

Quantum Computing 2022

James D. Whitfield*, †, Jun Yang*, Weishi Wang*, Joshuah T. Heath*, and Brent Harrison*

* Department of Physics and Astronomy, Dartmouth College

Hanover, New Hampshire, USA 03755

†AWS Center for Quantum Computing

Pasadena, California, USA 91125

Abstract—Quantum technology is full of figurative and literal noise obscuring its promise. In this overview, we will attempt to provide a sober assessment of the promise of quantum technology with a focus on computing. We provide a tour of quantum computing and quantum technology that is aimed to be comprehensible to scientists and engineers without becoming a popular account. The goal is not a comprehensive review nor a superficial introduction but rather to serve as a useful map to navigate the hype, the scientific literature, and upcoming press releases about quantum technology and quantum computing. We have aimed to cite the most recent topical reviews, key results, and guide the reader away from fallacies and towards active discussions in the current quantum computing literature. The goal of this article was to be pedantic and introductory without compromising on the science.

The field of quantum technology, especially quantum computing technology, has emerged as an active area of academic and corporate research and development. It has also emerged as an area of heavy investment by companies, governments, and private investors worldwide. For example, worldwide investment in 2021 was estimated at \$24.4 billion with the United States appropriating \$1.2 billion over five years through the National Quantum Initiative Act [1], [2]. So far this global investment has paid off in rampant technological progress, flagship experiments, and major intellectual developments.

Quantum technologies are the detectors, devices, and communication systems that rely on uniquely quantum resources, and quantum computing is the use of these quantum technologies to speed up or otherwise improve solutions to computational tasks and problems. The encapsulation of quantum mechanics has yielded a number of quantum technologies useful for storing and manipulating quantum information. Taking the position that information is physical [3], the objects carrying or storing that information may exhibit uniquely quantum behavior and then we refer to the information stored as *quantum information*. If we consider the smallest unit of information as a bit taking values of zero or one, then, similarly, the smallest unit of quantum information is the qubit (or quantum bit).

In Section I an overview of key modern quantum technologies is given. The theory driving these devices is given in Section II. Our presentation of quantum mechanics is somewhat novel in that we begin with the quantum state as a quantum probability density matrix extended directly from ordinary probability density vectors. These key ideas prepare

for Section III where we introduce the gate model of quantum computation and some equivalent formulations. In Section IV, we discuss the claims and counter-claims of quantum primacy¹. Algorithms for quantum computers are discussed in Section V where we have provided an overview of the quantum algorithms literature, including the quantum singular value transformation framework. In the last section, Section VI, we give an outlook on quantum technology including next steps for the reader including some sample code.

I. QUANTUM TECHNOLOGIES

Conventional computers² rely on semi-conductor technologies where electrical currents are the primary information carriers. Quantum computing technology is far more diverse in that there is a wider range of physical systems used as information carriers. In this section, we focus on technologies for realizing quantum information carriers that have received the most attention and traction. This section gives a high-level overview but for a more comprehensive view of quantum computing technologies see [8], [9].

We will begin with the most widely adopted technology thus far, namely superconducting qubits. Then we turn toward quantum computing systems where individual atoms are used as information carriers. Although not yet commercially relevant, nuclear magnetic resonance (NMR) quantum computing is included for historical and pedagogical reasons. From there we give mention to quantum light (photons) used to store and manipulate quantum information. Finally, this section will close with a brief look at other technologies that may be relevant in the coming years.

Superconducting qubits are the most widely available device architecture for quantum computing and have received the most commercial attention thus far [10], [11]. The core technology is the Josephson Junction [12] consisting of two

¹We have chosen the term *quantum primacy* rather than *quantum supremacy* originally defined in [4]. Both terms are used in the literature to refer to the same idea. However, there has been some controversy and debate about this naming convention in the literature and popular science press [5], [6], [7]. *Quantum advantage* is usually reserved for quantum computers merely edging out conventional computers on tasks of practical interest. This idea is slightly different from the original idea of quantum supremacy / primacy where the computational separation should be distinctively large.

²Our preference is to use the term ‘conventional’ computer rather than ‘classical’ computer since the production and fabrication of devices at the nanometer scale requires domain knowledge implicitly based on quantum mechanics even if quantum coherence is not used in the operation of the resulting device.

superconducting metals separated by an insulator. The superconducting state [13] requires low temperatures provided by dilution refrigerators. There are numerous ways to combine inductors, capacitors, and Josephson junctions into varying circuit designs [14] and functionalities [11]. The transmon design [15] (a circuit consisting of a Josephson junction in parallel with a capacitor) is widely used as part of different qubit architectures pursued at companies such as Amazon, D-Wave Systems, Rigetti Computing, Google, and IBM.

Now, superconducting quantum computing devices with over 50 interacting qubits are becoming increasingly available [16], [17], [18]. The discussion of the flagship quantum primacy experiments conducted with these devices is collected in Section IV. Presently, we continue with atom-based quantum computing platforms.

Atomic physics gave the earliest experimental evidence of quantum theory; thus, the use of atoms as quantum information carriers is not so surprising. Much more surprising, however, is the level of controllability that has been demonstrated to date [19], [20]. First, we consider ion trap quantum computing then neutral atom quantum computing.

Ion trap quantum computing manipulates microscopic crystals composed of a handful of ionized atoms [21]. Each ion is addressed optically to isolate and control individual qubits of quantum information. Vibrations within the micro-crystal allow for controllable interactions between the ions [21], [22], [23]. Although commercially available ion trap quantum computers are smaller than superconducting devices, the control over the quantum state is typically greater, allowing for experimental demonstration of quantum simulations [22] and quantum error correction [23].

Neutral atoms are trapped and manipulated using optics. Optical tweezers use laser light to trap and arrange the atoms into a 2D array [24]. The atom-atom interactions are mediated using the Rydberg excitations of electrons. When an electron is excited into a Rydberg state, the electron is far from the atomic nucleus but nonetheless remains bound to the atom. An apt analogy of the excited electron in a Rydberg state is that of a very distant but still gravitationally bound satellite of earth e.g. Halley's comet. The large extent of the electron in the Rydberg state can be used to mediate the multi-atom interactions. Early experimental setups only allowed model condensed matter systems to be tested [25] but recent progress has resulted in far more general quantum computing devices [26], [27].

In Nuclear Magnetic Resonance (NMR), the quantum spin of nuclei in a strong applied magnetic field is accessed and manipulated with microwave pulses [28]. Although not widely pursued at the industrial level, NMR quantum computing has important conceptual significance. Many of the key quantum computing concepts were first uncovered in this context and many of the first experimental implementations were done using NMR quantum computing [29], [28].

While light sources are important for manipulating superconductors and controlling atomic systems, quantum properties of light can also be used for computational gain. Each

unit of light (a photon) has two orthogonal directions for its polarization which can be manipulated quantum mechanically. However, the photons must be emitted in a highly correlated state before manipulation [30]. This process of obtaining sufficiently correlated units of light is probabilistic, but can be heralded (i.e. a secondary flag is able to indicate a successful generation attempt). By multiplexing entanglement generation trials, companies such as PsiQ [31] and Xanadu [32] as well as national collaborations are attempting to create quantum computers with light as the primary information carrier. Photons are also called 'flying' qubits since they can be used to couple distant quantum devices for computational or cryptographic purposes. Unlike some other quantum information paradigms, cryogenic temperatures are not needed.

Lastly, it is worth mentioning defect-based quantum computing. Here, defects in a crystal, like diamond, can be used to store and manipulate quantum information. Nitrogen vacancy defects in diamond have seen the most development, but other defects like silicon in diamond are also being explored [33]. The implantation of these vacancies as well as isotopic purity are difficult to control and have prevented defect-based quantum computing devices from reliably storing large amounts of quantum information. To date, the primary technological significance of nitrogen vacancy centers has been in the area of quantum sensing applications [34].

Following this rapid survey of devices used in quantum technology, we now turn to the theory. We will strive to give the principles behind the quantum theory of these devices and of quantum computing more broadly. We will begin with the probability-first approach to quantum mechanics [35]. Our introduction to quantum technology is aimed to allow engaged and knowledgeable readers opportunities to incorporate their own expertise. Similarly, our approach to quantum theory will allow readers to extend their analytical understanding of probability theory to quantum theory. This is a novel approach as compared to standard introductions e.g. [36].

II. QUANTUM THEORY

For our introduction to quantum theory, we will begin with the notion of a probability density vector \vec{p} with components p_i that correspond to the probability of the i th event occurring upon measurement, see Fig. 1. Note that any concept of measurement compatible with probability will also be compatible with the notion of quantum states discussed below. Probability vectors must satisfy the following three properties to maintain a sane interpretation of probability [37]:

$$\text{Normalization: } \|\vec{p}\|_1 = \sum_i |p_i| = 1 \quad (1a)$$

$$\text{Real valued: } p_i \text{ is a real number} \quad (1b)$$

$$\text{Positive semi-definite: } p_i \geq 0 \quad (1c)$$

To derive possible stochastic transformations, these mathematical properties constrain what is possible. Pursing probability theory by considering these constraints without concern for their origins is the essence of a kinematic approach. We will take this approach to quantum after introducing the quantum constraints extending Eq. (1).

