Survey of Quantum Algorithms: Foundations, Frameworks and Applications

Ashish Kumar Patra[1], Anurag K. S. V.[1], Vikas Dattatraya Ghevade[1], Sai Shankar P.[1], Ruchika Bhat[2], Amit Saxena[3], Raghavendra V.[4], and Jaiganesh G.*[,1]

[1]Qclairvoyance Quantum Labs, Secunderabad, TG 500094, India.
[2]The University of Arizona, Tucson, AZ 85721, USA.
[3]Centre for Development of Advanced Computing, Pune, MH 411007, India.
[4]SRM Institute of Science and Technology, Chennai, TN 603203, India.

Abstract

Quantum computing has transformed computational paradigms through its unique principles and algorithms. This paper provides a systematic overview of quantum algorithms, their application-based classification, and their industrial relevance. It surveys the broader areas of the field from a chronological perspective and highlights significant advances. The study also identifies emerging trends and key research directions in quantum algorithm development.

Contents

*(Corresponding Author) drjaiganesh15@gmail.com, jaiganesh@qclairvoyance.in

1 INTRODUCTION

Quantum computing's rapid advancement has opened new frontiers in computational sciences, with a promise to solve problems, which are classically intractable [1]. At the heart of this technological revolution lies the development of quantum algorithms, which exploit the principles of quantum mechanics, such as interference, entanglement [2], and superposition, to achieve computational advantages over classical counterparts. Over the past few decades, the quantum computing landscape has evolved dramatically, giving rise to a diverse array of algorithms designed for applications ranging from cryptography and optimization to machine learning and quantum simulation.

Quantum algorithms can be classified into several broad categories. Early breakthroughs, such as **Shor's algorithm** [3] for factoring integers and Grover's algorithm [4] for database search, established foundational paradigms for quantum computation. More recently, new frameworks such as variational algorithms, quantum machine learning (QML) techniques, and sampling-based approaches have gained prominence, particularly in the context of **NISQ devices** [5]. These advancements reflect the dynamic interplay between theoretical innovation and practical implementation in the field.

Computational models underpinning quantum algorithms include the circuit model, adiabatic quantum computation [6], and quantum walks [7].

In addition, there are diverse application domains ranging from **cryptanalysis** and **optimization** to materials science and data analysis. Both well-established algorithms and newly proposed approaches contribute to the ongoing evolution of the field.

Quantum computing represents one of the most profound shifts in computational paradigms, leveraging the principles of quantum mechanics such as superposition, entanglement, and interference. Quantum computing's true potential was envisioned by Richard Feynman in 1982, built upon the foundations of quantum mechanics developed by pioneers such as Paul Dirac, Erwin Schrödinger, Max Born, Albert Einstein, Warner Heisenberg, Max Planck, Wolfgang Pauli in the early 20th century. He famously noted that "nature isn't classical, dammit, and if you want to make a simulation of nature, you'd better make it quantum mechanical," thereby initiating the conceptual groundwork for quantum computing [8]. This visionary statement highlighted the inadequacy of classical computers in simulating quantum systems and spurred the development of quantum computational models.

The idea of a universal quantum computer, its theoretical formulation [9], and the exponential speed-up that can be achieved by such a device, for certain processes such as function evaluation [10] were demonstrated in the early 1990s. The first tangible milestones in quantum computing were marked by the creation of quantum algorithms that demonstrated potential speed-ups over classical counterparts. Simon's algorithm [11] and Shor's factoring algorithm [12], developed in the 1990s, were pivotal. Shor's algorithm, in particular, posed a threat to classical cryptographic systems by efficiently factoring integers and solving discrete logarithm problems, both critical to RSA encryption.

In parallel, Grover's algorithm [4] provided a quadratic speedup for unstructured search problems, showcasing quantum computing's broader applicability. These initial successes highlighted the promise of quantum algorithms and catalyzed further research into their development.

The subsequent decades witnessed a rapid expansion in quantum algorithm research. The advent of the NISQ era introduced new algorithmic paradigms, such as Variational Quantum Algorithms (VQAs). Variational Quantum Eigensolver (VQE) [13] and the Quantum Approximate Optimization Algorithm (QAOA) [14] exemplify this class, merging quantum and classical computation to solve optimization and quantum chemistry problems efficiently on NISQ devices.

Quantum algorithms have since diversified into categories based on their applications, such as QML [15], quantum cryptography [16] [17] [18], and quantum simulations [19] [20] [13]. For instance, quantum principal component analysis [21] and quantum support vector machines [22] demonstrate the applicability of quantum algorithms to data analysis and machine learning, while quantum key distribution (QKD) [23] underpins advancements in secure communications.

Despite significant advancements, the journey toward large-scale, Fault-Tolerant Quan-

tum Computation(FTQC) remains fraught with challenges. Issues such as decoherence [24], error correction [25], and limited qubit connectivity necessitate innovative solutions. Furthermore, the development of high-level programming languages for quantum computing and efficient algorithmic implementations remains an active area of research.

Additionally difficulties faced in loading classical data into quantum computers (or) state-preparation [26], where lucrative hardware related improvements have been proposed recently[27] and the exponential overhead of measurements for quantum tomography [28] (to obtain the complete quantum state) also hinder a full-scale usage of quantum computers in a practical sense.

The primary goal of our work is to highlight the rapid progression of quantum algorithm development over recent decades. It also seeks to emphasize the emergence of diverse algorithmic approaches tailored to both the **NISQ** era and the anticipated era of **FTQC**.

The structure of this survey is as follows: **Section 1** introduces fundamental concepts; **Section 2** covers key quantum mechanical principles; **Section 3** presents a classification of quantum algorithms; **Section 4** discusses some of the primitive algorithms; **Section 5** explores industrial applications; **Section 6** outlines future directions; and **Section 7** concludes with a summary of key insights.

2 Background of Quantum Algorithms

2.1 A Brief Introduction to Qubits

Akin to a classical bit, a two-level quantum system called a qubit can be used to perform computational operations. Such a single pure qubit state can be defined in a generalized fashion as:

$$|\psi\rangle = e^{i\delta}(\cos\theta\,|0\rangle + e^{i\phi}\sin\theta\,|1\rangle). \tag{1}$$

This expression represents the **quantum superposition** [1] of the computational basis states $|0\rangle$ and $|1\rangle$, where a valid quantum state is a linear combination of the solutions to the time-independent Schrödinger equation[29] (computational basis):

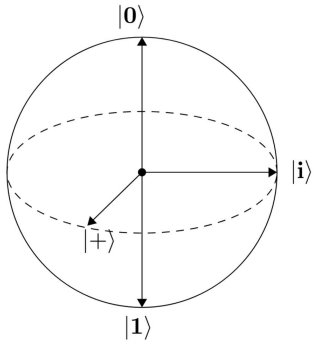

Figure 1: Bloch Sphere diagram

$$\hat{\mathbf{H}}\,|\psi\rangle = E\,|\psi\rangle \;\; ; |\psi\rangle = c_0\,|0\rangle + c_1\,|1\rangle\,; c_0, c_1 \in \mathbb{C} \tag{2}$$

The co-efficients of the basis states ($e^{i\delta}cos\theta$ and $e^{i(\delta+\phi)}sin\theta$) are complex amplitudes, and their squared magnitudes correspond to the probabilities of measurement outcomes.

