that we can essentially treat the qubits as noiseless), although I shall later argue that this is something of a false dichotomy.

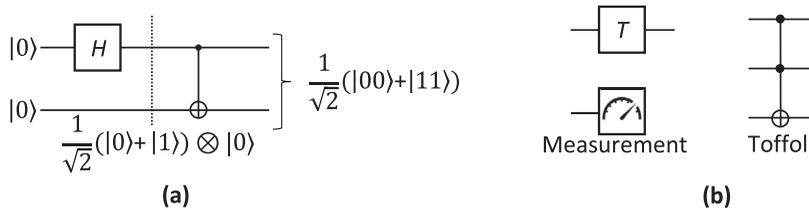


Figure 1. The quantum algorithms discussed in this work are generally formulated in terms of the quantum circuit (or quantum gate) model. Each wire corresponds to a qubit and the gates are unitary matrices, the qubits are initialized in a computational basis state $|0\rangle = [1, 0]^T$ or $|1\rangle = [0, 1]^T$, which are analogous to the 0 and 1 states of classical bits. Qubits differ as they may be put in a superposition of the computational basis states, for example, the Hadamard (H) gate in the circuit in panel (a) puts the first qubit in the superposition $(1/\sqrt{2})(|0\rangle + |1\rangle)$ (quantum states are such that the squared moduli of the coefficients of the computational basis states sum to one). The qubits in the circuit are composed using the tensor product, and hence the full state after the Hadamard gate is $(1/\sqrt{2})(|0\rangle + |1\rangle) \otimes |0\rangle$ (which can equivalently be expressed $(1/\sqrt{2})(|00\rangle + |11\rangle)$). The next gate is the two-qubit CNOT, which transforms the state into the “Bell state” $|\Phi^+\rangle = (1/\sqrt{2})(|00\rangle + |11\rangle)$ —a state which cannot be expressed as a tensor product of two single-qubit states, and is thus referred to as an entangled state. As well as the Hadamard and CNOT, some further important gates are shown in panel (b): the T gate, Toffoli, and measurement. Together H, T, and CNOT form a universal gate set—any quantum circuit can be expressed (to arbitrary precision) as a circuit containing just these; however, the Toffoli gate is also a useful primitive as it implements the classically universal NAND gate as a (three-qubit) unitary operation. Measurements are needed to extract information from the quantum state, and a (single-qubit computational basis) measurement yields a single classical bit. The measurement outcome is random, and each computational basis state is measured with probability equal to its coefficient’s modulus squared, for example, if the state $(1/\sqrt{2})(|0\rangle + |1\rangle)$ is measured, then the classical bits 0 and 1 are each measured with 50% probability, this is known as the “Born rule.”

In general, both “NISQ” and “full-scale” quantum algorithms are usually formulated using the quantum circuit model,¹ which is briefly introduced in Figure 1. It is common to use the circuit depth (the number of layers of operations) as a proxy for computational complexity and in the case of NISQ algorithms, the circuit depth dictates the number of operations that must be performed with the state remaining coherent, that is, before the noise becomes too great and the information contained within the state is lost. So it follows that, when designing quantum algorithms with resource constraints in mind, it is important to keep the circuit depth as low as possible—in order to achieve the computation within the physical qubit coherence time (NISQ) or with as little error correction as possible (full-scale).

Figure 2 summarizes some of the most important breakthroughs in quantum computing, and for further information, the reader is directed to Nielsen and Chuang (2010), which remains the authoritative textbook on the subject. The *quantum algorithm zoo* (Jordan, 2011) also provides a catalog of many suggested quantum algorithms—although the total number of algorithms can be somewhat misleading: many of the listed algorithms amount to different instances and applications of the same essential quantum speed-up. Additionally, *quantumalgorithms.org* brings together many important quantum algorithms for data analysis (Luongo, 2022).

2. How Will Quantum Computing Help Me With All My Data?

We can see, even from the concise introduction above, that quantum computation, as it is conventionally broached, is very much bound up with the theory of computational complexity. However, when I speak to

¹ Alternatives to the quantum circuit model include *linear optical quantum computing* (Knill et al., 2001), *adiabatic quantum computation* (Farhi et al., 2000), and *measurement-based quantum computation* (Briegleb et al., 2009).

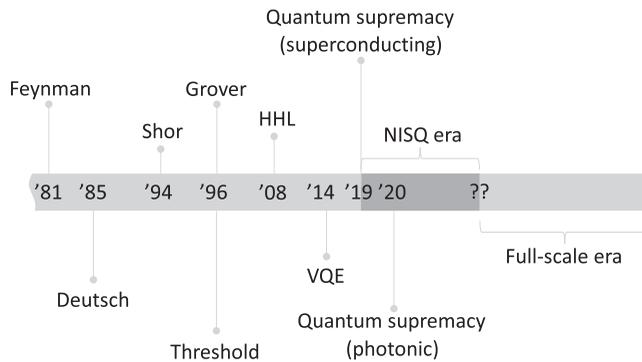


Figure 2. A timeline of some of the most important results in quantum computing. The variational quantum eigensolver (VQE) (Peruzzo et al., 2014) is widely acknowledged as one of the most promising NISQ algorithms; and Google’s demonstration of (superconducting) quantum supremacy may be taken as the start of the NISQ-era (in the sense that this was the first demonstration of a quantum algorithm significantly outperforming its classical counterpart). Quantum supremacy on a photonic quantum computer was first claimed by Jian-Wei Pan’s group (Zhong et al., 2020), and later by Xanadu (Madsen et al., 2022).

computational researchers from outside of quantum computing, invariably what they say to me is not, for example, “I am struggling with this *computationally* hard problem,” but rather they ask “how can quantum computing help me with all of my *data*?” For we are living through an era of unprecedented data generation, and this poses problems at every stage of the computational workflow. The most urgent question that researchers are asking of nascent computational technologies is how they can remedy these emerging and growing problems.

This is the challenge taken up by Aram Harrow in *small quantum computers and large classical datasets* (Harrow, 2020), which proposes using the quantum computer to do computationally intensive model searches when substantial data reduction is possible on the “large classical dataset.” However, the question of what quantum computing can do to deal with “big data” more generally is tricky: loading data onto the quantum computer is well-known to be a hard problem, not only with the small-scale quantum hardware that is available at present, but a fundamental problem in principle, and one which if we are not careful could easily nullify the quantum advantage. This has in turn brought about something of a divide between “pessimists” who believe that the data-loading problem is fundamentally an insurmountable obstacle, and “optimists” who focus on the unquestionable computational benefits once the data *is* loaded, and assume that some solution will emerge to the data-loading problem itself.

The purpose of this article is to provide one answer to the motivating question of what quantum computing can do to help with the massive proliferation of data, that is neither unduly pessimistic or optimistic, but rather is *realistic*—and illuminates a plausible path ahead for the eventual integration of quantum computing into data-centric applications.