For quantum states, we can analogously define the quantum probability density matrix, $\hat{\rho}$, as the generalization of the probability density vector with the following properties:

$$\text{Normalization: } \|\hat{\rho}\|_1 = \text{Tr}|\hat{\rho}| = \sum_i |\hat{\rho}_{ii}| = 1 \quad (2a)$$

$$\text{Real-valued spectrum: } \hat{\rho} = \hat{\rho}^\dagger \quad (2b)$$

$$\text{Positive semi-definite: } \vec{x}^\dagger \hat{\rho} \vec{x} \geq 0 \text{ for all } \vec{x} \quad (2c)$$

Here, \dagger indicates the conjugate transpose whereby $(A^\dagger)_{pq} = A_{qp}^*$ and $(a + bi)^* = a - bi$ with $i^2 = -1$. These mathematical extensions ensure that the diagonal of the quantum state, no matter how the matrix is represented or measured, remains a valid probability density vector. The kinematics of the quantum states then follow directly from maintaining these constraints to preserve a proper interpretation of the quantum state [35].

The notion of measurement is inherited from one's taste in probability. For instance, when considering quantum theory applied to estimating outcomes of an experiment, a Bayesian approach may be most appropriate. But in the case of repeatedly measuring the state of a reproducible state, a frequentist interpretation may be more apt. In principle, the interpretation of measurements presents no more of a foundational concern to quantum theory than it does already in probability theory. The key difference is that the probability density matrix may appear differently depending on the set of states being used to represent (and to measure) the state.

For the smallest comparative example, consider two outcomes: Outcome 0 and Outcome 1. The probability of getting Outcome 0 is given by a real number between values zero and unity. This is represented by the red line in Fig. 1. The possible quantum states (according to the constraints above) are all possible vectors within the sphere. The projection of the quantum state onto the probability axis gives the probability of obtaining an outcome corresponding to the event labelled at the positive endpoint of the probability line.

In quantum theory, we can select the orientation of the measurement as any line of probability through the Bloch sphere. For probability theory, the only allowed change of the measurement orientation is a permutation of the events. This would correspond to considering the probability of realizing Outcome 1 instead of Outcome 0. Those fluent with matrix analysis [38], will recognize that permutations are a discrete subset of the continuous orthogonal group which is itself a subset of the unitary group of transformations.

As an example of changing the quantum measurement orientation, consider the green probability line along the X -axis in Fig. 1c has two Outcome + or Outcome -. This new line of probability corresponds to obtaining the Outcome +. For any qubit state, we can obtain the outcome probability using the projection of the state on the green axis. An example is depicted by the dot placed along the green axis.

It is useful to note that the rotation of the state (with a fixed measurement orientation) and the rotation of the measurement basis (with a fixed state) are effectively the same. In the language of vector kinematics, the two types of rotations

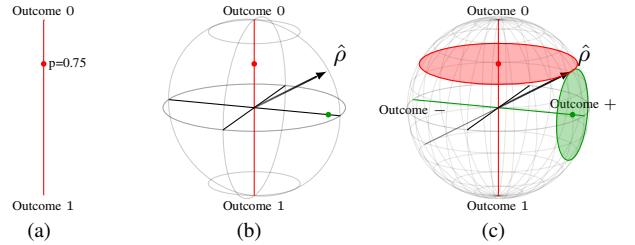


Fig. 1. In sub-figure (a), we visualize the p , the probability of obtaining Outcome 0, as a point along a line from zero to one. This picture is generalized as a Bloch sphere in sub-figure (b) used to visualize quantum probability distributions. Quantum probability distributions, $\hat{\rho}$, are vectors within the unit sphere. In sub-figure (c), the projection of the state $\hat{\rho}$ in two different measurement bases with the red axis corresponding to measurement in the Z basis. The two outcomes along the green axis are canonically labelled as Outcomes \pm and the project in green gives the probability of obtaining Outcome + in the X direction.

are termed active and passive rotations respectively [39]. In quantum physics, these two types of rotations are termed the Schrödinger and Heisenberg pictures [40].

For a concrete example of changing the measurement basis, consider the Hadamard transformation as given in Table I. This operation rotates the Z -axis of the sphere by 90° to obtain states in the X - Y plane. The probability of obtaining outcome zero or one is half since all states in the X - Y plane project to the midpoint of the Z probability axis. The Hadamard transform plays a special role in many quantum algorithms for its use in preparing quantum states with high coherence.

From linear algebra [41], we know that there always exists a basis for measurement in which the quantum probability density matrix is diagonal and, consequently, is just an ordinary probability density vector. This basis is called the eigenbasis and such a basis always exists for probability density matrices. The eigenbasis is the measurement basis in which coherences are not necessary to describe the state. The basis states for this measurement are called eigenstates and the probabilities are called eigenvalues. In this measurement basis, there is no essential difference between a probability density vector and a probability density matrix other than the organization of the probability values into a vector or a diagonal matrix.

The off-diagonal elements of the quantum probability density matrices are called coherences and are allowed to take values from the complex field of numbers so long as the state remains consistent with the kinematic constraints of Eq. (2). Quantum theory differs from ordinary probability theory through the use of these off-diagonal degrees of freedom (called coherences) that are generated after certain quantum operations. These coherences disappear as the system's behavior becomes less quantum and more akin to standard probability theory. The process of losing coherence is defined as decoherence and is the major stumbling block for technological realizations of quantum devices. Decoherence in experimental quantum devices often appears due to device control errors and unwanted interactions with the environment.

The discussion of quantum theory has thus far focused only

$$\hat{\rho}_{Bell} = \hat{\rho}_{AB} = \frac{1}{2} \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix} \neq \hat{\rho}_A \otimes \hat{\rho}_B$$

If Alice obtains...	... Bob will never obtain
0	1
1	0
+	—
—	+

Fig. 2. There are two bases that Alice and Bob may measure: the 0-1 basis (on the Z -axis of probability shown in red) and the \pm basis (measured along the X -axis of the Bloch sphere shown in green). After Alice has chosen her basis and obtained the outcome listed on the left column, Bob cannot obtain the outcome listed on the right no matter which basis he chooses to measure in.

on the kinematic differences between quantum and probability theory. However, the critical difference between quantum and probability theory is how correlations are treated. In ordinary probability, if we consider a probability density vector associated with two random variables, say \vec{p}_{AB} , then we can say the variables are correlated if $\vec{p}_{AB} \neq \vec{p}_A \times \vec{p}_B$. That is, if we cannot write the joint density as the product of two different densities. In quantum mechanics, the same idea of correlated states leads to definitions of the type: $\hat{\rho}_{AB} \neq \hat{\rho}_A \otimes \hat{\rho}_B$ where we have used the appropriate generalization of the Cartesian product for quantum probability density matrices known as the Kronecker tensor product.

Entanglement is a key distinguishing feature of quantum computation. Ordinary probability allows for correlated variables, but the use of coherence allows for the two systems A and B to be more strongly correlated than is otherwise possible with two probabilistic systems. This excess correlation available to quantum systems is called entanglement [42], [43], [44]. An important example of an entangled state is the Bell state defined and discussed further in Fig. 2.

Without high entanglement, quantum computers are not more powerful than conventional computation [45] but it should be noted that entanglement is necessary but not sufficient for quantum computation (as per the matchgate and Clifford gate sets discussed in Section III). The insights from the quantum information community have led to great advances in conventional computing through the study of matrix product states and more general tensor network states [44], [43], [46]. The key insight is that the compression and efficient manipulation of the quantum state becomes possible on conventional computers when the entanglement remains low [44], [45], [43]. By contrast, highly entangled states can be used to demonstrate violations to the local realism hypothesis [47] using Bell's inequality [48], [49]. Because of the violation of local realism, these highly-entangled states are also useful for communication security [50].

A. Quantum wave functions

In this brief introduction to quantum theory, we have not yet mentioned the Schrödinger equation nor wave functions. Instead, we have focused on quantum probability density matrices and their kinematics. If we followed historical and standard approaches, we would have begun with wave functions, linear algebra, and differential equations. The notion of quantization might have arisen as a consequence of solving a linear second-order differential equation. Note that superposition and quantization feature in sound and heat waves, in classical electromagnetic theory, and many connected mathematical situations that are not related to quantum mechanics. In order to keep the discussion focused on the essentials of quantum theory, we instead mention wave functions as a special case of the quantum probability density matrices, so called pure states.

Wave functions³ are quantum probability densities corresponding to a quantum state with deterministic outcome in at least one measurement direction. That is, there is a measurement direction where the probabilistic outcomes are describable as $\vec{p}_1 = (1, 0, 0, 0, \dots, 0)^T$ corresponding to vectors of unit length in Fig. 1. In this case, the quantum probability density matrix is called pure and satisfies the projector condition: $\hat{\rho}^2 = \hat{\rho}$. In the case of a pure quantum state, the quantum density matrix simplifies and can be characterized by a single vector $\vec{\psi}$, called a wave function⁴. Note that in every measurement basis except the eigenbasis of the probability density matrix there will be coherences. As these coherences are lost with respect to any other basis, the state will no longer be describable by a single vector $\vec{\psi}$ and the correct description is given by the quantum probability density matrix formalism.

In the next section, the choice between the wave function vectors and the more general quantum probability density matrices is not forced. This is because most quantum computing gate operations are considered to be noiseless unitary transformations. In the case of a single qubit, the unitary transformations correspond to rotations of the Bloch sphere vector without changing its length. More technically, the unitary matrices do not change the eigenvalues of the quantum probability density matrix. Consequently, if the quantum state is pure and there is only one non-zero eigenvalue, unitary operations will not decrease (nor increase) the purity.

B. Quantum states spaces compared to probability event spaces

In deterministic computing, the probabilities are enforced to be either zero or one; however, to extend this notion to quantum information theory, it is useful to think of the bit as taking values with some probability p . By beginning with this outlook, we can avoid misleading arguments about the size of

³Here we use standard terminology although we only consider wave functions as vectors, i.e. $\vec{\psi}$, rather than functions, i.e. $\psi(x)$.