Any manipulations to the quantum state can be modelled as operations on the qubit, which are called gates. A quantum gate $\hat{U}$ acts on a qubit (or set of qubits) and leads to a unitary evolution of the states of the qubit(s). A typical example of a quantum gate is the Hadamard Gate $\hat{H}$, given by:

$$\hat{H} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{3}$$

At least for the single qubit case, the evolution of a quantum state can be visualized on a 3-dimensional sphere called the Bloch Sphere. The usual convention is that the north

and south poles correspond to the states which will be measured directly, also called the computational basis states. The quantum states which are the eigenvectors of the bit-flip operator or Pauli X operator (which performs the following operations on a set of computational qubits: $|0\rangle \longrightarrow |1\rangle$ and vice-versa) are pointing towards the positive and negative x-axis respectively, and similarly with the eigenvectors of the Pauli-Y operator.

When more than one quantum system is considered, an additional uniquely quantum phenomenon arises: **entanglement** [1]. In entangled states, the quantum systems exhibit correlations that cannot be described independently of one another, regardless of the spatial separation between them. A key feature of entangled states is that they cannot be expressed as a tensor product of individual qubit states. For example, the Bell state:

$$\left|\Phi^+\right\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{4}$$

cannot be written as $|\psi\rangle_A \otimes |\phi\rangle_B$ for any single-qubit states $|\psi\rangle_A$ and $|\phi\rangle_B$ belonging to $\mathbb{H}^2$. This non-separability is the hallmark of entanglement and plays a central role in quantum information protocols and algorithms.

2.1.1 Quantum Gates - Introduction

A general single-qubit unitary operator can be represented as:

$$\hat{U} = \begin{bmatrix} U_{00} & U_{01} \\ U_{01} & U_{11} \end{bmatrix}, \tag{5}$$

where $\hat{U}\hat{U}^\dagger = \hat{U}^\dagger\hat{U} = \mathbb{I}$. The unitarity condition imposes the following constraints on the matrix elements:

$$\begin{aligned} |U_{00}|^2 + |U_{10}|^2 = |U_{01}|^2 + |U_{11}|^2 = 1, \\ U_{00}U_{01}^* + U_{10}U_{11}^* = 0. \end{aligned} \tag{6}$$

Equivalently, the columns (and rows) of a unitary matrix form orthonormal vectors in $\mathbb{C}^2$.

Most of the commonly used two-qubit gates in quantum computing can be broadly categorized into two types based on their operational structure:

- **Block-diagonal (Controlled-type) Gates** — These gates act conditionally on the state of one qubit and can be written in the form:

$$\hat{U}_{\text{ctrl}} = |0\rangle\langle 0| \otimes \hat{U} + |1\rangle\langle 1| \otimes \hat{V} = \begin{bmatrix} U_{00} & U_{01} & 0 & 0 \\ U_{10} & U_{11} & 0 & 0 \\ 0 & 0 & V_{00} & V_{01} \\ 0 & 0 & V_{10} & V_{11} \end{bmatrix}, \tag{7}$$

  where $\hat{U}$ and $\hat{V}$ are single-qubit unitary operators. This structure encompasses all controlled operations, including the CNOT and controlled-U gates. Such gates are block-diagonal in the computational basis and are typically entangling when $\hat{U} \neq \hat{V}$. In the case when $\hat{U} = \mathbb{I}$, and $\hat{V}$ is any other unitary gate, this leads to a **controlled operation of $\hat{V}$ on the second qubit if the first qubit is in $|1\rangle$ state.**

- **Permutation-type (Exchange-type) Gates**(also categorized as Non-Perfect Entanglers in [30]) — These gates act nontrivially only on the single-excitation subspace span$\{|01\rangle, |10\rangle\}$ and can be expressed as:

$$\hat{U}_{\text{perm}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & U_{00} & U_{01} & 0 \\ 0 & U_{10} & U_{11} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad \begin{bmatrix} U_{00} & U_{01} \\ U_{10} & U_{11} \end{bmatrix} \in U(2). \tag{8}$$

  These include gates such as SWAP, iSWAP, $\sqrt{\text{iSWAP}}$, and the fermionic SWAP, which preserve total excitation number and are frequently used in quantum simulation and fermionic encodings.

Both classes of gates are inherently *non-separable*, meaning they cannot be written as a tensor product of two single-qubit unitaries[31]:

$$\hat{K} \neq \hat{A} \otimes \hat{B}, \quad \hat{K} \in U(4), \quad \hat{K} \in \{\hat{U}_{\text{ctrl}}, \hat{U}_{\text{perm}}\}, \quad \hat{A}, \hat{B} \in U(2). \tag{9}$$

where $U(2)$ and $U(4)$ represent the groups containing all one qubit unitaries and two qubit unitaries respectively. They therefore serve as the fundamental non-local building blocks in quantum circuits.

The commonly used single-qubit gates and their matrix representations are: Pauli matrices:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \tag{10}$$

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \qquad P(\phi) = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{bmatrix},$$

$$R_x(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}, \quad R_y(\theta) = \begin{bmatrix} \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}, \quad R_z(\theta) = \begin{bmatrix} e^{-i\theta/2} & 0 \\ 0 & e^{i\theta/2} \end{bmatrix}.$$

Two-qubit gates can often be expressed as either block-diagonal (entangling) or permutation-type gates. A few commonly used examples are:

Controlled Gates (Entangling):

$$C\text{-}X : U = I,\, V = X, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \tag{11}$$

$$C\text{-}R_x(\theta) : U = I,\, V = e^{-i\frac{\theta}{2}X}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ 0 & 0 & -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}, \tag{12}$$

$$C\text{-}R_y(\theta) : U = I,\, V = e^{-i\frac{\theta}{2}Y}, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \\ 0 & 0 & \sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{bmatrix}, \tag{13}$$

$$C\text{-}R_z(\theta) : U = I,\ V = e^{-i\frac{\theta}{2}Z},\ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & e^{-i\theta/2} & 0 \\ 0 & 0 & 0 & e^{i\theta/2} \end{bmatrix}. \tag{14}$$

Permutation Gate:

$$SWAP : \begin{bmatrix} U_{00} & U_{01} \\ U_{10} & U_{11} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \qquad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

2.2 Relevant Concepts for the Quantum Algorithms

2.2.1 Variational Quantum Theorem

Given a Hamiltonian H, we begin with a trial quantum state $\left|\overline{0}\right\rangle$, which is an attempt to imitate the true ground state $|0\rangle$ of the Hamiltonian. We define the value $\overline{E}$ in the following manner: $\overline{E} := \frac{\langle\overline{0}|H|\overline{0}\rangle}{\langle\overline{0}|\overline{0}\rangle}$ and the true ground state energy (or eigen-value) to be E_0 (i.e., $\langle 0| H |0\rangle$).The variational theorem states that[29]:

$$\overline{E} \geq E_0 \tag{15}$$

Hence, by trying out various types of trial states $\left|\overline{0}\right\rangle$ or parameterizing accordingly, we can obtain the upper bound to E_0.

2.2.2 Quantum Phase Kickback

As we have encountered two-qubit gates previously in Sec.2.1.1 which perform a controlled operation, we had a clear distinction that the state of the control qubit will affect the target qubit's state and never otherwise. But for certain states of the target qubit(where the target qubit is a eigenstate of the unitary operator with an eigenvalue other than 1), the local phase of the control qubit is affected based on the state of the target qubit. This phenomenon is the so-called quantum phase-kickback[32].

In a general sense, consider a bi-partite product state of the form $|\phi\rangle \otimes |\psi\rangle$, where $|\phi\rangle := c_1 |0\rangle + c_2 |1\rangle$. We define a unitary U, which satisfies the property $U |\psi\rangle = e^{i\psi} |\psi\rangle$ (U is an eigen-operator of the eigen-state $|\psi\rangle$ with an eigen-value $e^{i\psi}$).