3. Quantum Computing and Machine Learning: A Match Made in Heaven?

In recent years, there has been an explosion of papers on “Quantum Machine Learning” (QML), and a cynic would say that this amounts to little more than a case of buzzword fusion to unlock funding sources and generate hype. But I am not a cynic—for one thing, some of the most respected researchers in quantum computing are working on QML—and there are (at least) two very good reasons to believe that quantum computing may ultimately offer significant computational advantages for machine learning tasks. These two reasons in turn inform two complementary approaches to QML.

One approach stems from the functional similarity between artificial neural networks (ANNs) and parameterized quantum circuits (PQCs) (Benedetti et al., 2019b), as shown in Figure 3. In particular, by

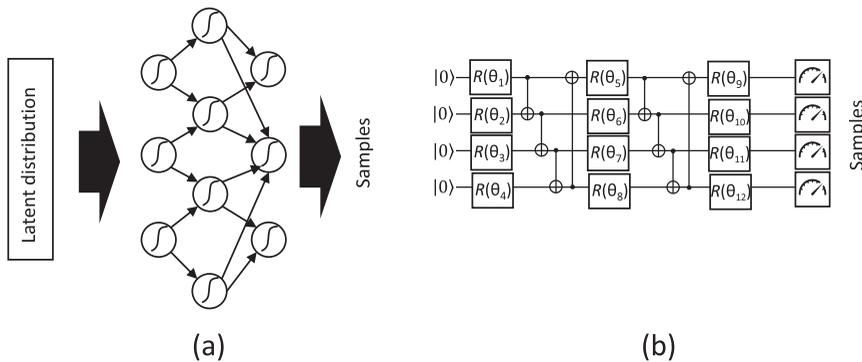


Figure 3. Classical and quantum generative models: (a) A classical generative model will typically use an artificial neural network to map samples from some standard latent space distribution (such as a uniform or Gaussian distribution) to samples from the target distribution; (b) using a parameterized quantum circuit as a quantum generative model is a similar concept, except that the measurement corresponds to a random collapse of the quantum state, and hence suffices to generate samples from a probability distribution even if the PQC operates on a fixed initial state such as $|0\rangle$.

virtue of the fact that we must always measure the quantum state to extract some information (and noting that measurement triggers a probabilistic “collapse” of the quantum superposition into one of some ensemble of possible states), a PQC is innately something we sample from, and is therefore, in a sense, analogous to an ANN trained as a generative model (Lloyd and Weedbrook, 2018; Benedetti et al., 2019a; Zoufal et al., 2019; Chang et al., 2021; Zoufal, 2021). (There are myriad proposals to use PQCs in place of ANNs for other learning tasks [Dong et al., 2008; Havlíček et al., 2019; Jerbi et al., 2021], but considering generative models suffices to illustrate my point here.) The original hope of quantum advantage in generative modeling stemmed from the fact that there is strong theoretical evidence for the existence of probability distributions from which samples can be prepared quantumly in polynomial time, but would require exponential time classically, for example, probability distributions sampled by *IQP circuits* (Brenner et al., 2010). Indeed, most proposals and demonstrations of *Quantum Supremacy* are sampling experiments (Harrow and Montanaro, 2017; Arute et al., 2019). The ramifications for generative modeling are that, should the target distribution be some such “classically intractable” distribution to sample, then we would need an infeasibly large ANN to train a generative model thereof, but only a relatively small PQC.

However, in practice, this is perhaps a slightly over-simplistic outlook: because the datasets of interest in engineering and other typical applications will themselves have been generated by some “classical” process, and so are unlikely to have probability distributions that we expect to be hard to classically sample from. (For instance, we do not, in general, expect classical random processes such as financial time series to exhibit the sort of correlations seen in the measurement statistics of highly entangled quantum circuits.) To put it another way, even though PQCs have greater expressivity, it is not clear that this can be harnessed for any useful application. Compounding this apparently fundamental obstacle is the fact that PQCs are incredibly hard to train (Bittel and Kliesch, 2021), and the cost function landscape is overwhelmingly dominated by large, flat regions termed *barren plateaus* (McClean et al., 2018; Arrasmith et al., 2021; Cerezo et al., 2021; Thanasiip et al., 2021; Wang et al., 2021). Nevertheless, in spite of these apparent problems, there is some evidence that QML based on PQC training will yield useful quantum advantage in classical data science applications (Coyle et al., 2020; Hubregtsen et al., 2020; Shalaginov and Dubrovsky, 2022). Indeed, in spite of the question marks hanging over PQCs as ML models in terms of their trainability and expressivity, there remains hope that such models may still have greater power in terms of generalization capability (Schreiber et al., 2022).

So we turn to the second approach to QML, which builds on the ability, in principle, of quantum computers to perform certain linear algebra computations (exponentially) faster than the best classical counterpart, and in particular, suggests that this feature can be used to enhance certain machine learning and data science tasks. One recent paper has suggested that the finding *Betti numbers*, a task in topological data analysis, may be feasible on NISQ machines (Akhalwaya et al., 2022)—although the question of whether the Betti numbers for which the computation can be exponentially sped-up is practically relevant has been raised (McArdle et al., 2022). Other than this, it is worth noting, that while the training of PQCs as ML models is championed by its proponents as a naturally NISQ application, the quantum enhancement of linear algebra computations is expected to require full-scale (fault-tolerant) quantum computers.

The most famous quantum algorithm for linear algebra is Harrow, Hassidim, and Lloyd’s algorithm for solving a linear system (ubiquitously known as “HHL”) (Harrow et al., 2009). Specifically, consider the system of linear equations:

$$A\mathbf{x} = \mathbf{b}, \quad (1)$$

which is solved by inverting A and then pre-multiplying \mathbf{b} by A^{-1} to find \mathbf{x} . Classically this computation takes time that is worse than linear in the size of A (even if A is sparse), whereas in certain circumstances HHL runs in time that is only poly-logarithmic in the size of A —thus giving an exponential improvement over the best classical algorithms. HHL leverages the fact that an n -qubit quantum circuit is nothing more than a $2^n \times 2^n$ unitary matrix and thus, in a sense, a quantum computer is simply a machine that performs exponentially big matrix multiplications. When the matrix A is sparse and well-conditioned (the ratio of its largest to its smallest eigenvalues is not too big) then it is possible to construct the matrix operation A^{-1} as a quantum circuit. Moreover, this n -qubit circuit is only polynomially deep (in n) and hence the entire algorithm runs in time that is poly-logarithmic in the size of the matrix, A (a full complexity analysis also accounts for the fact that each attempt at inversion only succeeds with a certain probability, however, the overall poly-logarithmic complexity continues to hold, even when this is included).