⁴Note that when there is a basis where \vec{p}_1 appears, then the quantum probability density matrix has only one eigenvector corresponding to the lone non-zero eigenvalue. This allows us to write $\hat{\rho} = \vec{\psi}\vec{\psi}^\dagger$ with $\vec{\psi}$ defining the wave function.

the quantum state space. Regardless of whether we have n bits or n qubits, we have 2^n possible basis states. In the case of n bits, we can describe the full state with a probability vector containing 2^n real-valued entries. For example, when $n = 3$

$$\vec{p} = \begin{pmatrix} \text{prob}(X = 000) \\ \text{prob}(X = 001) \\ \text{prob}(X = 010) \\ \text{prob}(X = 011) \\ \text{prob}(X = 100) \\ \text{prob}(X = 101) \\ \text{prob}(X = 110) \\ \text{prob}(X = 111) \end{pmatrix} \quad (3)$$

Due to the normalization condition, at most $2^n - 1$ real values are needed. The quantum system, by contrast, cannot be described by a probability vector alone. Nonetheless, upon measurement, for n qubits there will be only 2^n possible outcomes. Thus, the distinction between n qubits and n bits is deeper than a simple counting argument.

The key takeaway from this section on theory is that quantum coherence and quantum entanglement are the essential resources that quantum devices exploit for computational gains. One frequently encounters misleading claims about superposition or about the size of the quantum state space as reasons for quantum computational gain. By beginning with and comparing against probability theory rather than classical mechanics, we hope it is clear that superposition and counting arguments do not account for the power of quantum computation. Moreover, this probability-first approach has allowed us to sidestep philosophical discussion of measurement and proceed to the essential difference between probability and quantum theory. Armed with these basic ideas of quantum theory, we now turn toward computation beginning with the models of quantum computing in the next section.

III. MODELS OF QUANTUM COMPUTING

In this section, we will consider the models of computation used to design quantum algorithms. There are many excellent introductions to the quantum circuit model e.g. [36], [51], so we will be satisfied with a brief survey of the main ideas and key notions. This prepares us for a look at restricted quantum circuit models which are efficiently simulated without quantum resources. This connects to the central research questions of what is the difference between quantum and conventional computing. We then close this section with a mention of some alternative models of quantum computation that are computationally equivalent to the quantum circuit model.

Before delving into the circuit model, we will make two remarks to connect to the prior section. First, for one qubit, all quantum gates from the quantum circuit model can be visualized as transformations of the Bloch sphere. If the gates are performed perfectly then they correspond to rotations of the Bloch sphere. In more general transformations, the Bloch sphere may be deformed such that the length of the vector representing \hat{p} can be less than unit length. (Recall, from

TABLE I
SOME STANDARD QUANTUM GATES AND THEIR REPRESENTATIONS.

Operator	Gate	Matrix
Hadamard (H)		$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
Phase (S)		$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$
Controlled Not CNOT		$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$
$\pi/8$ (T)		$\begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$
Toffoli gate		$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$
Pauli <i>X</i>		$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
Pauli <i>Y</i>		$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$
Pauli <i>Z</i>		$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
Rotation about \vec{n}		$\exp(i\theta\vec{n} \cdot (X, Y, Z)/2)$

Fig. 1, that a vector of length zero would correspond to the mid-point of the probability axis i.e. $p = 0.5$.)

Second, we can consider moving away from the general picture of a gate or series of gates, U , acting on a quantum probability density matrix: $\mathcal{E}_U(\hat{p}) = \hat{p}_{\text{output}}$. If the circuit U is done without any errors, then we can write, $\mathcal{E}_U(\hat{p}) = U\hat{p}U^\dagger$. Further, if the probability density matrix is pure $\hat{p}^2 = \hat{p}$, then we may write $\hat{p} = \vec{\psi}\vec{\psi}^\dagger$. Combining these two statements, we have $\mathcal{E}_U(\vec{\psi}\vec{\psi}^\dagger) = U\vec{\psi}(U\vec{\psi})^\dagger$ allowing the quantum dynamics to be describable as $U\vec{\psi} = \vec{\psi}_{\text{output}}$. Thus, in the case that both the gates are perfectly implemented and the probability density matrix is pure, we can use the wave function picture of $U\vec{\psi}$ found in standard introductions [36], [51]. This distinction will not be relevant for the remainder of the review.

For conventional computing in the circuit models, the NAND gate is universal for computation. This means that every other circuit has an equivalent construction using exclusively NAND gates. The NAND gate takes two inputs and returns a single output 0 if and only if both inputs are 1. Because two bits are used for input and only one bit for output, this gate is irreversible and cannot be used as a quantum circuit element in a straightforward way. The Toffoli gate, as shown in Table I, is a reversible gate that is universal for conventional computation. Its existence as a quantum circuit element shows that all conventional computations can be converted to quantum computation. A few examples of quantum circuit elements are given in Table I.

A universal gate set is a fixed set of operations that when

composed together allow one to construct any arbitrary multi-qubit transformation. We note immediately that the number of gates need to construct an arbitrary unitary starting from a fixed set could potentially be very large. Nonetheless, given single-qubit rotations, almost any two-qubit gates will suffice [52]. There are a few special instances where the two-qubit gate is insufficient: for example, if the two-qubit gate is actually composed of two independent single-qubit gates. However, if an operation or matrix is chosen at random then almost surely it is sufficient for universal quantum computation [53], [52]. Thus, a universal gate set is achievable through almost any two-particle interaction; hence, the wide variety of physical qubits.

There are two gate sets of special interest because, not only do they fail to span the full set of transformations, they are also tractable with conventional computing. The first is the Clifford gate set[54]. The H, S, and CNOT gates are the generators of the Clifford group. A circuit composed of only the Clifford gates can be computed using conventional computation. Note that the T gate listed in the Table of gates allows for the Clifford set to realize universal quantum and some quantum algorithmic costs estimates are given by the T gate count of the implementation. The second specialized gate set is the matchgate [55] (or free-fermion [56] gate set) which can also be efficiently simulated without quantum resources. First, we begin with the Clifford gate.

In the case of the Clifford gates (H, S, CNOT given in Table I), a family of operators (the Pauli group) $\mathcal{P} = \{P_1, P_2, P_3, P_4, P_5, \dots\}$ is preserved such that action of Clifford gates will only transform each element P_j into another element P_k but not into linear combinations of elements as in $0.4P_k + 0.6P_{k'}$. This gate set was studied in the theory of quantum error correction [54], [57], [58], [36] leading to the development of the stabilizer formalism⁵ widely employed in quantum information science. Note that many of the early ideas for quantum algorithms that relied exclusively on superposition are simulatable with the Clifford formalism e.g. Deutsch's algorithm [59].

A second set of gates that are also tractable with conventional computing is the matchgate set [55]. This set is also known as free-fermion gates [56]. Both the perfect matching problem⁶ for *planar* graphs [60] and the physical

⁵The stabilizer formalism extends families of operators, like the Pauli group \mathcal{P} , to a corresponding set of vectors $\{\vec{\psi}_k\}_{k=1}^n$ that are ‘stabilized’ by that particular family of operators. Here stabilized means that the action of any operator from the family does not change the corresponding set of vectors $\{\vec{\psi}_j\}$ e.g. in the case only one vector is stabilized: $P_m \vec{\psi}_0 = \vec{\psi}_0$. If an unwanted error occurs then the resulting vector will leave the stabilized subspace and in many instances, this can be both detected and corrected [54], [57].

⁶The perfect matching problem for a graph with given vertices and edges requires selecting a set of edges where every vertex has only one neighbor. The selected edge perfectly matches all vertices into pairs.

description of *non-interacting* fermionic particles⁷ are deeply connected to the computational and mathematical properties of determinants.⁸ This connection enables circuits composed of matchgates to be efficiently computed without the help of quantum resources [56], [55].

So far, we have been discussing quantum operations as a discrete set of gates that are applied in a prescribed sequence. However, the quantum circuit model is not the only way to describe quantum computation just as there are equivalently powerful ways to control a conventional computer e.g. Fortran or C++. Similar to the difference between conventional programming languages⁹, some quantum computational models align with certain quantum devices better than others.

Some important alternatives to the quantum circuit model include quantum Turing machines [63], adiabatic quantum computation [64], [65], quantum walk models [66], and dissipative quantum computing [67]. Additionally, a wide variety of layered circuit designs have been shown to be universal for quantum computation such as quantum approximate optimization [68] and variational quantum eigensolver [69]. In Section V, we collect these observations using the quantum signal processing framework [70].

IV. QUANTUM PRIMACY EXPERIMENTS

Quantum primacy refers to the situation where a quantum device has performed a computation that cannot be done in any other way. Note that the computational task may or may not have practical significance but is chosen merely to show the separation between quantum and conventional computing devices.

The two most widely discussed approaches to quantum primacy [71], [72] are the BosonSampling [61] and Random Circuit Sampling [73] problems. In the first of these problems, Random Circuit Sampling, considers the sampling from the output distribution of randomly generated quantum

⁷Fermion are identical particles that carry the antisymmetric one-dimensional representation of the symmetric group of permutations. Since this representation is one-dimensional, the action of permutations does not change the quantum probability density matrix. The other one-dimensional representation of the symmetric group irrespective of the number of elements being permuted is the completely symmetric representation which corresponds to bosons discussed further in Section IV.