The action of the gate C-U (with control on the first qubit and target being the second qubit) on the bi-partite state $|\phi\rangle \otimes |\psi\rangle$ will lead to a state of the form $|\phi'\rangle \otimes |\psi\rangle$, where $|\phi'\rangle = c_1 |0\rangle + c_2 e^{i\psi} |1\rangle$.

As an example, consider the quantum state $|\phi\rangle = |ctr\rangle \otimes |tgt\rangle = |1\rangle \otimes (\frac{|0\rangle - |1\rangle}{2})$, where we designate the former as the *control* qubit and the latter as *target* qubit. Upon action of a controlled NOT Operation (C_X), the state will be transformed in the following manner:

$$C_X |\phi\rangle = |1\rangle \otimes \frac{|1\rangle - |0\rangle}{\sqrt{2}} = -|1\rangle \otimes \left(\frac{|0\rangle - |1\rangle}{\sqrt{2}}\right) \tag{16}$$

The whole quantum state acquired a global phase of -1, in this case. Considering another case (Figure 2), where the target qubit is in the state $|-\rangle = \frac{|0\rangle - |1\rangle}{\sqrt{2}}$, then the following phenomenon is observed.

$$
\begin{aligned}
C_X \left|+\right\rangle \otimes \left|-\right\rangle &= C_X \left(\frac{\left|00\right\rangle - \left|01\right\rangle + \left|10\right\rangle - \left|11\right\rangle}{2} \right) \\
&= \frac{1}{2} \left(\left|00\right\rangle - \left|01\right\rangle + \left|11\right\rangle - \left|10\right\rangle \right) \\
&= \left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}} \right) \otimes \left(\frac{\left|0\right\rangle - \left|1\right\rangle}{\sqrt{2}} \right) = \left|-\right\rangle \otimes \left|-\right\rangle
\end{aligned}
\tag{17}
$$

As is known, the state $\left|-\right\rangle$ is an eigen-state of the Pauli gate X, with eigenvalue -1, and that is exactly the local phase acquired by the control qubit. Had the target state been $\left|+\right\rangle$, then the local phase acquired by the control state would have been +1, or in other words, the state would have remained unchanged.

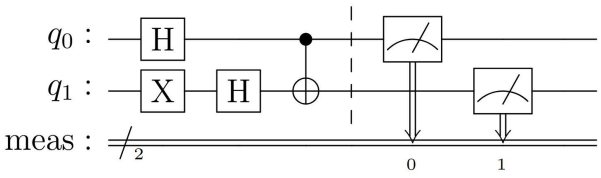

Figure 2: Example Quantum Circuit for Phase Kickback Demonstration

2.2.3 Transverse Ising model - connection to the QUBO problem

The transverse Ising model is a cornerstone in quantum mechanics and quantum computing, used to study spin systems and optimization problems. Its Hamiltonian is given by:

$$
\mathbf{H} = -J \sum_{\langle i,j \rangle} \sigma_i^z \sigma_j^z - h \sum_i \sigma_i^x, \tag{18}
$$

where:

- σ_i^z and σ_i^x are the Pauli matrices acting on the i-th spin.
- J is the coupling constant, representing the interaction strength between nearest neighbors $\langle i, j \rangle$.
- h is the transverse field strength, inducing quantum fluctuations.

The transverse Ising Hamiltonian can be connected to the Quadratic Unconstrained Binary Optimization (QUBO) problem through its classical counterpart. By considering the limit where the transverse field $h \to 0$, the Hamiltonian reduces to:

$$
\mathbf{H}_{\text{classical}} = -J \sum_{\langle i,j \rangle} s_i s_j, \tag{19}
$$

where $s_i \in \{-1, 1\}$ are classical spin variables. Mapping these spin variables to binary variables $x_i \in \{0, 1\}$ via the transformation $s_i = 2x_i - 1$, the Hamiltonian can be expressed as a QUBO problem:

$$\mathbf{H}_{\text{QUBO}} = \sum_{i,j} Q_{ij} x_i x_j, \tag{20}$$

where the coefficients Q_{ij} encode the problem constraints and objective function.

The transverse Ising model serves as a quantum framework for solving optimization problems, where the Hamiltonian's ground state corresponds to the optimal solution of the QUBO problem [33, 14].

2.2.4 Quantum Fourier Transform Subroutine

The **Quantum Fourier Transform (QFT)**[34] is the quantum analogue of the classical Discrete Fourier Transform (DFT). Similar to the DFT, which maps a discrete set of time-domain samples $f(t)$ to their representation in the frequency domain $\mathcal{F}(\omega)$, the QFT transforms quantum states from the computational basis to the Fourier basis. The action of the QFT on a computational basis state $|\psi_j\rangle$ is defined as

$$\hat{U}_{\text{QFT}} |\psi_j\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{i\frac{2\pi jk}{N}} |\psi_k\rangle, \quad |\psi_k\rangle \in \{0,1\}^{\log_2 N}. \tag{21}$$

The corresponding inverse transform is given by

$$\hat{U}^{\dagger}_{\text{QFT}} |\psi_k\rangle = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{-i\frac{2\pi jk}{N}} |\psi_j\rangle. \tag{22}$$

A more intuitive **product representation**[1] of the QFT can be expressed as

$$\begin{aligned} \hat{U}_{\text{QFT}} |j_1\rangle \otimes |j_2\rangle \cdots \otimes |j_n\rangle = \frac{1}{2^{n/2}} \left(|0\rangle + e^{2\pi i(0.j_n)} |1\rangle\right) \otimes \left(|0\rangle + e^{2\pi i(0.j_{n-1}j_n)}\right) \\ \ldots \otimes \left(|0\rangle + e^{2\pi i(0.j_1 j_2 \ldots j_n)}\right), \end{aligned} \tag{23}$$

which reveals how information encoded in the computational basis is distributed across the *local phases* of individual qubits. This phase-encoding property forms the foundation of several key quantum algorithms. In particular, the **QPE** algorithm (see Sec. 4.3.5) exploits the inverse QFT to extract eigenphase information from a unitary operator with exponential precision.

2.3 Key Developments and Breakthroughs

The first significant quantum algorithm was the Deutsch-Josza algorithm, introduced by **David Deutsch** and **Richard Jozsa** in 1992 [10]. This algorithm was among the first to demonstrate quantum advantage, solving a specific promise problem exponentially faster than any classical counterpart. It laid the foundation for later developments in quantum query complexity.

In 1995, **Alexei Kitaev** developed the notion of Quantum Phase Estimation(QPE)[35] by using a single ancilla qubit (Iterative QPE), which later was used in conjunction with the inverse of Quantum Fourier Transform(QFT) to create QPE, which later became a fundamental subroutine in many quantum applications, including Shor's Algorithm[36].

In 1996, **Lov Grover** proposed Grover's algorithm [4], which provides a quadratic speedup for unstructured search problems. Around the same time, Peter Shor introduced

his famous factorization algorithm [3], which demonstrated an exponential advantage over classical integer factorization methods, challenging the security of RSA cryptography.

The late 1990s saw the emergence of Simon's algorithm (1997) [11], which further reinforced the separation between quantum and classical computation. Around 2000, **Edward Farhi** introduced Quantum Annealing [6], a powerful approach for solving combinatorial optimization problems that inspired the development of quantum adiabatic algorithms.