HHL does, however, suffer from a number of caveats, one of which is the model for access to the data: it is assumed that the quantum computer has access to \mathbf{b} as some quantum state b , the preparation of which is not counted in the algorithm’s complexity. Indeed, one can immediately see that complexity at least linear in the size of \mathbf{x} (and hence the size of A) would be incurred even to read x , and so an overall poly-logarithmic complexity can only be possible if the appropriate quantum state is pre-prepared. The question of the need for a reasonable data access model is one of the problems raised by Scott Aaronson when discussing the potential for quantum advantage in machine learning applications (Aaronson, 2015)].

This issue was clarified and generalized by Ewin Tang, who showed that all proposed QML algorithms of this second approach can be *dequantized* if a classical algorithm is given commensurate data access (Tang, 2019; Chia et al., 2020a,b; Gilyén et al., 2022; Tang, 2021). “Dequantized” means that there is no *exponential* quantum advantage—although there could still be a practically useful *polynomial* advantage. That is, with Tang’s results, the quantum and classical algorithms both run in time polynomial in the logarithm of the size of the linear algebra objects in question, however that polynomial may be of much higher degree for the classical algorithm—thus the quantum algorithms may still provide a practically beneficial speed-up. This was indeed the case for Tang’s original dequantization breakthrough (Tang, 2019), where she proposed a “quantum-inspired” classical version algorithm of the *quantum recommendation system* of Kerenidis and Prakash (2017).

Finally, it is also pertinent to note that HHL itself has only been dequantized for low-rank matrix inversion (Chia et al., 2018, 2020b): there is still an exponential quantum advantage when the matrix to be inverted is full- (or close to full-) rank. Indeed, fast matrix inversion is still seen as being a potential “killer application” of quantum computing, and is a fertile area of research in, for example, partial differential equation (PDE) solving, when a finite-difference approach can be used to turn the PDE into a system of linear equations (Berry et al., 2017; Childs and Liu, 2020; Lloyd et al., 2020; Childs et al., 2021; Liu et al., 2021).

4. Quantum Data: The Holy Grail?

We have seen that the drawbacks of QML do not entirely diminish its potential practical utility. However, what is in some ways even more notable is that, until now, we have solely been talking about *classical* data, so we may ask: “what about *quantum* data?” That is, what if we have some quantum sensing or metrology process delivering quantum states directly as training data?

Taking the two approaches to QML in turn, when manipulating quantum rather than classical data, it is certainly more reasonable to expect that there may be some fundamental reason why a QML model may be required. In particular, even if the quantum data is immediately measured to give a classical sample, in general, such a sample may exhibit “nonclassical” correlations that cannot be reproduced by any reasonable-sized classical algorithm (for instance, as already noted, the correlations present in measurements of highly-entangled IQP circuits are believed to need exponentially large classical circuits to reproduce). Moreover, it has been shown that, in certain instances, barren plateaus are *not* present in generative modeling of *quantum* data (in this case, when the quantum state itself, not a measurement thereof, is delivered as training data to the model) (Kiani et al., 2022; Kieferova et al., 2021).

Turning to the second approach to QML, to provide commensurate data access to compare classical and quantum algorithms Tang’s dequantization results ordain the classical algorithms with *sample and query* access to the data. Suppose we have some N -element vector x , “query access” means the value x_i can be extracted (for any i), and “sample access” means we sample a number, i between 0 and $N - 1$ with probability $x_i / \sum_j x_j$. If the quantum state is prepared from classical data then (as Tang asserts) it is reasonable to assume that sample and query access could be attained in about the same number of operations. If, however, the data is presented as a quantum state, then only sample access is available to a classical algorithm (sample access is obtained simply by measuring the quantum state in question). This in turn implies that, when the input is quantum data, the dequantization results no longer necessarily hold, and the possibility of exponential quantum advantage is upheld.

5. Monte Carlo or Bust?

Responding by basically saying “soon there may be even more data which is quantum in nature and thus intrinsically *needs* QML” is only really half an answer to the motivating question of what quantum computing can do to help processing vast datasets. For the implicit emphasis in the question was on the data we already have, and expect in the immediate future. To answer this, it is helpful to step back and ask: what is it we want *from* these large datasets? Invariably, the aim will be to extract some quantities pertaining to the dataset as a whole and, moreover (even if it is not immediately thought of in these terms), such quantities will usually amount to some sort of expectation of the distribution that the data has been sampled from (or simple combinations of expectation values). For instance, obviously recognizable quantities such as the mean and higher moments are expectation values, however other quantities such as various measures of risk will be found by computing an appropriate expectation. Additionally, quantities that are usually thought of not as *expectations* but rather as *probabilities* such as the probability of rain in a weather forecast will actually be found by numerically integrating over a number of marginal parameters².

Such a desideratum coincides with one of the (still relatively few) fundamental computational tasks that we know admits a provable quantum advantage, namely quantum Monte Carlo integration (QMCI) (Montanaro, 2015). Furthermore, significant progress has been made to allow such an advantage to be realized with minimal quantum resources.

Monte Carlo integration (MCI) is the process of numerically estimating some expectation value

$$\mathbb{E}(f(x)) = \int_x f(x)p(x)dx, \quad (2)$$

²More generally, probabilities *are* expectations (over indicator functions). For instance, $p(A) = \mathbb{E}(I(x \in A))$, where I is the indicator function.

which cannot be evaluated analytically, but where the probability distribution, $p(x)$, can be sampled from (and $f(\cdot)$ is some function). Notably, on any digital computer (classical or quantum) the integral will actually be a sum, owing to the necessary quantization and truncation of the support of $p(x)$, thus:

$$\mathbb{E}(f(x)) = \sum_x f(x)p(x) \approx \frac{1}{q} \sum_{i=1}^q f(X_i), \tag{3}$$

where $X_i \sim p(x)$ are i.i.d samples. The approximate equality represents the process of MCI, and in particular, the mean squared error (MSE) is $\mathcal{O}(q^{-1})$. When performing high-dimensional integrals numerically, MCI is the most efficient method. (Note that quasi-Monte Carlo (Morokoff and Caflisch, 1995) and other non-i.i.d classical methods have better convergence in q , but suffer the *curse of dimensionality*—the complexity grows exponentially in the number of dimensions—and hence are inefficient for high-dimensional integrals.)

If we break down (classical) MCI, we can see that it amounts to a very simple three-step process: sample from $p(x)$, apply the function $f(\cdot)$, and then average over many such samples with the function applied. In QMCI, there is an analogous three-step process: first, we take as an input a *state preparation circuit*, P , which prepares a quantum state p that samples from $p(x)$ when measured on the computational basis

$$|p\rangle = \sum_x \sqrt{p(x)}|x\rangle, \tag{4}$$

where, for simplicity, we assume that $p(x)$ is supported over some $N = 2^n$ points.