⁸For the present discussion, let us recall [41] that a determinant is characterized by

$$\det(A) = \sum_{\pi \in S_n} \text{sgn}(\pi) \prod_i^n A_{i,\pi(i)} \quad (4)$$

where A is an $n \times n$ matrix and S_n is the set of permutations of n objects. Here $\text{sgn}(\pi)$ is ± 1 depending if the permutation π is composed of an even e.g. $P_{12}P_{23}P_{12} = P_{13}$ number of transpositions. This summation over n factorial factors to obtain $\det(A)$ is equal to the product of the eigenvalues of A . The eigenvalue computation is much faster than summing over the formula in Eq. (4). The same combinatoric function without the $\text{sgn}(\pi)$ is called the permanent: $\text{perm}(A) = \sum_{\pi \in S_n} \prod_i^n A_{i,\pi(i)}$. This corresponds to non-interacting particles called bosons [61].

⁹One may argue the identity of programming languages mostly lies in their syntaxes and programming paradigms. However, in the current context, two programming paradigms are called equivalent if they both simulate the same class of functions up to a polynomial-time overhead due to the conversion [62].

circuits [73]. It is important to note that these quantum primacy experiments are based on sampling arguments which further underline the tight correspondence between quantum and probability theory discussed earlier. The second problem, BosonSampling, a few photons are sent into a randomly selected linear-optics setup and the computational task is to sample from the output distribution of photons exiting the optical network [61].

After the attention received by the first experimental result claiming quantum primacy [16], there has been a fascinating string of results both experimentally and theoretically leaving the exact boundary of primacy still an open area of active research. The first claim of quantum primacy was accomplished using a 53 transmon qubit device executing and sampling around 1,500 random gates as part of a circuit with 20 layers deep [16]. However, this was for brute force approaches. Since then, larger experimental implementations of Random Circuit Sampling with more qubits (56 qubits and with 60 qubits) with a circuit depth of 24 layers [17], [18].

Initially, it was claimed [16] that conventional computers would take 10,000 years to accomplish the same task. This assumed a brute force simulation however this has been vastly improved using tensor networks methods based on the analysis of states with low entanglement [44], [43]. The application of heavy computational resources and clever entanglement truncation schemes have led to much improved conventional computational results largely closing the computational gap [74], [75], [46], [76], [77]. Regardless of the classical computational progress, the cost of performing tensor network simulations scales exponentially with the number of qubits so these approaches can be defeated by going to larger experimental qubit devices. Note that even if the original quantum primacy circuits are simulated without quantum resources, performing increasingly larger Random Circuit Sampling instances should increase the difficulty beyond conventional computing since tensor networks scaling exponentially with the system size. Theoretically, this idea is supported [71], [72], [78] based on complexity conjectures. Unfortunately, the standard statistical benchmark used to certify correct RCS, the linear cross entropy, has major deficiencies [79].

The linear cross entropy benchmark was chosen as an easy to estimate proxy for the fidelity of the quantum state prepared. The fidelity, the total variation distance and the cross entropy require exponential samples to estimate while the linear cross entropy does not. However, there are statistical vulnerabilities when the linear cross entropy is used instead of other metrics that require exponential more sampling. More startling is the asymptotic scaling of the linear cross entropy as compared with standard measures like the fidelity. For conventional algorithms attempting to hack or otherwise spoof the linear cross entropy, it can be shown that their performance actually improves with system size. This behavior is drastically different from the scaling of the fidelity between N approximate

states and N ideal states asymptotically.¹⁰ Thus, there is need for a new route to validating the results of Random Circuit Sampling.

The second route to quantum primacy also under active debate uses the BosonSampling problem [61]. The BosonSampling problem relies on the hardness of approximating the permanent of a matrix and its deep connection to the nature of indistinguishable bosons. BosonSampling can be implemented with linear optics but requires less than full-fledged universal photonic quantum computing [30]. The first claims of quantum primacy using a quantum optics setup appeared in 2020 [80] and have been since been improved [81]. Similar to the discussion ensuing the Random Circuit Sampling primacy experiment, improved conventional computational methods are claimed to simulate the early experimental BosonSampling primacy results [82]. However, the mid-2022 announcement from Xanadu Computing provides a new quantum primacy experimental claim [?].

The experimental and theoretical progress in understanding the boundaries between quantum and classical computing is an active and growing area of research. This race between quantum and conventional technologies an interesting area to follow in 2022. Next, we turn our attention toward practical applications and quantum algorithms.

V. QUANTUM COMPUTING ALGORITHMS AND APPLICATIONS

Applications of quantum computing are as diverse as the fields necessary to create quantum information processing technology. In recent years, the excitement has attracted further talent and investment sparking a rapid growth in the quantum algorithm development [83]. There now exists a zoo of quantum algorithms [84] and, in this section, we hope to provide a taxonomy with for understanding and traversing the literature. We do so by dividing our quantum algorithm classification into two overlapping groups: the first wave quantum algorithms based on formal methods and the second wave quantum algorithms largely based on optimization methods. We will connect these two groupings using the quantum singular value transformation framework which broadly and elegantly combines decades of theoretical quantum algorithms research.

Initial ideas about quantum computation were generated with heavy input from the mathematics and computer science community, where the questions of noise were considered only formally and shown to be theoretically surmountable. Therefore, the first wave of quantum algorithms assumed noiseless quantum device operation (or otherwise fully quantum error corrected systems). The availability of noisy intermediate-scale quantum (NISQ) [85] computers spurred the second wave of quantum algorithms and quantum tasks that more easily take into account noise and advances in algorithm design on conventional computers. We begin with the first wave, followed by the second wave.

¹⁰If an approximate state has overlap s with the ideal state then, the combined state of N non-interacting copies have overlap of s^N which is asymptotically vanishing.

A. First wave quantum algorithms for ideal quantum devices

The initial burst of quantum computing algorithms was developed with an ideal quantum computer in mind. Quantum Turing models, quantum gate model, and query-based oracle models were studied largely without noise. The development of quantum error correction [57], [58] was pursued alongside the first wave quantum algorithms with sufficient theoretical success that algorithm designers felt they could largely ignore noise and details of the physical architecture. The canonical reference for this wave of quantum algorithm development is [36] which despite being over 20 years old still stands as a solid reference for the theoretical background of quantum computing and quantum information. The recent review article Ref. [86], also provides a canonical introduction to quantum computing with an up to date perspective on the field.

The two foundational results culminating from these effort are the Grover search and the quantum Fourier transform subroutines. We begin with Grover search.

Grover search is a quantum algorithm for finding a marked item in an unsorted database of size N . This task ordinarily takes a number of trials that is proportional to the size of the database. There is no particular shortcut other than testing each item in the database for a hit. An apt example is searching for a specific tool in an unsorted shed. Each item can be evaluated as correct or incorrect by the mechanic e.g. who sits far away from the shed. With Grover's quantum algorithm, the marked item can be found in a time that is proportional to the square root of the database size.

Grover's search was announced in 1996 [87] and it was one of the few quantum algorithms that had proof of optimality for the task it solved [88]. This means for problems belonging to the complexity class NP, quantum algorithms can, at most, achieve quadratic speedup [88]. Informally, this means if the best known computational method for solving a sufficiently generic problem is just the brute-force testing of every possible solution, then optimal use of quantum resources will achieve only a quadratic improvement over the conventional brute-force algorithm.

To understand the major themes of quantum algorithm development, it is useful to delve deeper into the theory behind the Grover search as it serves as the backbone of most other quadratic speed ups in quantum algorithm development. The key idea of the Grover algorithm is the use of a product of reflections where¹¹ $R^2 = 1$. Reflections are built as $R = (2\Pi - 1)$ whenever Π is a projector $\Pi = \Pi^2$.

The Grover operator, $W = R_A R_x$ is a product of two reflection operators that after approximately \sqrt{N} iterations, transform the initial state $\hat{\rho}_0$ into a marked state $\hat{\rho}_x$ with high probability. The first of the two reflection operators, $R_A = 2\hat{\rho}_s - I$, performs a reflection about the pure state generated by applying the Hadamard transformation on all qubits initialized in the Outcome 0 state of the Bloch sphere. The resulting quantum probability density matrix $\hat{\rho}_s$ is such that all matrix

¹¹A reflection matrix is an unitary (or orthogonal) matrix with determinant equal to -1.

elements including populations and coherences are equal to $1/N$. Second, another reflection operator, $R_x = 2\hat{\rho}_x - I$, provides phase oracle access to the target pure state, $\hat{\rho}_x^2 = \hat{\rho}_x$. See Fig. 3a. We will return to this analysis in Section V-C but for now we continue with the quantum Fourier transform and its applications.

Quantum factoring and quantum simulation are the two most important applications of quantum computing. Both rely heavily on the quantum Fourier transform. The quantum Fourier transform (QFT) algorithmic sub-routine improves exponentially over the conventional computing Fourier transform implementations. The exponential separation between these two techniques is at the heart of many of the most exciting applications of quantum algorithms like factoring [89], phase estimation [90] and the matrix inversion algorithm [91].

The difficulty of factoring large prime numbers provides modern communication security [92]. The commercial and cryptographic importance of Shor's quantum factoring algorithm sparked much of the excitement and attention of quantum computing in the first wave of quantum algorithm development. For factoring composite numbers, the quantum algorithm [89] requires solving for the periodicity of $f(x) = a^x \bmod N$ with N the composite number to be factored, a is a randomly selected integer, and $x \bmod y$ equal to the remainder resulting from division of x by y .

A similar idea extends the applications of quantum Fourier transform to finding the periodicity of a quantum time-evolution. This is leveraged in quantum simulation tasks [93], [94] resulting in the phase estimation technique [90]. The phase estimation algorithm allows the eigenvalues of a Hamiltonian to be probed and has found many applications in quantum chemistry and the simulation of physics [95], [96], [97], [98].