The early 2000s brought further advancements in amplitude-based techniques. In 2002, **Gilles Brassard** and **Peter Høyer** developed Amplitude Amplification (which is a generalization of Grover's Search Algorithm) and Amplitude Estimation [37]. The Hadamard Test, introduced in 2006 by Aharonov, Jones, and Landau [38], became an essential primitive for estimating the expectation values of quantum states.

The **Harrow-Hassidim-Lloyd** (HHL) algorithm, developed in 2008 [39], marked a significant breakthrough in solving quantum linear systems, providing exponential speedup for certain matrix inversion problems. With HHL, the concept of classical data encoding into quantum computers began an even more active field of research. Algorithms like Quantum Signal Processing [40] and (Block Encoding [41], a sub-routine to encode classical data into quantum computers) were built to solve similar problems.

The following decade saw the introduction of Boson Sampling (2010) by **Aaronson** and **Arkhipov** [42], an algorithm that demonstrated *"quantum supremacy"* in photonic quantum computing.

QAOA introduced by **Farhi** et al., in 2014 [14], and VQE by **Peruzzo**, **Aspuru-Guzik** et al., in 2014 [13], became two of the most influential heuristic quantum algorithms for combinatorial optimization and quantum chemistry simulations respectively.

The latter half of the 2010s saw the introduction of Quantum Singular Value Transformation (QSVT) in 2019 by **Gilyén** and **Wiebe** [43], which further advanced quantum linear algebra techniques. More recently, in 2023, Bayesian Quantum Phase Estimation (Bayesian QPE) [44] was introduced by **Joseph Smith et al.,** improving upon traditional QPE methods by leveraging Bayesian inference.

Other notable algorithms which left an indelible mark on the field are **Linear Combination of Unitaries** (2012, Childs et al.,) [45], which would allow the operation of a non-unitary gate on a quantum system which can be written as a sum of unitaries.

These advancements are diagrammatically represented in Figure 3. They highlight the rapid evolution of quantum algorithms and their increasing applicability in diverse domains such as cryptography, optimization, and simulation. With continued progress in quantum hardware, these algorithms are expected to play an even greater role in solving complex computational problems.

3 Categories of Quantum Algorithms

The broad landscape of quantum algorithms can be categorized into several key classes based on their application domains. **Search Algorithms** include well-known examples like Grover's algorithm and Quantum Walks, which provide exponential and quadratic speedups for integer factorization and unstructured search, respectively. **Quantum Simulation** focuses on simulating quantum systems, such as molecules and materials, with algorithms like VQE and QPE. Shor's algorithm is described in the **Factorization Algorithms** section. **Quantum Optimization** leverages algorithms like QAOA to tackle combinatorial optimization problems. **Quantum Cryptography** employs protocols

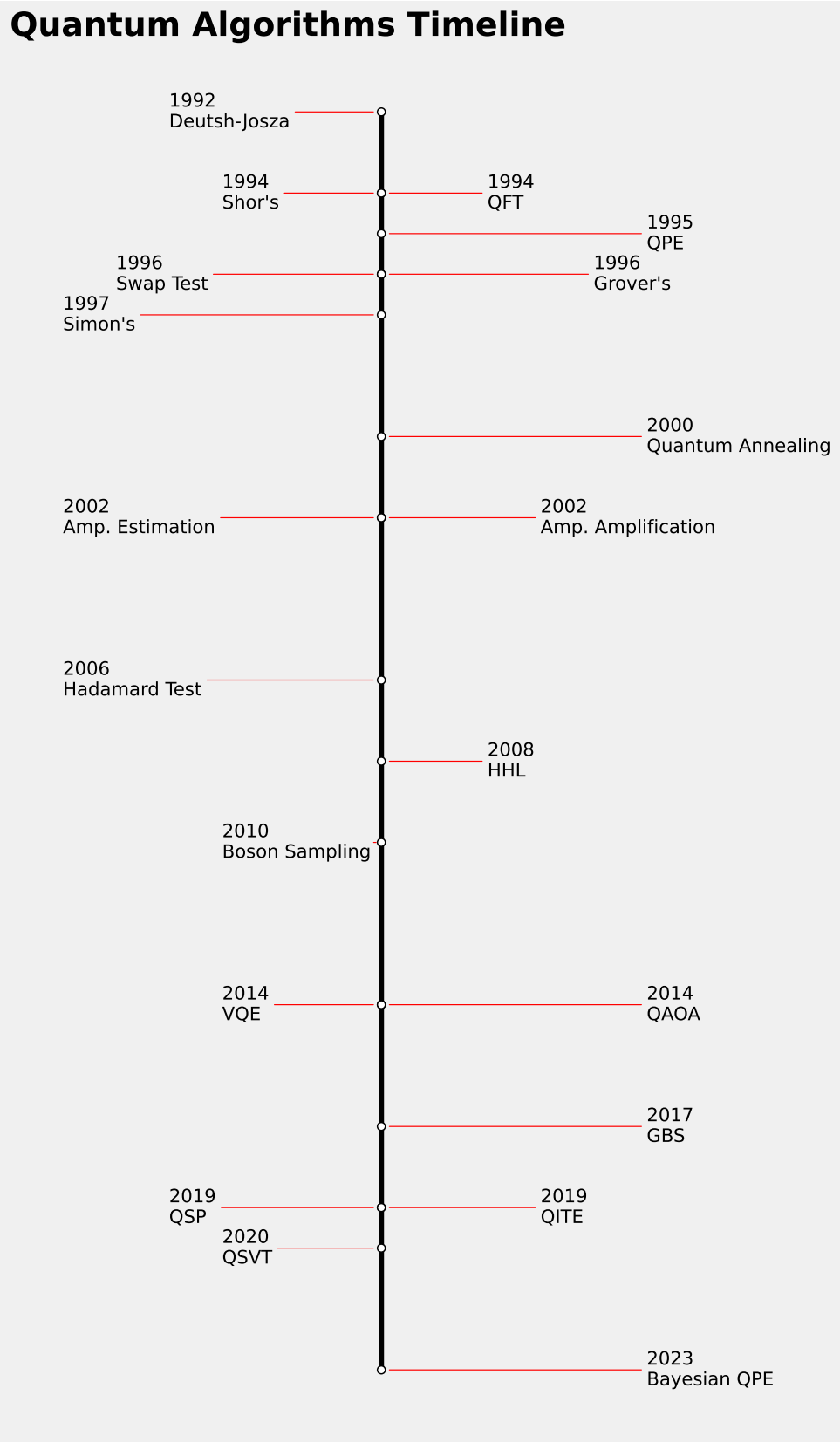

Figure 3: Quantum Algorithms Timeline

such as BB84 and B92 to ensure secure communication using principles of quantum mechanics. **QML Algorithms** aim to enhance classical machine learning with quantum resources, including algorithms like quantum support vector machines and quantum principal component analysis. Lastly, **Quantum Inspired Algorithms** and **Classical Simulation of Quantum Systems** utilize quantum principles to develop efficient classical algorithms and simulate quantum phenomena on classical hardware.

As of today, over 130 quantum algorithms have been systematically classified into distinct categories based on their objectives and methodologies in the paper [46]. Table 1 summarizes these classifications, by showing some popular quantum algorithms under each category of broader application.

Table 1: Some Famous Quantum Algorithms categorized based on Application

Category	**Examples**
Search	Grover's[4], Hidden subgroup problems[11]
Arithmetic	HHL [39],QSVT[43]
Optimization	QAOA[14], Quantum Annealing[6]
Machine Learning	Quantum Neural Networks[47], Quantum SVMs[22]
Simulation	VQE[13], PQE [20]
Factorization	Shor's[12]
Cryptography	QKD[16][18]

Quantum algorithms have transcended theoretical confines to influence diverse domains. Quantum simulations provide insights into molecular structures and chemical reactions, aiding drug discovery [48]. Optimization algorithms like QAOA are revolutionizing logistical and financial operations, while quantum cryptography ensures secure communication in the face of advancing quantum threats.