Second, a circuit, denoted R , is applied to $|p\rangle$ with one further qubit appended such that the following state is prepared:

$$R|p\rangle|0\rangle = \sum_x \sqrt{p(x)}|x\rangle \left(\sqrt{1-f(x)}|0\rangle + \sqrt{f(x)}|1\rangle \right). \tag{5}$$

The circuit R thus encodes the function applied, $f(\cdot)$ and in particular has the property that, when measured, the appended qubit has a probability of being one equal to

$$\sum_x p(x)f(x), \tag{6}$$

according to the Born rule, which tells us to square and sum all terms in the sum where the final qubit is in state $|1\rangle$. This is exactly the value we are trying to estimate with MCI, and it turns out that the (quantum) algorithm *quantum amplitude estimation* (QAE) can estimate this with MSE $\mathcal{O}(q^{-2})$, where q is now the number of uses of the circuit P (Brassard et al., 2002). Accepting (for now) that this quantity “ q ” corresponds to that in classical MCI, we can see that this represents a quadratic advantage in convergence: for a certain desired MSE, only about square root as many samples are required quantumly as would be classically.

However, QAE was not originally seen as an ideal candidate as a source of *near term* quantum advantage, as it uses *quantum phase estimation* (Kitaev, 1996) an algorithm that is expected to require full-scale quantum computers. That all changed with the advent of *amplitude estimation without phase estimation* (Suzuki et al., 2020), which showed how to obtain the full quadratic quantum advantage, but using a number of shallow-depth circuits and classical post-processing to estimate the expectation value. A number of other proposals have since followed in the same vein (Aaronson and Rall, 2020; Giurgica-Tiron et al., 2022; Nakaji, 2020; Grinko et al., 2021).

Two more, complementary, breakthroughs have further fueled the hope that QMCI can be a source of near-term quantum advantage:

1. *Noise-aware QAE* (Herbert et al., 2021) takes an advantage of the fact that QAE circuits have a very specific structure to handle device noise as if it were estimation uncertainty. This suggests that

significantly less error correction may be needed to achieve a useful advantage in QMCI, compared to other calculations of comparable size.

2. *Quantum Monte Carlo integration: the full advantage in minimal circuit depth* (Herbert, 2022) shows how to decompose the Monte Carlo integral as a Fourier series such that the circuit R , which may in general constitute an unreasonably large contribution to the total circuit depth, can be replaced by minimally deep circuits of rotation gates (this procedure is hereafter referred to as “Fourier QMCI”).

In particular, the second of these informs us of the sorts of applications that are likely to see a (relatively) early quantum advantage. For instance, Fourier QMCI is especially advantageous for numerical integrals that can be decomposed as a product of some $p(x)$, for which a suitable encoding can be prepared by a relatively shallow state preparation circuit (i.e., as described in equation 4); and some $f(x)$ which can be extended as a piecewise periodic function whose Fourier series can be calculated and satisfies certain smoothness conditions. Areas in which computationally intensive numerical integrations are commonplace include computational fluid dynamics and high-energy physics—indeed, in the latter QMCI solutions have begun to be explored (Agliardi et al., 2022).

For general numerical integrals, the randomness in MCI is a device to enable efficient numerical integration; however, for most data-centric MCIs, we do expect that $p(x)$ will have a more literal role as the probability distribution from which the data has been sampled. This in turn raises the question of how to construct the state preparation circuit, P .

From a theoretical point of view, it is always possible to construct a suitable P from the corresponding classical sampling process (Herbert, 2021) (this resolves the earlier question of why the number of classical samples can be compared to the number of quantum uses of the state preparation circuit—the two uses of “ q ” that were treated as equivalent), and such a result may well ultimately find practical application. For example, ANNs trained as generative models are instances of classical sampling processes, and so such generative models can be converted into suitable circuits, P . However, in the near-term such quantum circuits are likely to be infeasibly deep, and so instead we should focus on applications that leverage the quantum advantage in a more direct manner. In particular, the quantum advantage is manifested in a quadratic reduction in the number of samples required to attain a certain required accuracy, and so applications, where a very large number of samples are required, provide a good starting point—especially when those samples are from (relatively) simple stochastic processes.

With regards to data-centric engineering and science, one helpful way to think about which applications that QMCI will impact in the near-term is in terms of the distinction between parametric and nonparametric models. For we have already established that we must operate on some model for the generation of the observed data: in cases where the best (classical) approach is to use the dataset to fit parameters from some parameterized family of distributions then we expect the corresponding circuit P to be relatively easy to construct. Conversely, if the model is nonparametric (or even something like a deep neural network that, by some, may be regarded as parametric—just with an enormous number of parameters that do not correspond in the natural and straightforward way to the statistics of the observed data as do traditional parametric models) then there is a real risk that the circuit P will be hard to construct in the near-term.

Turning now to some specific examples, one promising area concerns time-series data, where some large dataset is used to fit the parameters of some model such as a hidden Markov model, auto-regressive model, or auto-regressive moving average model (amongst many others). Notably, using an abundance of historical data to tune the parameters of time-series models is commonplace throughout applications of MCI in financial and actuarial engineering. Indeed, it is the case that the vast majority of the early QMCI literature has focused on financial applications (Rebentrost and Lloyd, 2018; Rebentrost et al., 2018; Egger et al., 2020, 2021; Orús et al., 2019; Woerner and Egger, 2019; Bouland et al., 2020; Kaneko et al., 2020; Stamatopoulos et al., 2020, 2022; An et al., 2021; Chakrabarti et al., 2021; Herman et al., 2022).

That is not to say, however, that QMCI will ultimately only find application in financial engineering. For instance, the “function applied” in financial applications of QMCI usually corresponds to some sort of