B. Second wave algorithms for noisy intermediate-scale quantum devices

This second wave of quantum algorithms attempts to circumvent the use of error correction by relaxing the strict theoretical impositions of quantum computer science [63] and turns towards heuristic optimization procedures. Many of these approaches are designed to take into account the hardware strengths and limitations as well as leverage advances in conventional optimization techniques.

The results of quantum computational complexity theory usually focus on worst case analysis [63], [99], [100] leaving open the possibility that quantum resources may make progress using heuristic optimization methods [101], [102]. The resulting wave of approaches typically use feedback between a quantum device and a conventional computer. Error mitigation strategies can be included as part of the quantum optimization loop. In the literature, this family of methods include variational quantum algorithms, variational quantum eigensolvers (especially in the context of quantum simulation problems), quantum approximate optimization

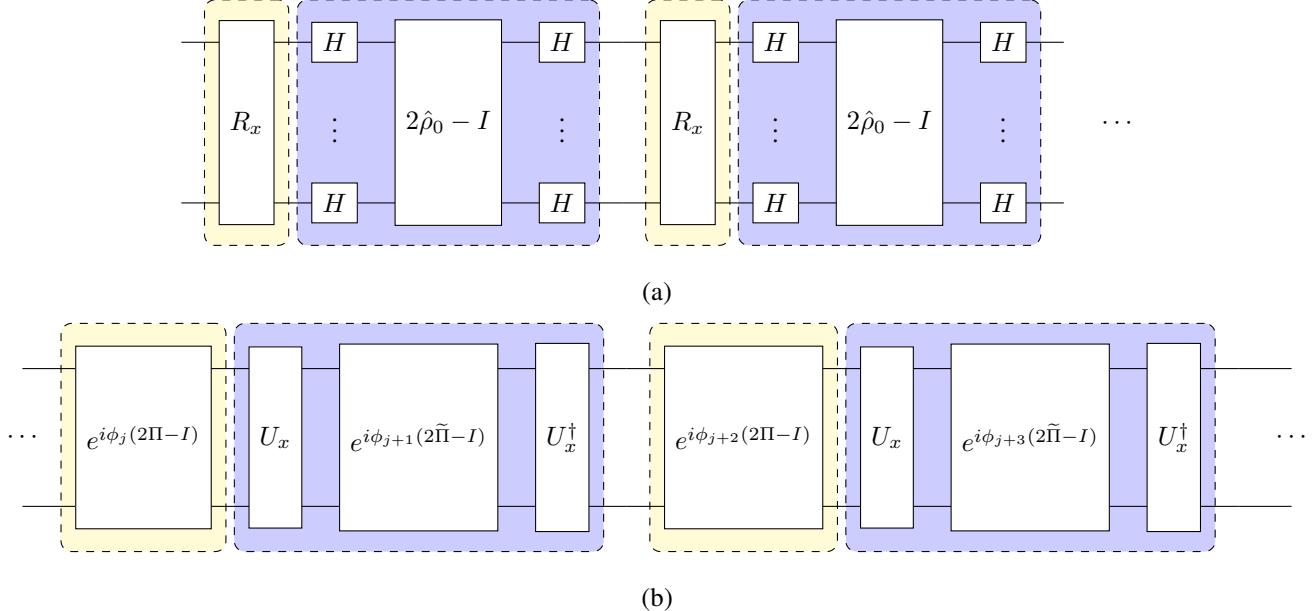


Fig. 3. The Grover search and the quantum singular value transform share conceptual similarities. The Grover search circuit consisting of two reflections shown in Subfig. (a). The first reflection R_x is constructed using the oracle deciding if a guess is correct or incorrect. The second reflection $2\hat{\rho}_0 - I$ is about the quantum state $\hat{\rho}_0$ where all qubits are in the $+Z$ state of the Bloch sphere. The pre- and post-fixed Hadamard gates transform the each qubit of the state into the $+X$ state. This yields the uniform superposition. In Subfig. (b), the generalized circuit for the quantum singular value transformation still consists of a product of two reflections. However, these reflection operations are performed with step-dependent control parameters ϕ_j about the subspaces identified by either projector Π or $\tilde{\Pi}$. (Note, since $R^2 = (2\Pi - I)^2 = I$, we have $\exp(i\phi R) = \cos(\phi)I + i\sin(\phi)R$.) The analysis of the possible functions of U_x reachable after d iterations results in the quantum singular value transformation framework [70].

algorithm, hybrid quantum-classical algorithms¹² and other anasatz methods. Most of these approaches utilize a conventional computer to perform an optimization procedure using information extracted from the quantum device, usually in an iterative fashion. These quantum optimization methods have been applied to diverse areas such as portfolio optimization [103] and quantum machine learning [104]. Many of the flagship experiments in quantum optimization were used to find the lowest energy state of the quantum Hamiltonians found in quantum magnets, molecular compounds and other physical simulations contexts, quantum magnets, and other quantum simulation processes [95], [105], [106], [107], [98], [96].

Wide interest in quantum computation that is intrinsically robust against errors began with the introduction of the adiabatic quantum computing model [64]. In this model quantum instead of thermal fluctuations are used to perform optimization. The first quantum computing company founded, D-Wave Systems, began with this vision in mind. However, the design of the D-Wave devices only allows optimization of

Hamiltonian problems of the form

$$H(\vec{s}) = \sum_{edges: \{ij\}} J_{ij} s_i s_j + \sum_{vertices} h_i s_i \quad (5)$$

Here H , with instances defined by $\{J_{ij}, h_i\}$, is a function of the binary variables $s_i = \pm 1$ e.g. $\vec{s} = (+1, -1, +1\dots)$. The goal is to find the configuration \vec{s}_* that minimizes $H(\vec{s})$. By varying the parameterization of H , we can encode any other constraint satisfaction problems from the computational complexity class NP [60]. By restricting s_i to be ± 1 , the D-Wave devices are only able to solve non-quantum problems with utilizing quantum resources [108].

One important caveat worth highlighting is that optimization problems can be arbitrarily hard. Thus, many of the claims about quantum optimization must be made heuristically. Because other optimization problems can be mapped to Eq. (5), solving this problem for all instances of $\{J_{ij}, h_i\}$ is not likely to be possible with a conventional computer [62]. As noted above one can, at best, achieve a quadratic speed up on this problem. Moreover, when the Hamiltonian in Eq. (5) is generalized to a quantum Hamiltonian where the inputs and the constraints can vary with their quantum degrees of freedom e.g. coherences, the problem only becomes more difficult. The full analysis of worst-case optimizations problems of lowest energy states of quantum Hamiltonians resulted in the theory of Quantum Merlin-Arthur problems [99], [100]. The takeaway from the results of computer science is that: even

¹²These methods highly overlap and often have the same computational power and expressivity. The term *hybrid quantum-classical algorithms* is not a useful way to describe the second wave algorithms because all quantum devices will simultaneously utilize conventional computing to control the quantum device or to dynamically correct errors.

with the promise that quantum resources can verify correct solutions efficiently, generating a targeted specific quantum state is not efficiently doable with or without quantum resources. In other words, just as quantum resources can help solve problems, they can also lead to increased difficulty.

So far, we have classified quantum computing algorithms into two branches: those designed for idealized abstractions of quantum computers and the heuristic methods inspired by the quantum hardware currently available. These are not distinct classifications but rather conceptual guidance for navigating the growing literature of quantum algorithms and their applications. Next, we examine the framework of the quantum singular value transformation. The purpose of including this specific theoretical framework is: (1) it captures the results and ideas from both types of quantum algorithms and (2) the technical underpinning allows it to subsume many known theoretical results bounding the performance of various quantum algorithms.

C. The quantum singular value transformation framework

The framework of quantum singular value transformations resulted from vast generalization of known techniques for analyzing quantum algorithms. The key idea is that a generalized Grover search protocol allows for nearly arbitrary polynomial transformation of a given matrix A_x embedded into the upper left block of a unitary matrix. Polynomial approximations to $\exp(A_x)$ and A_x^{-1} allows this generalization to subsume not only Grover search but also quantum Fourier transforms, matrix inversion quantum algorithms, quantum adiabatic optimization and many other. Here, we try to give a clear overview of the structure of the quantum singular value transformation and we refer the reader to [109], [70] and references therein for more precise notions.

The core insight begins with Grover search. There the quadratic speed up over conventional algorithms is achieved using a product of two reflections as mentioned above. The product of two reflection operators in the case of Grover search results in a two-dimensional subspace where the analysis of the quantum algorithm proceeds. This two-dimensional subspace exists no matter how large the database is and allows one to consider the algorithm using an effective two-dimensional space i.e. a qubit. This is the essence of the qubitization idea [110], [111].

The first step in the generalization of Grover's search is to allow reflections about more general subspaces than those generated by one dimensional projectors e.g. pure quantum probability density matrices. This insight was first used to effect in the quantization of pairs of Markov chains [111] where the product of reflections was decomposed into one- and two-dimensional subspaces.

The second important realization stems from the mathematical characterization of all possible functions achievable using NMR pulse sequences on a single spin. In NMR, the effective time evolution can be approximated as an alternating sequence of fixed free evolution followed by an experimentally controllable pulse. For the one qubit situation, this can be

described as a fixed rotation about the Z -axis of the Bloch sphere, $U_Z(x)$, followed by a parameterized rotation about the X -axis $U_X(\phi_j)$. Mathematically, this gives:

$$W(\vec{\phi}) = U_Z(x)U_X(\phi_1)U_Z(x)U_X(\phi_2)\dots U_Z(x)U_X(\phi_d) \quad (6)$$

The full analytic description of the qubit gates accessible by control parameters $\vec{\phi}$ to after d alternations is the second major ingredient of the quantum singular value transformation.