The impact of quantum algorithms extends further into artificial intelligence, energy optimization, and weather forecasting. This widespread applicability underscores the transformative potential of quantum computing in addressing real-world challenges. Based on applications(Figure 4), the quantum algorithms can be broadly classified into the following sections:

3.1 Quantum Search

Quantum search algorithms are designed to enhance the efficiency of search processes in unstructured data sets. Grover's algorithm [4] is the most notable example, achieving a quadratic speedup over classical algorithms by reducing the search complexity from $O(N)$ to $O(\sqrt{N})$. Amplitude amplification [37] extends Grover's technique and is employed in a variety of applications, including optimization and decision-making tasks. These algorithms underscore the potential of quantum computing in handling large-scale database searches and have been extensively studied for their applicability in fields such as bio-informatics and cryptography.

3.2 Optimization

Optimization algorithms such as QAOA [14] and the VQE [13] leverage quantum-classical hybrid approaches to solve combinatorial and continuous optimization problems. QAOA, inspired by adiabatic quantum computing, is particularly effective in solving graph theory problems, while VQE has found applications in quantum chemistry and material science.

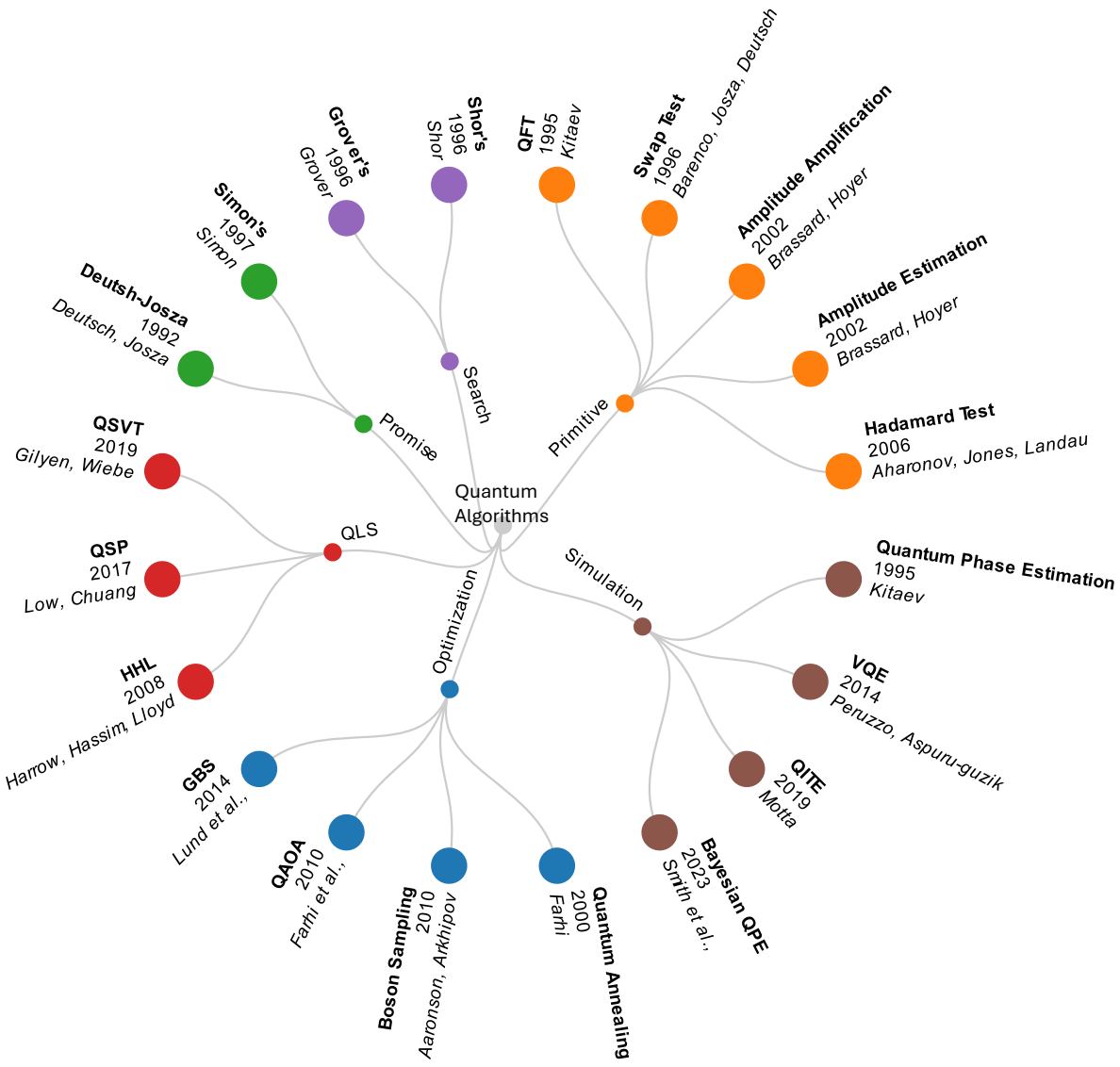

Figure 4: Quantum Algorithms classified based on Application

These algorithms represent a significant stride in addressing real-world optimization challenges, including supply chain logistics and energy minimization.

3.3 Simulation

Quantum simulation algorithms are tailored to model complex quantum systems that are computationally intractable for classical computers. Quantum chemistry simulations [49][50][13][51][20] leverage the intrinsic properties of quantum systems to study molecular structures and reactions, aiding drug discovery and material design. Quantum Monte Carlo methods [52] combine classical stochastic techniques with quantum enhancements to explore physical systems with high precision. These algorithms are critical for advancing scientific research in chemistry, physics, and biology.

3.4 Factorization

Quantum factorization algorithms aim to efficiently decompose large composite integers into their prime factors, a problem that threatens the security of widely used classical cryptographic algorithms such as Rivest-Shamir-Adleman (RSA). Shor's algorithm [36] achieves this by employing quantum phase estimation and the inverse of QFT to determine the period of modular exponentiation functions, enabling factorization in polynomial time, leading to an exponential speed-up in comparison to existing classical algorithms.

3.5 Quantum Arithmetic

Quantum arithmetic-based algorithms enable efficient solutions to linear algebra and optimization problems, crucial for applications in quantum recommendation systems [53] and finance. The HHL algorithm [39] and its extensions [54, 55] accelerate solving linear systems, while QSVT [43] enhances matrix manipulations. Techniques like the Hadamard Test [38] aid quantum state verification, and QAE [37] improves probabilistic inference for financial modeling. Advances in Quantum Matrix Multiplication [56] and variance estimation [57] further extend computational capabilities, demonstrating the growing impact of quantum computing in data processing and machine learning.

3.6 Machine Learning

QML algorithms aim to enhance classical machine learning techniques by exploiting quantum parallelism. Quantum neural networks [15] and quantum support vector machines (QSVMs) [22] are notable examples, offering potential speedups in pattern recognition and data classification tasks. Quantum principal component analysis (QPCA) [21] further exemplifies the application of quantum algorithms in data compression and feature extraction. These algorithms are poised to revolutionize fields such as artificial intelligence, financial modeling, and image processing.