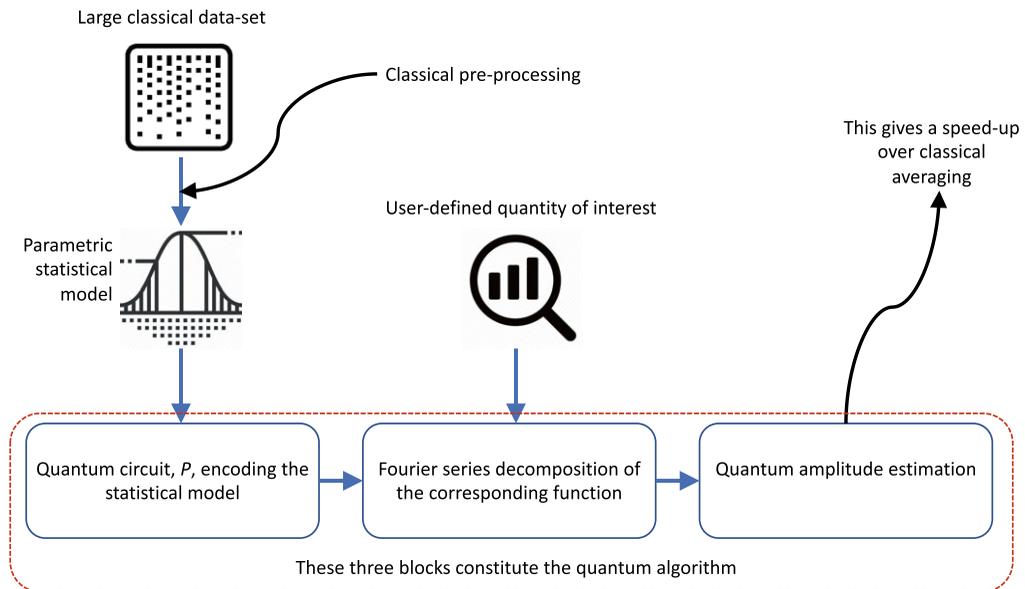


Figure 4. The general framework of Fourier quantum Monte Carlo integration (QMCI) applied to large classical datasets. In the future (as quantum hardware matures), the parametric statistical model may be optionally replaced by a nonparametric (or ML) model; or perhaps even a quantum machine learning model. In all cases, QMCI only ever requires sample access to the data.

thresholded average—most obviously when calculating the expected return on a European option the function applied is essentially a ReLU function—and functions of these types (i.e., piece-wise linear) are likely to be similar to suitable functions to calculate notions of “cost” in, for example, supply-chain and logistic-optimization applications (Ozkan and Kilic, 2019). Moreover, Monte Carlo methods are widely used in virtually every area of data-centric engineering and science from medical imaging (Chen et al., 2002) to chemical, biochemical, and environmental (Sin and Espuña, 2020) to energy modeling (Dhaundiyal et al., 2019) and handling big data in general (Ji and Li, 2016). Indeed, rather than exhaustively cataloging every conceivable application of QMCI to data-centric engineering and science, a better approach is to set out the general framework, as is shown in Figure 4, such that expert readers can see how the quantum advantage may be realized in their respective domains.

At first sight, the scheme laid out in Figure 4 may appear to side-step the central question of how quantum computing will help with large datasets, as the data handling itself represents a classical pre-processing step. However, this is simply a reflection of the reality that data-loading is generally hard and that it is prudent to focus on tasks where there is an unequivocal quantum advantage (i.e., in estimate converge as a function of number of samples). Moreover, many data-centric applications (e.g., those in finance) are indeed of the form where the data-loading can be achieved by fitting the parameters of some statistical model, but thereafter the statistical estimation is the bottleneck: and it is exactly those applications that quantum computing can incontrovertibly enhance.

6. Outlook and Speculation

In this article, I have deliberately homed in on QMCI as a likely source of near-term quantum advantage in data science applications. This is a personal view, and many others still focus on the possibility that there will be genuine useful “NISQ advantage” (e.g., using PQC as ML models). However, since this *is* my view, as explicitly noted in the introduction, I see the established division into the NISQ and full-scale eras of quantum computing as overly simplistic. Instead, I prefer to think about how we can use *resource*

constrained quantum hardware, that is neither strictly NISQ (there may be enough qubits for some mild error correction) but neither is full-scale (in the sense that full-scale algorithms may be typified by an attitude of being unconcerned with resource demands that appear only as negligible contributions to asymptotic complexity)—and hence bridges between the established ideas of NISQ and full-scale quantum computing. Certainly, it is true that the first applications of quantum computing with a provable advantage will be those where consideration of resource management has been given special attention. For reasons that have recurred throughout this article, I am of the view that QMCI shows great promise to be such an application.

In light of this, I feel it is incumbent upon me to offer some predictions about when this will come to fruition. As any responsible quantum computing researcher will attest, such predictions are hard to make at present, but now that the leading quantum hardware manufacturers are beginning to commit to roadmaps with quantified targets, it is at least possible for theorists to make loose predictions contingent on said roadmaps being met. The first comment to make is that both IBM and Google have committed to reaching 1 million qubit by the end of the decade (Google, 2021; IBM, 2021), and such a figure would certainly be enough for useful quantum advantage in many applications including QMCI.

To make a more precise prediction, with regards to QMCI, we benefit from the fact that a leading research group has published a paper setting out suitable benchmarks to facilitate resource estimation for when there will be a useful quantum advantage in QMCI (Chakrabarti et al., 2021). The specific focus of the paper in question is on QMCI applications in finance, but the broad principle is likely to extend to other applications. We have estimated that our Fourier QMCI algorithm (Herbert, 2022) reduces resource requirements (number of quantum operations) by at least 30% to over 90% in some cases for the benchmarks set out, and if the circuits were re-constructed in a slightly different way we have estimated that the total number of physical superconducting qubits required would be in the 1,000s to 10,000 s. The exact value within this range depends on whether qubit qualities improve enough for low-overhead error-correction codes (Tomita and Svore, 2014) to be practical; and in particular, whether the overhead can be reduced by exploiting asymmetries in the noise (Ataides et al., 2021; Higgott et al., 2022)—both of which remain very active research topics.

This resource estimation places a useful advantage in QMCI in the 5-year horizon for the leading superconducting roadmaps and a similar timescale is likely for trapped-ion devices, for example (Quantinuum, 2022), note that trapped-ion quantum computers typically have many fewer, but higher-quality qubits, and tend to have less specific roadmaps). This is also consistent with other predictions, for example, that of *QCWare* and *Goldman Sachs* (Goldman Sachs and QC Ware, 2021).

Rather than leaving the prediction at that, it is worth “playing devil’s advocate” and exploring whether this is unreasonably hubristic—particularly in light of a widely-circulated paper suggesting that near-term quantum advantage would be hard to obtain for algorithms exhibiting only a quadratic speed-up (Babbush et al., 2021). There are three central claims in (Babbush et al., 2021), which provide a useful framework to scrutinize the legitimacy of my prediction:

1. *Quadratic-advantage quantum algorithms are dominated by circuits of Toffoli gates, which are extremely expensive to implement using error-corrected quantum computation.* This is certainly true for un-optimized algorithms, however, Fourier QMCI (Herbert, 2022) moves to classical post-processing exactly those Toffoli-heavy circuits, while upholding the full quantum advantage.
2. *Error-correction overheads are, in any case, expensive.* Again, this is true, which is why bespoke approaches to error correction that exploit the specific algorithm structure and handle a certain amount of the device noise at the application level (as does noise-aware QAE [Herbert et al., 2021]) are crucial to achieve near-term quantum advantage.
3. *Algorithms for which there is a quadratic quantum advantage can typically be massively parallelized when performed classically, meaning that a useful quantum advantage only occurs at much larger problems sizes, once the parallelism has been accounted for.* This is again true in the case of MCI, but leaves out one very significant detail, namely that QMCI can itself be massively parallelized.