As may be expected, the resulting final gate is describable using polynomials of degree d of the variable x .

$$W(\vec{\phi}) = \begin{pmatrix} \text{poly}^{(00)}(x) & \text{poly}^{(01)}(x) \\ \text{poly}^{(10)}(x) & \text{poly}^{(11)}(x) \end{pmatrix} \quad (7)$$

Here each entry is a polynomial with known properties proven in a series of papers all summarised in reference [70]. There now exists numerical algorithms for selecting the parameters to reach target functions [109].

The culmination of these two lines of work resulted in the full analysis of the circuit shown in Fig. 3b. The theorems proven in the quantum singular value framework [70] allows one to extend the understanding of $W(\vec{\phi})$ to the transformation of subblocks of unitary matrices. Creating a unitary, U_x , with a specific matrix A_x present in the 00 sub-block is called a block encoding of matrix A_x .

$$U_x = \begin{pmatrix} A_x & \bullet \\ \bullet & \bullet \end{pmatrix}$$

The analysis needed to prove properties of the polynomials in Eq. (7) can be extended to prove that the output of the circuit in Fig. 3b will give

$$W(\vec{\phi}) = \begin{pmatrix} \text{poly}^{(00)}(A_x) & \bullet \\ \bullet & \bullet \end{pmatrix} \quad (8)$$

The quantum singular value transform In Eq. (8), the action of the polynomial on matrix A_x is defined by its action on the singular values of A_x giving the framework its name. The singular value decomposition of A_x is given as

$$A_x = \sum_k \sigma_k \vec{\psi}_k^L \vec{\psi}_k^R$$

with $\vec{\psi}_k^L$ and $\vec{\psi}_k^R$ the left and right singular vectors corresponding to singular value σ_k . Then, a polynomial of matrix A_x is given as

$$\text{poly}(A_x) = \sum_k \text{poly}(\sigma_k) \vec{\psi}_k^L \vec{\psi}_k^R.$$

The realization of polynomial transformations of the singular values of any block encoded matrix allows the framework to subsume most major quantum algorithms including their asymptotic cost analysis [109], [70].

Despite the proofs and analysis of the quantum singular value framework being mathematically involved, we have attempted to provide a useful overview. It should be noted that this section roughly follows the timeline of quantum algorithm development. We started with the first wave that began in the mid-1990's, followed by the crescendo of the second wave activity invigorated by the recent availability of actual quantum

devices, then we ended with a powerful framework that will likely seem more activity in the coming years.

VI. OUTLOOK

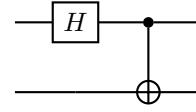
We began with technology, continued with quantum theory and its models of computation. Now with the overview of quantum algorithms and applications completed, we can add to a few final topics to complete our roadmap to the literature and status update on the field. While maps are useful, deeper appreciation occurs by experiencing the terrain. Therefore, we have ended with a few pointers to resources for accessing quantum devices.

Through the use of cloud computing, companies have already begun offering public access to quantum devices for academic or industrial research purposes. There are mainly two types of services. The first type is cloud services providing access to a single company's collection of quantum devices. The most widely known is the Qiskit cloud services offered by IBM Quantum [112]. The second type are multi-platform services that work as intermediaries to give users options to access quantum devices owned by multiple vendors. A key example of such a service is Amazon Braket [113] offered through Amazon Web Services.

In most cases, the cloud computing interfaces to quantum devices are implemented in Python to provide starting points for accessing working quantum devices. To give an idea of how quantum devices are manipulated, the circuit and corresponding Python snippets for the previously introduced Bell state (Fig. 2) given in Fig. 4.

Finally, we would like to highlight a few resources for learning the implementation of quantum algorithms. The IBM Qiskit textbook [51] provides an college-level introduction to quantum information with integrated programming exercises within the text. Similarly, the Codebook [114] by Xanadu provides an introductory course built around the Pennylane package providing differentiable programming of quantum computers. QBraid is an online platform for developing quantum software that sponsors quantum hackathons and offers introductory quantum tutorials [115]. There are many other introductory resources and coding platforms available through public, academic, and commercial channels.

Armed with the knowledge and perspectives of this review, we hope that the reader is prepared to appreciate and follow this growing area of interest. We have tried to provide a useful starting point for interested graduate students in science and engineering. There are many exciting things to look forward to in the area of quantum computing in 2022. This includes new quantum devices coming online, demonstration of definitive quantum primacy and of practical quantum advantage, stable quantum memory maintained by dynamical quantum error correction and the integration of multiple types of qubits into a single platform. We are looking forward to both watching and to participating in the evolution of quantum technology and this paper is our invitation to the reader.



```
from qiskit import QuantumCircuit, IBMQ
```

```
#Create the Bell-state circuit
# with 2 qubits and 2 bits
qc = QuantumCircuit(2,2)
qc.h(0)
qc.cx(0,1)
qc.measure_all()

IBMQ.load_account()
provider = IBMQ.get_provider(
    group='deployed')
backend = provider.backend.ibmq_manila
job = execute(qc, backend, shots=1000)
#####
from braket.circuits import Circuit
from braket.aws import AwsDevice

qc = Circuit().h(0).cnot(0,1)

#select a device or simulator
#Rigetti qubit device
device = AwsDevice(
    "arn:aws:braket::device/qpu/rigetti/Aspen-9")
# IonQ quantum device
device = AwsDevice(
    "arn:aws:braket::device/qpu/ionq/ionQdevice")

result = device.run(qc,shots=1000).result()
```

Fig. 4. The quantum circuit and sample Python code for IBM Qiskit and the Amazon Braket service to create the Bell state described in Fig 2. Note that although these commands are valid, these code snippets will not run without properly initializing an account with the correct permissions and running the proper commands to load those credentials.

VII. ACKNOWLEDGEMENTS

We would like to thank K. Setia, K. Stewart, T. Takeshita, and S. Gulania for useful discussions. Thanks to U. Vazirani for point out key quantum primacy references. This work was supported by the US NSF (PHYS-1820747, EPSCoR-1921199) and by the Office of Science, Office of Advanced Scientific Computing Research under programs Fundamental Algorithmic Research for Quantum Computing and Optimization, Verification, and Engineered Reliability of Quantum Computers project. This paper was partially supported by the “Quantum Chemistry for Quantum Computers” project sponsored by the DOE. JDW holds concurrent appointments at Dartmouth College and as an Amazon Visiting Academic. This paper describes work performed at Dartmouth College and is not associated with Amazon.

This paper is released under a Creative Commons Attribution 4.0 International License [116].

REFERENCES

- [1] “Overview on quantum initiatives worldwide - update mid 2021,” July 2021, <https://qureca.com/overview-on-quantum-initiatives-worldwide-update-mid-2021/> [Accessed: Jan 7, 2022]. 1
- [2] M. G. Raymer and C. Monroe, “The US national quantum initiative,” *Quantum Science and Technology*, vol. 4, no. 2, p. 020504, 2019. 1

[3] R. Landauer, “Information is Physical,” *Phys. Today*, vol. 44, no. 5, p. 23, Jan 2008. 1

[4] J. Preskill, “Quantum computing and the entanglement frontier,” *arXiv:1203.5813*, Nov. 2012. 1

[5] I. Durham, D. Garisto, and K. Wiesner, “Physicists Need to Be More Careful with How They Name Things,” *Scientific American*, Feb 2021. 1

[6] C. Palacios-Berraquero, L. Mueck, and D. M. Persaud, “Instead of ‘supremacy’ use ‘quantum advantage’,” *Nature*, vol. 576, no. 7786, pp. 213–213, Dec. 2019. 1

[7] J. Preskill, “Why I Called It ‘Quantum Supremacy’,” *Quanta Magazine*, Oct. 2019. 1

[8] T. D. Ladd, F. Jelezko, R. Laflamme, Y. Nakamura *et al.*, “Quantum computers,” *Nature*, vol. 464, no. 7285, pp. 45–53, Mar 2010. 1

[9] A. J. Heinrich, W. D. Oliver, L. M. Vandersypen, A. Ardavan *et al.*, “Quantum-coherent nanoscience,” *Nat. Nanotechnol.*, pp. 1–12, 2021. 1

[10] S. Rasmussen, K. Christensen, S. Pedersen, L. Kristensen *et al.*, “Superconducting circuit companion—an introduction with worked examples,” *PRX Quantum*, vol. 2, p. 040204, Dec 2021. 1

[11] M. Kjaergaard, M. E. Schwartz, J. Braumüller, P. Krantz *et al.*, “Superconducting Qubits: Current State of Play,” *Annu. Rev. Condens. Matter Phys.*, vol. 11, no. 1, pp. 369–395, Mar 2020. 1, 2

[12] B. D. Josephson, “Possible new effects in superconductive tunnelling,” *Physics Letters*, vol. 1, no. 7, pp. 251–253, Jul 1962. 1

[13] M. Tinkham, *Introduction to Superconductivity: Second Edition (Dover Books on Physics)*. Dover Publications, Jun 2004. 2

[14] A. Blais, A. L. Grimsme, S. Girvin, and A. Wallraff, “Circuit quantum electrodynamics,” *Rev. Mod. Phys.*, vol. 93, no. 2, p. 025005, 2021. 2

[15] J. Koch, T. M. Yu, J. Gambetta, A. A. Houck *et al.*, “Charge-insensitive qubit design derived from the cooper pair box,” *Phys. Rev. A*, vol. 76, p. 042319, Oct 2007. 2

[16] F. Arute, K. Arya, R. Babbush, D. Bacon *et al.*, “Quantum supremacy using a programmable superconducting processor,” *Nature*, vol. 574, no. 7779, pp. 505–510, Oct 2019. 2, 7