3.7 Cryptography

Quantum cryptography leverages quantum mechanics to enhance security in communication systems. Quantum Key Distribution (QKD) [23][18][58][17], based on the principles of superposition and no-cloning, ensures secure key exchange even in the presence of eavesdroppers. Lattice-based cryptography [59] [60], resistant to quantum attacks, represents a post-quantum cryptographic approach that addresses vulnerabilities exposed by algorithms like Shor's. These advancements in cryptography are pivotal in safeguarding information in the quantum era.

4 Analysis of key Quantum Algorithms

4.1 Search Algorithms

To our best knowledge, the current **generalized workflow of Search Algorithms** which have been implemented on **gate-based (digital) quantum computers** is the following:

1. Start with an initial state which is either:
   - Maximum superposition of all states.
   - Problem-specific initial state
2. Construct an **operator pair**. The Operator Pair are in an abstract sense two operators (or) oracles which perform the following operations:
   (a) Mixer Operator - An operator which takes a state which is in computational basis to a different basis with more superposition in the new basis states, like transforming the state into Hadamard state from computational basis state.
   (b) Driver Operator - An operator which drives the state to move towards the solution state. The initial problem is usually encoded here as a Hamiltonian, which is exponentiated to create this unitary gate.

   For certain algorithms, the above repeated set of operator pairs is parameterized, e.g., QAOA.
3. Determine the number of repetitions required to arrive at the result state, and apply the operator pair for that many repetitions. Eg. In Grover's Algorithm, given there are N quantum states in which M are winner states and $M << N$, the operator pair must be iteratively acted on the system for roughly 'I' iterations, where $I \approx \frac{\pi}{4}\sqrt{\frac{N}{M}}$. For certain QAOA algorithms, the number of iterations is fixed to a heuristic number. One can move to step 5 from here if the operator pairs are not parameterized.
4. (Only for parameterized ansatze or Operator pairs) The parameterized operator pairs in QAOA are classically optimized to move towards the solution state asymptotically, by measuring the quantum circuit to obtain the cost function value. Once the pre-defined convergence limit is reached the classical optimization process is taken to be completed.
5. Measurement of the Quantum System: The quantum circuit is finally measured to obtain a set of computational basis-vectors with different probabilities. The basis vector with the highest probability is taken to be the solution state. This step is final in the case of non-parameterized algorithms, otherwise, step 4 (classical optimization) and step 5(quantum measurement) are performed iteratively until convergence.

The above generalized workflow (Figure 5) is applicable to the following quantum algorithms and provided with them are the associated operator pairs:

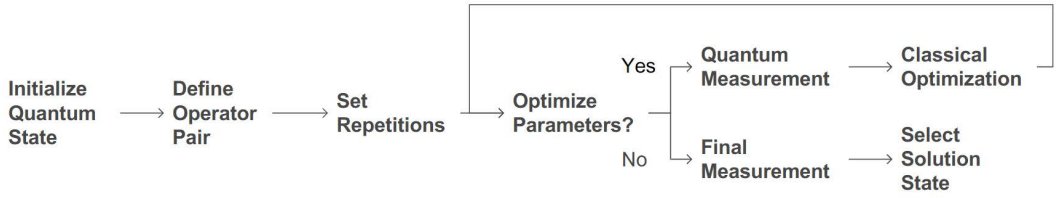

Figure 5: Generalized Workflow for Current Quantum Search Algorithms working on gate-based quantum computers

Table 2: Operator(Oracle) Pairs in Grover's Algorithm, Quantum Walk Algorithm, and QAOA

Algorithm	**Mixer Operator**	**Driver Operator**
Grover's Algorithm [4]	Diffusion Operator	Inversion Oracle
Quantum Walk Algorithm [61]	Coin Operator	Shift Operator
QAOA [14]	Mixer Hamiltonian	Cost Hamiltonian

Aside from QAOA (which can be considered to be the NISQ version of search algorithm, owing to its hybrid quantum-classical nature), the other search algorithms do not require parameter optimization. The interplay between Grover's Search ALgorithm, quantum walks (and QAOA) highlights the broader theme of quantum information processing: leveraging interference and amplitude amplification to achieve computational speedups beyond classical limits. A comparison of the operator(oracle) pairs is provided in Table 2

4.1.1 Quantum Walks

Introduction to Quantum Walks (History and Origin)

Quantum walks, introduced as the quantum counterpart of classical random walks, have their roots in the foundational principles of quantum mechanics and computational theory. The idea of quantum walks was formalized in 1993 by Aharonov et al. [62], although earlier conceptual links can be traced to Feynman's pioneering work on quantum simulations [8]. Quantum walks differ fundamentally from classical random walks by leveraging the principles of superposition and interference. These characteristics allow quantum walks to exhibit faster mixing times, enhanced spreading rates, and greater algorithmic power.

The initial exploration of quantum walks was motivated by their potential to surpass classical algorithms in specific computational problems. Over the years, quantum walks have evolved into an indispensable tool for constructing quantum algorithms and for modeling physical phenomena in various domains, including quantum computing and quantum biology. Two primary models of quantum walks, discrete-time and continuous-time, have been extensively studied, each offering unique insights and applications.

The theory of quantum walks encompasses two main frameworks: discrete-time and continuous-time models. The evolution of the system is governed by unitary operations that act on the coin and walker states in discrete-time quantum walks. The state of the system at time $t + 1$ is given by:

$$|\psi(t+1)\rangle = \hat{U}|\psi(t)\rangle, \tag{1}$$

where $\hat{U} = \hat{S}(\hat{C} \otimes \mathbb{I})$ is the evolution operator. Here, $\hat{C}$ is the **coin operator**, which determines the superposition of the coin state, and $\hat{S}$ is the **shift operator** that moves the walker based on the coin state. The Hadamard coin operator, commonly used, is represented as:

$$\hat{C} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}. \tag{2}$$

For continuous-time quantum walks, the system evolves under the time-dependent Schrödinger equation:

$$i\hbar \frac{\partial |\psi(t)\rangle}{\partial t} = \hat{H}|\psi(t)\rangle, \tag{3}$$

where $\hat{H}$ is the Hamiltonian that encodes the structure of the underlying graph or lattice. The continuous-time framework eliminates the need for a coin operator, as the evolution is determined solely by the Hamiltonian.

These mathematical foundations highlight the differences between quantum walks and classical random walks, showcasing the advantages of quantum interference and coherence in algorithmic and physical contexts.

The implementation of a discrete quantum walk involves a sequence of well-defined steps. First, the system is initialized with a walker and a coin in their respective Hilbert spaces:

$$|\psi(0)\rangle = |\text{position}\rangle \otimes |\text{coin}\rangle. \tag{4}$$

Next, the coin operator is applied to create a superposition of states. The Hadamard operator, for instance, acts on the coin state to produce:

$$\hat{C}|c\rangle = \frac{1}{\sqrt{2}}(|0\rangle + (-1)^c|1\rangle). \tag{5}$$

The shift operator then moves the walker based on the coin state:

$$\hat{S}|x, c\rangle = \begin{cases} |x+1, c\rangle & \text{if } c = 0, \\ |x-1, c\rangle & \text{if } c = 1. \end{cases} \tag{6}$$

The combined evolution operator $\hat{U}$ is applied iteratively, producing interference patterns in the probability distribution of the walker. After a sufficient number of steps, measurements are performed to extract the final state of the system.