The third item reveals an important subtlety: in the (quantum computing) sector, we obsess about the “route to scale” in terms of adding ever more qubits to the same chip—but once quantum computers reach moderate scale, it will be just as important to scale up the *number* of quantum computers available. Just as classical HPC can accelerate classical data-centric applications by running calculations on different cores in parallel, so it will be the case that running quantum circuits in parallel will be crucial for an early advantage in data-centric applications. (To be clear, here the parallelization is classical—there is no need for entangled connections between the different cores—although the subject of *distributed quantum computing*, where there are entangled connections between the different quantum cores is a fascinating topic in its own right, see, e.g., Cirac et al., 1999; Cuomo et al., 2020; Meter et al., 2007.) Indeed, for our above resource estimates, we have only quantified when quantum hardware will be capable of running the requisite quantum circuits—in order for this to translate into a practical benefit, sufficient quantum cores must be available to the user.

So where does that leave us? That there exist data science-relevant quantum algorithms, such as QMCI, that exhibit a provable quantum advantage, coupled with the fact that the leading players are now beginning to scale up quantum hardware, provides a great cause for optimism that quantum computing will impact on data-centric engineering and science applications in the near- to medium-term. However, in quantum computing we have learned that optimism must always go hand-in-hand with caution: there are serious engineering challenges at every layer, from the design of the quantum computer itself, to the control software, to the optimization of the algorithms that run the desired applications. In particular, we know that preparing quantum states that encode the relevant model or distribution is usually the bottleneck in QMCI—cracking this *data-loading problem* is the key to unleashing the power of quantum computation onto myriad applications within finance, supply chain & logistics, medical imaging, and energy modeling.

Acknowledgments. My thanks to Ewin Tang and Seth Lloyd for kindly answering my questions regarding quantum linear algebra and the dequantization thereof—of course, any remaining errors or misconceptions are my own. I thank Alexandre Krajenbrink, Sam Duffield, and Konstantinos Meichanetzidis for reviewing and providing valuable suggestions. I also thank the reviewers at DCE, whose suggestions helped to further improve the work.

Competing Interests. The author declares no competing interests exist.

Data Availability Statement. Data availability is not applicable to this article as no new data were created or analyzed in this study.

Author Contributions. S.H. contributed to the conceptualization, formal analysis, investigation, writing—original draft, and review and editing.

Ethics Statement. The research meets all ethical guidelines, including adherence to the legal requirements of the study country.

Funding Statement. This work received no specific grant from any funding agency, commercial or not-for-profit sectors.

References

- Aaronson S (2015) Read the fine print. *Nature Physics* 11(4), 291–293.
- Aaronson S (2022) How much structure is needed for huge quantum speedups?
- Aaronson S and Rall P (2020) Quantum approximate counting, simplified. In *Symposium on Simplicity in Algorithms*, 24–32.
- Agliardi G, Grossi M, Pellen M and Prati E (2022) Quantum integration of elementary particle processes. *Physics Letters B*, 137228.
- Aharonov D and Ben-Or M (2008) Fault-tolerant quantum computation with constant error rate. *SIAM Journal on Computing* 38(4), 1207–1282.
- Akhalwaya IY, Ubaru S, Clarkson KL, Squillante MS, Jejjala V, He Y-H, Naidoo K, Kalantzis V and Horesh L (2022) Exponential advantage on noisy quantum computers.
- An D, Linden N, Liu J-P, Montanaro A, Shao C and Wang J (2021) Quantum-accelerated multilevel Monte Carlo methods for stochastic differential equations in mathematical finance. *Quantum* 5, 481.
- Arrasmith A, Cerezo M, Czarnik P, Cincio L and Coles PJ (2021) Effect of barren plateaus on gradient-free optimization. *Quantum* 5, 558.

- Arute F, Arya K, Babbush R, Bacon D, Bardin JC, Barends R, Biswas R, Boixo S, Brandao FGSL, Buell DA, Burkett B, Chen Y, Chen Z, Chiaro B, Collins R, Courtney W, Dunsworth A, Farhi E, Foxen B, Fowler A, Gidney C, Giustina M, Graff R, Guerin K, Habegger S, Harrigan MP, Hartmann MJ, Ho A, Hoffmann M, Huang T, Humble TS, Isakov SV, Jeffrey E, Jiang Z, Kafri D, Kechedzhi K, Kelly J, Klimov PV, Knysh S, Korotkov A, Kostritsa F, Landhuis D, Lindmark M, Lucero E, Lyakh D, Mandrà S, McClean JR, McEwen M, Megrant A, Mi X, Michielsen K, Mohseni M, Mutus J, Naaman O, Neeley M, Neill C, Niu MY, Ostby E, Petukhov A, Platt JC, Quintana C, Rieffel EG, Roushan P, Rubin NC, Sank D, Satzinger KJ, Smelyanskiy V, Sung KJ, Trevithick MD, Vainsencher A, Villalonga B, White T, Yao ZJ, Yeh P, Zalcman A, Neven H and Martinis JM (2019) Quantum supremacy using a programmable superconducting processor. *Nature* 574(7779), 505–510.
- Ataides JPB, Tuckett DK, Bartlett SD, Flammia ST and Brown BJ (2021) The XZZX surface code. *Nature Communications* 12(1).
- Babbush R, McClean JR, Newman M, Gidney C, Boixo S and Neven H (2021) Focus beyond quadratic speedups for error-corrected quantum advantage. *PRX Quantum* 2(1).
- Benedetti M, Garcia-Pintos D, Perdomo O, Leyton-Ortega V, Nam Y and Perdomo-Ortiz A (2019a) A generative modeling approach for benchmarking and training shallow quantum circuits. *npj Quantum Information* 5(1).
- Benedetti M, Lloyd E, Sack S and Fiorentini M (2019b) Parameterized quantum circuits as machine learning models. *Quantum Science and Technology* 4(4), 043001.
- Benioff P (1980) The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. *Journal of Statistical Physics* 22, 563–591.
- Berry DW, Childs AM, Ostrander A and Wang G (2017) Quantum algorithm for linear differential equations with exponentially improved dependence on precision. *Communications in Mathematical Physics* 356(3), 1057–1081.
- Bittel L and Kliesch M (2021) Training variational quantum algorithms is NP-hard.
- Bouland A, van Dam W, Joorati H, Kerenedis I and Prakash A (2020) Prospects and challenges of quantum finance.
- Brassard G, Høyer P, Mosca M and Tapp A (2002) Quantum amplitude amplification and estimation. In *Quantum Computation and Information*. Providence, RI: American Mathematical Society, pp. 53–74.
- Bremner MJ, Jozsa R and Shepherd DJ (2010) Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 467(2126), 459–472.
- Briegel HJ, Browne DE, Dür W, Raussendorf R and den Nest MV (2009) Measurement-based quantum computation. *Nature Physics* 5(1), 19–26.
- Cerezo M, Sone A, Volkoff T, Cincio L and Coles PJ (2021) Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nature Communications* 12(1).
- Chakrabarti S, Krishnakumar R, Mazzola G, Stamatopoulos N, Woerner S and Zeng WJ (2021) A threshold for quantum advantage in derivative pricing. *Quantum* 5, 463.
- Chang SY, Agnew E, Combarro EF, Grossi M, Herbert S and Vallecorsa S (2022) Running the Dual-PQC GAN on noisy simulators and real quantum hardware.
- Chang SY, Herbert S, Vallecorsa S, Combarro EF and Duncan R (2021) Dual-parameterized quantum circuit GAN model in high energy physics. *EPJ Web of Conferences* 251, 03050.
- Chen JZ, Pizer SM, Chaney EL and Joshi SC (2002) Medical image synthesis via Monte Carlo simulation. In *Proceedings of the 5th International Conference on Medical Image Computing and Computer-Assisted Intervention-Part I, MICCAI '02*. Berlin, Heidelberg: Springer-Verlag, pp. 347–354.
- Chia N-H, Gilyén A, Li T, Lin H-H, Tang E and Wang C (2020a) Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. ACM.
- Chia N-H, Gilyén A, Lin H-H, Lloyd S, Tang E and Wang C (2020b) Quantum-inspired algorithms for solving low-rank linear equation systems with logarithmic dependence on the dimension. In Cao Y, Cheng S-W and Li M (eds), *31st International Symposium on Algorithms and Computation (ISAAC 2020)*, Volume 181 of *Leibniz International Proceedings in Informatics (LIPIcs)*. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum für Informatik, pp. 47:1–47:17.
- Chia N-H, Lin H-H and Wang C (2018) Quantum-inspired sublinear classical algorithms for solving low-rank linear systems.
- Childs AM and Liu J-P (2020) Quantum spectral methods for differential equations. *Communications in Mathematical Physics* 375(2), 1427–1457.
- Childs AM, Liu J-P and Ostrander A (2021) High-precision quantum algorithms for partial differential equations. *Quantum* 5, 574.
- Cirac JI, Ekert AK, Huelga SF and Macchiavello C (1999) Distributed quantum computation over noisy channels. *Physical Review A* 59(6), 4249–4254.
- Coyle B, Mills D, Danos V and Kashefi E (2020) The born supremacy: Quantum advantage and training of an Ising born machine. *npj Quantum Information* 6(1).
- Cuomo D, Caleffi M and Cacciapuoti AS (2020) Towards a distributed quantum computing ecosystem. *IET Quantum Communication* 1(1), 3–8.
- Deutsch D (1985) Quantum theory, the Church–Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400:117–197.