[17] Q. Zhu, S. Cao, F. Chen, M.-C. Chen *et al.*, “Quantum computational advantage via 60-qubit 24-cycle random circuit sampling,” *Sci. Bull.*, p. S2095927321006733, Oct. 2021. 2, 7

[18] Y. Wu, W.-S. Bao, S. Cao, F. Chen *et al.*, “Strong Quantum Computational Advantage Using a Superconducting Quantum Processor,” *Phys. Rev. Lett.*, vol. 127, no. 18, p. 180501, Oct. 2021. 2, 7

[19] D. J. Wineland, “Nobel lecture: Superposition, entanglement, and raising schrödinger’s cat,” *Rev. Mod. Phys.*, vol. 85, pp. 1103–1114, Jul 2013. 2

[20] S. Haroche, “Nobel lecture: Controlling photons in a box and exploring the quantum to classical boundary,” *Rev. Mod. Phys.*, vol. 85, pp. 1083–1102, Jul 2013. 2

[21] C. D. Bruzewicz, J. Chiaverini, R. McConnell, and J. M. Sage, “Trapped-ion quantum computing: Progress and challenges,” *Appl. Phys. Rev.*, vol. 6, no. 2, p. 021314, 2019. 2

[22] C. Monroe, W. C. Campbell, L.-M. Duan, Z.-X. Gong *et al.*, “Programmable quantum simulations of spin systems with trapped ions,” *Rev. Mod. Phys.*, vol. 93, p. 025001, Apr 2021. 2

[23] C. Ryan-Anderson, J. G. Bohnet, K. Lee, D. Gresh *et al.*, “Realization of real-time fault-tolerant quantum error correction,” *Phys. Rev. X*, vol. 11, p. 041058, Dec 2021. 2

[24] X. Wu, X. Liang, Y. Tian, F. Yang *et al.*, “A concise review of Rydberg atom based quantum computation and quantum simulation,” *Chin. Phys. B*, vol. 30, no. 2, p. 020305, Feb 2021. 2

[25] S. Ebadi, T. T. Wang, H. Levine, A. Keesling *et al.*, “Quantum phases of matter on a 256-atom programmable quantum simulator,” *Nature*, vol. 595, no. 7866, pp. 227–232, Jul 2021. 2

[26] D. Bluvstein, H. Levine, G. Semeghini, T. T. Wang *et al.*, “A quantum processor based on coherent transport of entangled atom arrays,” *arXiv:2112.03923*, Dec. 2021. 2

[27] T. M. Graham, Y. Song, J. Scott, C. Poole *et al.*, “Demonstration of multi-qubit entanglement and algorithms on a programmable neutral atom quantum computer,” dec 2022. 2

[28] T. Xin, B.-X. Wang, K.-R. Li, X.-Y. Kong *et al.*, “Nuclear magnetic resonance for quantum computing: Techniques and recent achievements,” *Chin. Phys. B*, vol. 27, no. 2, p. 020308, 2018. 2

[29] D. G. Cory, M. D. Price, and T. F. Havel, “Nuclear magnetic resonance spectroscopy: An experimentally accessible paradigm for quantum computing,” *Physica D*, vol. 120, no. 1-2, pp. 82–101, 1998. 2

[30] E. Knill, R. Laflamme, and G. J. Milburn, “A scheme for efficient quantum computation with linear optics,” *Nature*, vol. 409, no. 6816, pp. 46–52, 2001. 2, 7

[31] S. Bartolucci, P. M. Birchall, M. Gimeno-Segovia, E. Johnston *et al.*, “Creation of Entangled Photonic States Using Linear Optics,” *arXiv:2106.13825*, Jun. 2021. 2

[32] J. E. Bourassa, R. N. Alexander, M. Vasmer, A. Patil *et al.*, “Blueprint for a scalable photonic fault-tolerant quantum computer,” *Quantum*, vol. 5, p. 392, 2021. 2

[33] A. Chatterjee, P. Stevenson, S. De Franceschi, A. Morello *et al.*, “Semiconductor qubits in practice,” *Nat. Rev. Phys.*, vol. 3, no. 3, pp. 157–177, Mar 2021. 2

[34] J. Zhang, L. Xu, G. Bian, P. Fan *et al.*, “Diamond Nitrogen-Vacancy Center Magnetometry: Advances and Challenges,” *arXiv:2010.10231*, Oct. 2020. 2

[35] J. D. Whitfield, *Understanding the Schrödinger Equation: Some [Non]Linear Perspective*. Nova Scientific Publishing, 2020, ch. “Understanding the Schrödinger Equation as a Kinematic Statement: A Probability-First Approach to Quantum”. 2, 3

[36] M. A. Nielsen and I. Chuang, *Quantum computation and quantum information*. American Association of Physics Teachers, 2002. 2, 5, 6, 8

[37] C. M. Grinstead and J. L. Snell, *Introduction to probability*. American Mathematical Soc., 2012. 2

[38] R. A. Horn and C. R. Johnson, *Matrix analysis*, 2nd ed. Cambridge ; New York: Cambridge University Press, 2012. 3

[39] S. T. Thornton and J. B. Marion, *Classical dynamics of particles and systems*, 5th ed. Brooks/Cole, 2004. 3

[40] D. J. Griffiths, *Introduction to Quantum Mechanics*. Cambridge, England, UK: Cambridge University Press, Aug 2018. 3

[41] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, England, UK: Cambridge University Press, Oct 2012. 3, 6

[42] F. G. S. L. Brandao, M. Christandl, A. W. Harrow, and M. Walter, “The Mathematics of Entanglement,” *arXiv:1604.01790*, Apr. 2016. 4

[43] J. I. Cirac, D. Pérez-García, N. Schuch, and F. Verstraete, “Matrix product states and projected entangled pair states: Concepts, symmetries, theorems,” *Rev. Mod. Phys.*, vol. 93, p. 045003, Dec 2021. 4, 7

[44] F. Verstraete, V. Murg, and J. I. Cirac, “Matrix product states, projected entangled pair states, and variational renormalization group methods for quantum spin systems,” *Adv. Phys.*, vol. 57, no. 2, pp. 143–224, Mar 2008. 4, 7

[45] G. Vidal, “Efficient Classical Simulation of Slightly Entangled Quantum Computations,” *Phys. Rev. Lett.*, vol. 91, no. 14, p. 147902, Oct 2003. 4

[46] F. Pan, K. Chen, and P. Zhang, “Solving the sampling problem of the Sycamore quantum supremacy circuits,” *arXiv:2111.03011*, Nov. 2021. 4, 7

[47] A. Einstein, B. Podolsky, and N. Rosen, “Can Quantum-Mechanical Description of Physical Reality Be Considered Complete?” *Phys. Rev.*, vol. 47, no. 10, pp. 777–780, May 1935. 4

[48] J. S. Bell, “On the Einstein Podolsky Rosen paradox,” *Phys. Phys. Fiz.*, vol. 1, no. 3, pp. 195–200, Nov 1964. 4

[49] M. Giustina, M. A. M. Versteegh, S. Wengerowsky, J. Handsteiner *et al.*, “Significant-loophole-free test of bell’s theorem with entangled photons,” *Phys. Rev. Lett.*, vol. 115, p. 250401, Dec 2015. 4

[50] A. Broadbent and C. Schaffner, “Quantum cryptography beyond quantum key distribution,” *Designs, Codes and Cryptography*, vol. 78, no. 1, pp. 351–382, 2016. 4

[51] “Learn quantum computation using qiskit,” <http://qiskit.org/textbook> [Accessed: Jan 7, 2022]. 5, 11

[52] S. Lloyd, “Almost any quantum logic gate is universal,” *Phys. Rev. Lett.*, vol. 75, pp. 346–349, Jul 1995. 6

[53] D. E. Deutsch, A. Barenco, and A. Ekert, “Universality in quantum computation,” *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, vol. 449, no. 1937, pp. 669–677, 1995. 6

[54] D. Gottesman, “The Heisenberg Representation of Quantum Computers,” *arXiv:quant-ph/9807006*, Jul. 1998. 6

[55] L. G. Valiant, “Quantum circuits that can be simulated classically in polynomial time,” *SIAM J. Comput.*, vol. 31, no. 4, pp. 1229–1254, 2002. 6

[56] B. M. Terhal and D. P. DiVincenzo, “Classical simulation of noninteracting-fermion quantum circuits,” *Phys. Rev. A*, vol. 65, no. 3, p. 032325, Mar 2002. 6

[57] D. Gottesman, “An introduction to quantum error correction and fault-tolerant quantum computation,” in *Quantum information science and its contributions to mathematics, Proceedings of Symposia in Applied Mathematics*, vol. 68, 2010, pp. 13–58. 6, 8

[58] S. M. Girvin, “Introduction to Quantum Error Correction and Fault Tolerance,” *arXiv:2111.08894*, Nov. 2021. 6, 8

[59] D. Deutsch, “Quantum theory, the church-turing principle and the universal quantum computer,” *Proc. R. Soc. London A - Math. Phys. Sci.*, vol. 400, no. 1818, pp. 97–117, 1985. 6

[60] F. Barahona, “On the computational complexity of ising spin glass models,” *J. Phys. A. Math. Gen.*, vol. 15, no. 10, p. 3241, 1982. 6, 9

[61] S. Aaronson and A. Arkhipov, “The Computational Complexity of Linear Optics,” *Theory of Computing*, vol. 9, no. 1, pp. 143–252, Feb 2013. 6, 7

[62] M. Sipser, *Introduction to the theory of computation*, 3rd ed. Boston, MA: Cengage Learning, 2012. 6, 9