4.1.2 Grover's Algorithm

Grover's algorithm, introduced by Lov Grover in 1996, is one of the foundational quantum algorithms that demonstrates a significant speedup over classical counterparts for unstructured search problems [4]. The algorithm provides a quadratic speedup compared to classical search methods, reducing the number of queries from $O(N)$ to $O(\sqrt{N})$ for an unsorted database of size N, where $N = 2^n$. This advantage makes it particularly relevant in cryptographic applications, where brute-force attacks can be significantly accelerated [63]. Unlike Shor's algorithm, which efficiently factors large numbers using quantum Fourier transforms, Grover's algorithm applies to a broader class of problems, particularly those expressible as an oracle-based search. Over the years, it has been extended and

generalized in various ways, including applications in optimization and database search [37].

The algorithm operates by iteratively amplifying the probability amplitude of the target solution using two key operations: the **Inversion Oracle** and the **Grover diffusion operator**. The *inversion oracle* is a quantum subroutine that marks the correct solution by applying a phase shift of -1, represented as U_f such that $U_f |x\rangle = (-1)^{f(x)} |x\rangle$, where f(x) = 0, if x is not marked, otherwise f(x)=1. The *diffusion operator* is a reflection about the mean, implemented as:

$$\hat{D} = H^{\otimes n}(2 |0\rangle \langle 0| - I)H^{\otimes n} = 2|\psi\rangle\langle\psi| - I, |\psi\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle \tag{7}$$

where $|\psi\rangle$ is the superposition state with uniform probability amplitudes, and H is the Hadamard gate. The combined effect of the oracle and diffusion operations gradually increases the amplitude of the marked state while suppressing others. After approximately $O(\sqrt{N})$ iterations, the probability of measuring the correct state is maximized, yielding the desired result with high probability [1]. A recent work[64] by Shukla et al., has extended the Grover's algorithm for arbitrary search space size.

A deeper perspective on Grover's algorithm emerges when viewed through the lens of quantum walks. Quantum walks, both discrete and continuous, serve as the quantum analogs of classical random walks and form the basis for various quantum search algorithms. Grover's search can be interpreted as a repetitive two-step quantum walk on a complete graph, where the oracle acts as a selective phase inversion, and the diffusion operator corresponds to a global mixing step [65]. This connection has led to further generalizations, such as spatial search on graphs and quantum walk-based search algorithms that outperform Grover's in specific cases [66].

4.2 Quantum Optimization Algorithms

4.2.1 Quantum Approximate Optimization Algorithm

The Quantum Approximate Optimization Algorithm is a variational hybrid quantum-classical algorithm designed to solve combinatorial optimization problems. It operates by encoding the problem into a cost Hamiltonian, $\hat{H}_C$, and driving the system through a series of parameterized quantum gates that optimize the solution space.

The algorithm starts by initializing a quantum state, typically the equal superposition state $|+\rangle^{\otimes n}$. The state evolves through alternating applications of two unitary operators derived from the problem's cost and mixer Hamiltonians. The cost Hamiltonian is constructed to encode the problem's objective function, while the mixer Hamiltonian facilitates exploration of the solution space. For a given depth p, the QAOA evolution is given by:

$$U(\gamma, \beta) = e^{-i\beta_p \hat{H}_M} e^{-i\gamma_p \hat{H}_C} \cdots e^{-i\beta_1 \hat{H}_M} e^{-i\gamma_1 \hat{H}_C}. \tag{8}$$

Here, $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_p)$ and $\beta = (\beta_1, \beta_2, \ldots, \beta_p)$ are parameters that define the unitary evolution. These parameters are optimized using a classical optimizer to maximize the objective function:

$$F(\gamma, \beta) = \langle \psi(\gamma, \beta)| \hat{H}_C |\psi(\gamma, \beta)\rangle . \tag{9}$$

The variational nature of QAOA allows it to leverage the hybrid quantum-classical paradigm effectively. The quantum circuit generates candidate solutions, and the classical optimizer iteratively adjusts the parameters γ and β to improve the cost function. As

p increases, the QAOA is expected to provide better approximations to the optimal solution. The QAOA is particularly well-suited for problems representable in the Quadratic Unconstrained Binary Optimization (QUBO) form, including MaxCut, graph coloring, and other NP-hard problems.

The choice of the cost and mixer Hamiltonians is crucial. The cost Hamiltonian, $\hat{H}_C$, encodes the problem, typically represented as:

$$\hat{H}_C = \sum_i h_i \sigma_{z_i} + \sum_{i<j} J_{ij} \sigma_{z_i} \sigma_{z_j}, \tag{10}$$

where σ_{z_i} are Pauli-Z operators, h_i represents local biases, and J_{ij} are interaction coefficients. The mixer Hamiltonian, $\hat{H}_M$, is often chosen as:

$$\hat{H}_M = \sum_i \sigma_{x_i}, \tag{11}$$

where σ_{x_i} are Pauli-X operators. This mixer facilitates transitions between basis states, ensuring a thorough exploration of the solution space.

The generalized workflow of QAOA [67], also illustrated in Figure 6, is the following:

1. **Initialization:** Start with a problem encoded in a cost Hamiltonian $\hat{H}_C$, which represents the objective function to be optimized, and a Hamiltonian mix $\hat{H}_M$, which facilitates the exploration of the solution space.

   (a) Initialize the quantum state $|\psi_0\rangle = |+\rangle^{\otimes n}$, which is an equal superposition of all the basis states.

   (b) Choose a circuit depth p, and initialize the variational parameters $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_p)$ and $\beta = (\beta_1, \beta_2, \ldots, \beta_p)$.

2. **Parametrized Quantum Evolution:**

   (a) Apply the Hamiltonian unitary cost $e^{-i\gamma_k \hat{H}_C}$ for the current iteration k, followed by the Hamiltonian unitary mixer $e^{-i\beta_k \hat{H}_M}$, where $k = 1, 2, \ldots, p$. This alternation produces the parameterized state:

   $$|\psi(\gamma, \beta)\rangle = e^{-i\beta_p \hat{H}_M} e^{-i\gamma_p \hat{H}_C} \cdots e^{-i\beta_1 \hat{H}_M} e^{-i\gamma_1 \hat{H}_C} |\psi_0\rangle . \tag{12}$$

   (b) At the end of the evolution, the state $|\psi(\gamma, \beta)\rangle$ represents the current approximation of the solution to the optimization problem.

3. **Loss Function Evaluation:**

   (a) Measure the state $|\psi(\gamma, \beta)\rangle$ on the computational basis to estimate the expectation value of the cost Hamiltonian:

   $$F(\gamma, \beta) = \langle \psi(\gamma, \beta)| \hat{H}_C |\psi(\gamma, \beta)\rangle . \tag{13}$$

   This procedure consists of measuring the expectation values of each Pauli string in $\hat{H}_C$ and combining them according to their corresponding coefficients.

4. **Classical Optimization:**

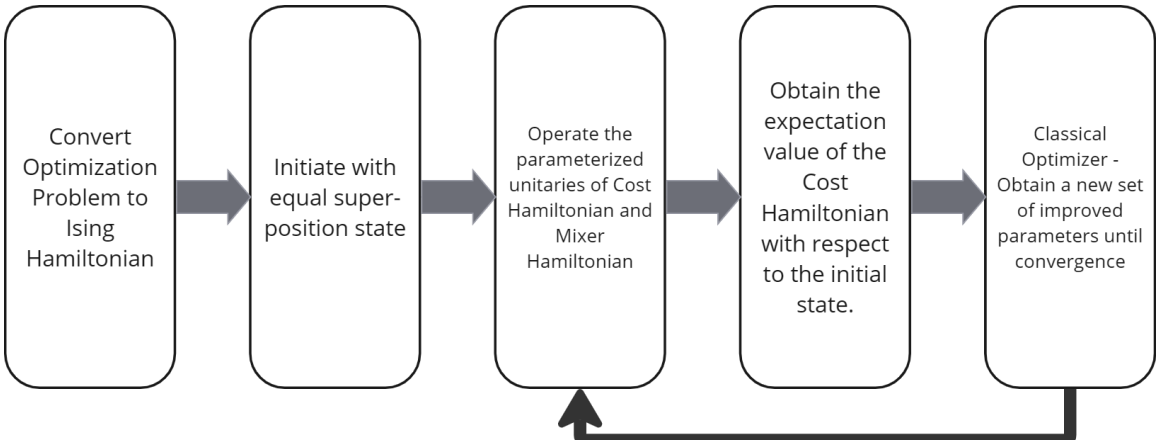

Figure 6: Simplified Workflow of QAOA Algorithm

(a) Pass $F(\gamma, \beta)$ to a classical optimization algorithm (e.g., Nelder-Mead, gradient-based methods) to update the parameters γ and β for the next iteration.