- Dhaundiyal A, Singh SB, Atsu D and Dhaundiyal R** (2019) Application of Monte Carlo simulation for energy modeling. *Journal of the American Chemical Society* 4, 4984–4990.
- Dong D, Chen C, Li H and Tarn T-J** (2008) Quantum reinforcement learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 38(5), 1207–1220.
- Egger DJ, Gambella C, Marecek J, McFaddin S, Mevissen M, Raymond R, Simonetto A, Woerner S and Yndurain E** (2020) Quantum computing for finance: State-of-the-art and future prospects. *IEEE Transactions on Quantum Engineering* 1, 1–24.
- Egger DJ, Gutiérrez RG, Mestre JC and Woerner S** (2021) Credit risk analysis using quantum computers. *IEEE Transactions on Computers* 70(12), 2136–2145.
- Farhi E, Goldstone J, Gutmann S and Sipser M** (2000) Quantum computation by adiabatic evolution.
- Feynman RP** (1982) Simulating physics with computers. *International Journal of Theoretical Physics* 21, 467–488.
- Gilyén A, Song Z and Tang E** (2022) An improved quantum-inspired algorithm for linear regression. *Quantum* 6, 754.
- Giurgica-Tiron T, Kerendis I, Labib F, Prakash A and Zeng W** (2022) Low depth algorithms for quantum amplitude estimation. *Quantum* 6, 745.
- Goldman Sachs and QC Ware** (2021) Collaboration brings new way to price risky assets within reach of quantum computers.
- Google** (2021) *Quantum Roadmap*.
- Grinko D, Gacon J, Zoufal C and Woerner S** (2021) Iterative quantum amplitude estimation. *npj Quantum Information* 7(1).
- Grover LK** (1996) A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, STOC '96*. New York: Association for Computing Machinery, pp. 212–219.
- Harrow AW** (2020) Small quantum computers and large classical data sets.
- Harrow AW, Hassidim A and Lloyd S** (2009) Quantum algorithm for linear systems of equations. *Physical Review Letters* 103 (15).
- Harrow AW and Montanaro A** (2017) Quantum computational supremacy. *Nature* 549(7671), 203–209.
- Havlíček V, Córcoles AD, Temme K, Harrow AW, Kandala A, Chow JM and Gambetta JM** (2019) Supervised learning with quantum-enhanced feature spaces. *Nature* 567(7747), 209–212.
- Herbert S** (2021) Every Classical Sampling Circuit is a Quantum Sampling Circuit.
- Herbert S** (2022) Quantum Monte Carlo integration: the full advantage in minimal circuit depth. *Quantum* 6, 823.
- Herbert S, Guichard R and Ng D** (2021) Noise-aware quantum amplitude estimation.
- Herman D, Googin C, Liu X, Galda A, Safo I, Sun Y, Pistoia M and Alexeev Y** (2022) A survey of quantum computing for finance.
- Higgott O, Bohdanowicz TC, Kubica A, Flammia ST and Campbell ET** (2022) Fragile boundaries of tailored surface codes.
- Hubregtzen T, Pichlmeier J, Stecher P and Bertels K** (2020) Evaluation of parameterized quantum circuits: on the relation between classification accuracy, expressibility and entangling capability.
- IBM** (2021) *Quantum Roadmap*.
- Jerbi S, Gyurik C, Marshall SC, Briegel HJ and Dunjko V** (2021) Parametrized quantum policies for reinforcement learning.
- Ji H and Li Y** (2016) *Monte Carlo Methods and Their Applications in Big Data Analysis*, 125–139.
- Jordan S** (2011) Quantum Algorithm Zoo.
- Kaneko K, Miyamoto K, Takeda N and Yoshino K** (2020) Quantum pricing with a smile: Implementation of local volatility model on quantum computer.
- Kerendis I and Prakash A** (2017) Quantum recommendation systems. In *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*.
- Kiani BT, De Palma G, Marvian M, Liu Z-W and Lloyd S** (2022) Learning quantum data with the quantum earth mover's distance. *Quantum Science and Technology* 7(4), 045002.
- Kieferova M, Carlos OM and Wiebe N** (2021) Quantum generative training using Rényi divergences.
- Kitaev A** (2003) Fault-tolerant quantum computation by anyons. *Annals of Physics* 303(1), 2–30.
- Kitaev AY** (1996) Quantum measurements and the abelian stabilizer problem. *Electronic Colloquium on Computational Complexity* 3.
- Knill E, Laflamme R and Milburn GJ** (2001) A scheme for efficient quantum computation with linear optics. *Nature* 409, 46–52.
- Knill E, Laflamme R and Zurek WH** (1998) Resilient quantum computation: Error models and thresholds. *Proceedings of the Royal Society of London. Series A: Mathematical Physical and Engineering Sciences* 454(1969), 365–384.
- Lenstra AK, Lenstra HW, Manasse MS and Pollard JM** (1993) The number field sieve. In Lenstra AK and Lenstra HW (eds), *The Development of the Number Field Sieve*. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 11–42.
- Liu J-P, Kolden HØ, Krovi HK, Loureiro NF, Trivisa K and Childs AM** (2021) Efficient quantum algorithm for dissipative nonlinear differential equations. *Proceedings of the National Academy of Sciences* 118(35).
- Lloyd S, De Palma G, Gokler C, Kiani B, Liu Z-W, Marvian M, Tennie F and Palmer T** (2020) Quantum algorithm for nonlinear differential equations.
- Lloyd S and Weedbrook C** (2018) Quantum generative adversarial learning. *Physical Review Letters* 121(4).
- Luongo A** (2022) Quantum Algorithms for Data Analysis.
- Madsen L, Laudenbach F, Askarani M, Rortais F, Vincent T, Bulmer J, Miatto F, Neuhaus L, Helt L, Collins M, Lita A, Gerrits T, Nam S, Vaidya V, Menotti M, Dhand I, Vernon Z, Quesada N and Lavoie J** (2022) Quantum computational advantage with a programmable photonic processor. *Nature* 606, 75–81.
- Manin Y** (1980) Computable and Uncomputable.