[63] E. Bernstein and U. Vazirani, “Quantum complexity theory,” *SIAM Journal on computing*, vol. 26, no. 5, pp. 1411–1473, 1997. 6, 8

[64] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, “Quantum Computation by Adiabatic Evolution,” *arXiv:quant-ph/0001106*, Jan. 2000. 6, 9

[65] D. Aharonov, W. Van Dam, J. Kempe, Z. Landau *et al.*, “Adiabatic quantum computation is equivalent to standard quantum computation,” *SIAM J. Comp.*, vol. 37, no. 1, pp. 166–194, 2007. 6

[66] A. M. Childs, “Universal computation by quantum walk,” *Phys. Rev. Lett.*, vol. 102, no. 18, p. 180501, 2009. 6

[67] F. Verstraete, M. M. Wolf, and J. I. Cirac, “Quantum computation and quantum-state engineering driven by dissipation,” *Nat. Phys.*, vol. 5, no. 9, pp. 633–636, 2009. 6

[68] E. Farhi, J. Goldstone, and S. Gutmann, “A Quantum Approximate Optimization Algorithm,” *arXiv:1411.4028*, Nov. 2014. 6

[69] J. Biamonte, “Universal variational quantum computation,” *Phys. Rev. A*, vol. 103, no. 3, p. L030401, 2021. 6

[70] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, “Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics,” in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, 2019, pp. 193–204. 6, 9, 10

[71] A. P. Lund, M. J. Bremner, and T. C. Ralph, “Quantum sampling problems, BosonSampling and quantum supremacy,” *NJP Quantum Inf.*, vol. 3, no. 1, pp. 1–8, 2017. 6, 7

[72] A. W. Harrow and A. Montanaro, “Quantum computational supremacy,” *Nature*, vol. 549, no. 7671, pp. 203–209, 2017. 6, 7

[73] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, R. Babbush *et al.*, “Characterizing quantum supremacy in near-term devices,” *Nat. Phys.*, vol. 14, no. 6, pp. 595–600, 2018. 6, 7

[74] J. Gray and S. Kourtis, “Hyper-optimized tensor network contraction,” *Quantum*, vol. 5, p. 410, 2021. 7

[75] Y. Zhou, E. M. Stoudenmire, and X. Waintal, “What limits the simulation of quantum computers?” *arXiv:2002.07730*, Feb. 2020. 7

[76] X. Liu, C. Guo, Y. Liu, Y. Yang *et al.*, “Redefining the quantum supremacy baseline with a new generation sunway supercomputer,” *arXiv:2111.01066*, Nov. 2021. 7

[77] G. Kalachev, P. Panteleev, and M.-H. Yung, “Recursive multi-tensor contraction for xeb verification of quantum circuits,” *arXiv:2108.05665*, Aug. 2021. 7

[78] S. Aaronson and S. Gunn, “On the classical hardness of spoofing linear cross-entropy benchmarking,” *arXiv:1910.12085*, Oct. 2019. 7

[79] X. Gao, M. Kalinowski, C.-N. Chou, M. D. Lukin *et al.*, “Limitations of linear cross-entropy as a measure for quantum advantage,” *arXiv:2112.01657*, Dec. 2021. 7

[80] H.-S. Zhong, H. Wang, Y.-H. Deng, M.-C. Chen *et al.*, “Quantum computational advantage using photons,” *Science*, vol. 370, no. 6523, pp. 1460–1463, Dec. 2020. 7

[81] H.-S. Zhong, Y.-H. Deng, J. Qin, H. Wang *et al.*, “Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light,” *Phys. Rev. Lett.*, vol. 127, no. 18, p. 180502, Oct. 2021. 7

[82] B. Villalonga, M. Y. Niu, L. Li, H. Neven *et al.*, “Efficient approximation of experimental Gaussian boson sampling,” *arXiv:2109.11525*, Sep. 2021. 7

[83] A. Montanaro, “Quantum algorithms: an overview,” *NPJ Quantum Inf.*, vol. 2, no. 1, pp. 1–8, 2016. 7

[84] S. Jordan, “Quantum Algorithm Zoo,” 2022, <https://quantumalgorithmzoo.org/> [Accessed: Jan 3, 2022]. 7

[85] J. Preskill, “Quantum computing in the NISQ era and beyond,” *Quantum*, vol. 2, p. 79, 2018. 7

[86] —, “Quantum computing 40 years later,” *arXiv:2106.10522*, June 2021. 8

[87] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *STOC ’96: Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing*. New York, NY, USA: Association for Computing Machinery, Jul 1996, pp. 212–219. 8

[88] C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, “Strengths and weaknesses of quantum computing,” *SIAM J. Comput.*, vol. 26, no. 5, pp. 1510–1523, 1997. 8

[89] P. W. Shor, “Algorithms for quantum computation: discrete logarithms and factoring,” in *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE, Nov 1994, pp. 124–134. 8

[90] A. Y. Kitaev, “Quantum measurements and the Abelian Stabilizer Problem,” *arXiv:quant-ph/9511026*, Nov. 1995. 8

[91] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum Algorithm for Linear Systems of Equations,” *Phys. Rev. Lett.*, vol. 103, no. 15, p. 150502, Oct 2009. 8

[92] R. L. Rivest, A. Shamir, and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems,” *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, 1978. 8

[93] R. P. Feynman, “Simulating physics with computers,” *Int. J. Theor. Phys.*, vol. 21, no. 6, pp. 467–488, Jun 1982. 8

[94] S. Lloyd, “Universal quantum simulators,” *Science*, vol. 273, pp. 1073–1078, 1996. 8

[95] J. D. Whitfield, J. Biamonte, and A. Aspuru-Guzik, “Simulation of electronic structure hamiltonians using quantum computers,” *Mol. Phys.*, vol. 109, no. 5, pp. 735–750, 2011. 8, 9

[96] I. M. Georgescu, S. Ashhab, and F. Nori, “Quantum simulation,” *Rev. Mod. Phys.*, vol. 86, pp. 153–185, Mar 2014. 8, 9

[97] B. Bauer, S. Bravyi, M. Motta, and G. K.-L. Chan, “Quantum algorithms for quantum chemistry and quantum materials science,” *Chem. Rev.*, vol. 120, no. 22, pp. 12 685–12 717, 2020. 8

[98] Y. Cao, J. Romero, J. P. Olson, M. Degroote *et al.*, “Quantum Chemistry in the Age of Quantum Computing,” *Chem. Rev.*, vol. 119, no. 19, pp. 10 856–10 915, Oct 2019. 8, 9

[99] A. Y. Kitaev, A. Shen, M. N. Vyalyi, and M. N. Vyalyi, *Classical and quantum computation*. American Mathematical Soc., 2002, no. 47. 8, 9

[100] S. Gharibian, Y. Huang, Z. Landau, and S. W. Shin, “Quantum hamiltonian complexity,” *Found. Trends Theor. Comput. Sci.*, vol. 10, no. 3, pp. 159–282, 2015. 8, 9

[101] J. Tilly, H. Chen, S. Cao, D. Picozzi *et al.*, “The Variational Quantum Eigensolver: a review of methods and best practices,” *arXiv:2111.05176*, Nov. 2021. 8

[102] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin *et al.*, “Variational quantum algorithms,” *Nat. Rev. Phys.*, pp. 1–20, 2021. 8

[103] R. Orús, S. Mugel, and E. Lizaso, “Quantum computing for finance: Overview and prospects,” *Rev. Phys.*, vol. 4, p. 100028, 2019. 9

[104] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost *et al.*, “Quantum machine learning,” *Nature*, vol. 549, pp. 195–202, Sep 2017. 9

[105] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung *et al.*, “A variational eigenvalue solver on a photonic quantum processor,” *Nat. Commun.*, vol. 5, no. 1, pp. 1–7, 2014. 9

[106] A. Kandala, A. Mezzacapo, K. Temme, M. Takita *et al.*, “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets,” *Nature*, vol. 549, no. 7671, pp. 242–246, Sep 2017. 9

[107] P. J. J. O’Malley, R. Babbush, I. D. Kivlichan, J. Romero *et al.*, “Scalable Quantum Simulation of Molecular Energies,” *Phys. Rev. X*, vol. 6, no. 3, p. 031007, Jul 2016. 9

[108] R. Ayanzadeh, J. Dorband, M. Haleem, and T. Finin, “Multi-qubit correction for quantum annealers,” *Sci. Rep.*, vol. 11, no. 1, pp. 1–12, 2021. 9

- [109] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, “Grand unification of quantum algorithms,” *PRX Quantum*, vol. 2, no. 4, 2021. 10
- [110] G. H. Low and I. L. Chuang, “Hamiltonian simulation by qubitization,” *Quantum*, vol. 3, p. 163, 2019. 10
- [111] M. Szegedy, “Quantum speed-up of markov chain based algorithms,” in *45th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2004, pp. 32–41. 10
- [112] M. S. Anis *et al.*, “Qiskit: An open-source framework for quantum computing,” doi:10.5281/zenodo.2573505 [Accessed: Jan 3, 2022]. 11
- [113] “Amazon braket,” <https://aws.amazon.com/braket/> [Accessed: Jan 7, 2022]. 11
- [114] C. Albornoz, G. Alonso, P. A. M. Andrenkov, A. Asadi *et al.*, “Xanadu quantum codebook,” 2021, <https://codebook.xanadu.ai> [Accessed: Jan 7, 2022]. 11
- [115] “qBraid: Cloud-based IDE for quantum computing,” <https://qbraid.com> [Accessed: Jan 7, 2022]. 11
- [116] “Creative commons attribution 4.0 international (cc by 4.0)” <https://creativecommons.org/licenses/by/4.0/> [Accessed: Jan 7, 2022]. 11