- McArdle S, Gilyén A and Berta M** (2022) A streamlined quantum algorithm for topological data analysis with exponentially fewer qubits.
- McClean JR, Boixo S, Smelyanskiy VN, Babbush R and Neven H** (2018) Barren plateaus in quantum neural network training landscapes. *Nature Communications* 9(1).
- Meter RV, Nemoto K and Munro W** (2007) Communication links for distributed quantum computation. *IEEE Transactions on Computers* 56(12), 1643–1653.
- Montanaro A** (2015) Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 471(2181), 20150301.
- Morokoff WJ and Caffisch RE** (1995) Quasi-Monte Carlo integration. *Journal of Computational Physics* 122(2), 218–230.
- Nakaji K** (2020). Faster amplitude estimation. *Quantum Information and Computation* 20(13&14), 1109–1123.
- Nielsen MA and Chuang IL** (2010) *Quantum Computation and Quantum Information. 10th Anniversary Edition*. Cambridge University Press.
- Orús R, Mugel S and Lizaso E** (2019) Quantum computing for finance: Overview and prospects. *Reviews in Physics* 4, 100028.
- Ozkan O and Kilic S** (2019) A Monte Carlo simulation for reliability estimation of logistics and supply chain networks. *IFAC-PapersOnLine* 52(13), 2080–2085.
- Peruzzo A, McClean J, Shadbolt P, Yung M-H, Zhou X-Q, Love PJ, Aspuru-Guzik A and O’Brien JL** (2014) A variational eigenvalue solver on a photonic quantum processor. *Nature Communications* 5(1).
- Preskill J** (2018) Quantum computing in the NISQ era and beyond. *Quantum* 2, 79.
- Preskill J** (2021) Quantum Computing 40 Years Later.
- Quantinuum** (2022) Quantum Roadmap.
- Rebentrost P, Gupta B and Bromley TR** (2018) Quantum computational finance: Monte Carlo pricing of financial derivatives. *Physical Review A* 98(2).
- Rebentrost P and Lloyd S** (2018) Quantum computational finance: quantum algorithm for portfolio optimization.
- Schreiber FJ, Eisert J and Meyer JJ** (2022) Classical surrogates for quantum learning models.
- Shalaginov MY and Dubrovsky M** (2022) Quantum proof of work with parameterized quantum circuits.
- Shor P** (1994) Algorithms for quantum computation: Discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 124–134.
- Shor P** (1996) Fault-tolerant quantum computation. In *Proceedings of 37th Conference on Foundations of Computer Science*, 56–65.
- Sin G and España A** (2020) Editorial: Applications of Monte Carlo method in chemical, biochemical and environmental engineering. *Frontiers in Energy Research* 8.
- Stamatopoulos N, Egger DJ, Sun Y, Zoufal C, Iten R, Shen N and Woerner S** (2020) Option pricing using quantum computers. *Quantum* 4, 291.
- Stamatopoulos N, Mazzola G, Woerner S and Zeng WJ** (2022) Towards quantum advantage in financial market risk using quantum gradient algorithms. *Quantum* 6, 770.
- Suzuki Y, Uno S, Raymond R, Tanaka T, Onodera T and Yamamoto N** (2020) Amplitude estimation without phase estimation. *Quantum Information Processing* 19(2).
- Tang E** (2019) A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019. New York: Association for Computing Machinery, pp. 217–228.
- Tang E** (2021) Quantum principal component analysis only achieves an exponential speedup because of its state preparation assumptions. *Physical Review Letters* 127, 060503.
- Thanasilp S, Wang S, Nghiem NA, Coles PJ, Cerezo M.** (2021) *Subtleties in the trainability of quantum machine learning models.*
- Tomita Y and Svore KM** (2014) Low-distance surface codes under realistic quantum noise. *Physical Review A* 90(6).
- Wang S, Fontana E, Cerezo M, Sharma K, Sone A, Cincio L and Coles PJ** (2021) Noise-induced barren plateaus in variational quantum algorithms. *Nature Communications* 12(1).
- Woerner S and Egger DJ** (2019) Quantum risk analysis. *npj Quantum Information* 5(1).
- Zhong H-S, Wang H, Deng Y-H, Chen M-C, Peng L-C, Luo Y-H, Qin J, Wu D, Ding X, Hu Y, Hu P, Yang X-Y, Zhang W-J, Li H, Li Y, Jiang X, Gan L, Yang G, You L, Wang Z, Li L, Liu N-L, Lu C-Y and Pan J-W** (2020) Quantum computational advantage using photons. *Science* 370(6523), 1460–1463.
- Zoufal C** (2021) Generative Quantum Machine Learning.
- Zoufal C, Lucchi A and Woerner S** (2019) Quantum generative adversarial networks for learning and loading random distributions. *npj Quantum Information* 5(1).