(b) Use the updated parameters to prepare a new parameterized state $|\psi(\gamma', \beta')\rangle$ and repeat the quantum-classical loop until convergence or a predefined stopping criterion is met.

5. **Result:** After convergence, the state $|\psi(\gamma^*, \beta^*)\rangle$ and the corresponding bitstring with the lowest energy represent the approximate solution to the optimization problem.

QAOA performance depends on various factors, including depth p, the quality of the classical optimizer, and the fidelity of the quantum hardware. While it is computationally expensive to simulate QAOA for large systems on classical computers, the algorithm's scalability on quantum devices offers a path toward solving challenging optimization problems more efficiently than classical methods.

Variants of QAOA: Many QAOA variants have been developed to address the limitations of the original algorithm, improving its efficiency, adaptability, and robustness. Table 3 summarizes the prominent variants and their enhancements.

Each variant addresses specific challenges. For instance, Digitized Counterdiabatic(DC)-QAOA reduces circuit depth through counterdiabatic driving [69], while Multi Angle(MA)-QAOA improves approximation ratios for graph-based problems [68]. Warm-started QAOA variants, such as WS-QAOA [71], leverage classical relaxation techniques for better initialization, retaining guarantees like the Goemans-Williamson bound [73].

4.2.2 Quantum Annealing

Quantum annealing (QA) is a heuristic optimization algorithm grounded in the *adiabatic theorem* of quantum mechanics. The theorem states that a quantum system prepared in the ground state of an initial Hamiltonian, $\mathcal{H}_i$, will remain in the ground state if the system evolves slowly enough to a final Hamiltonian, $\mathcal{H}_f$, provided the evolution respects certain adiabatic conditions [74]. The time-dependent Hamiltonian of the system

Table 3: Prominent QAOA Variants and Applications

Variant	**Key Features**	**Applications**
MA-QAOA [68]	Introduces unique parameters for each element in cost and mixer Hamiltonians, reducing circuit depth.	MaxCut problems, improved approximation ratios.
DC-QAOA [69]	Incorporates counterdiabatic driving to speed up convergence.	Ising models, classical optimization problems.
ADAPT-QAOA [70]	Iteratively selects mixers based on gradient optimization.	Customizable ansatz for specific problems.
WS-QAOA [71]	Warm-starts parameters using solutions from relaxed problems.	Retains GW bound for MaxCut problems.
GM-QAOA [72]	Uses Grover-like selective mixers for constrained problems.	k-Vertex-Cover, Traveling Salesperson Problem.

is represented as:

$$\mathcal{H}(t) = A(t)\mathcal{H}_i + B(t)\mathcal{H}_f, \tag{14}$$

where $t \in [0, T_a]$ is the time, $A(t)$ and $B(t)$ are monotonic functions such that $A(0) = 1, B(0) = 0$ and $A(T_a) = 0, B(T_a) = 1$. The goal of QA is to find the ground state of $\mathcal{H}_f$, which encodes the optimal solution to a given problem.

The initial Hamiltonian, $\mathcal{H}_i$, is typically a transverse-field Hamiltonian:

$$\mathcal{H}_i = \sum_i \sigma_i^x, \tag{15}$$

where σ_i^x are the Pauli-X operators acting on qubits. The final Hamiltonian is represented in the Ising model form:

$$\mathcal{H}_f = \sum_i h_i \sigma_i^z + \sum_{i<j} J_{ij} \sigma_i^z \sigma_j^z, \tag{16}$$

where h_i and J_{ij} are problem-specific parameters, and σ_i^z are the Pauli-Z operators.

QA leverages quantum tunneling and entanglement to escape local minima by tunneling through high-energy barriers, which is particularly useful for combinatorial optimization problems. The evolution process involves gradual suppression of quantum fluctuations introduced by $\mathcal{H}_i$, thereby allowing the system to settle into the ground state of $\mathcal{H}_f$ [75].

4.2.3 Boson Sampling and Gaussian Boson Sampling

Sampling is a quantum computational model that leverages the interference of indistinguishable photons in a linear optical network to perform computations that are believed to be classically intractable. The output probability distribution in Boson Sampling is determined by the permanent of a submatrix derived from the network's unitary transformation. Since computing the permanent is #P-hard, Boson Sampling serves as a compelling demonstration of quantum computational advantage, particularly in the context of quantum supremacy. This model is particularly effective for proving the hardness of certain quantum problems, as it directly relates to the complexity of simulating quantum systems with classical computers [42].

Gaussian Boson Sampling (GBS) [76] extends the Boson Sampling framework by utilizing Gaussian states, such as squeezed vacuum states, which are more experimentally accessible. In **GBS**, the **output probability distribution** involves the **Hafnian of a matrix** derived from the covariance matrix of the Gaussian state. The Hafnian, like the permanent, is computationally hard, ensuring that GBS also demonstrates quantum computational advantage. However, GBS offers broader applicability due to the Hafnian's connection to general graph structures. Specifically, the Hafnian can be used to count perfect matchings in general graphs, which is relevant for a wide range of problems in quantum chemistry, such as simulating molecular vibronic spectra [77], and in combinatorial optimization, such as finding dense subgraphs [78]. Moreover, through specific matrix constructions, the Hafnian can also address problems typically solved using the permanent for bipartite graphs, thus encompassing a wider array of applications compared to standard Boson Sampling.

In traditional BS, the system is initialized with single-photon Fock states injected into an m-mode interferometer, and the sampling problem involves computing output distributions governed by the permanent of a submatrix of the unitary transformation U representing the network. The probability of detecting a photon configuration $S = (s_1, \ldots, s_m)$ is given by:

$$P(S) = \frac{|\mathrm{Per}(U_S)|^2}{\prod_{i=1}^{m} s_i!},$$

where $\mathrm{Per}(U_S)$ is the matrix permanent of the submatrix U_S obtained by choosing rows and columns based on the input and output photon configurations [42, 79].

Gaussian Boson Sampling extends the BS framework by using Gaussian states, specifically squeezed vacuum states—as inputs instead of single-photon Fock states. This modification offers a more feasible experimental realization due to the relative ease of generating squeezed light. While the underlying optical network remains linear and unitary, the statistics of the output configurations are now determined by the Hafnian rather than the permanent. For a given photon-number pattern $\mathbf{k} = (k_1, k_2, \ldots, k_M)$ from a Gaussian state with covariance matrix σ, the probability $P(\mathbf{k})$ is:

$$P(\mathbf{k}) = \frac{\mathrm{Haf}(O)}{\sqrt{\det \sigma}\, k_1! k_2! \cdots k_M!} \tag{17}$$

where O is a matrix constructed from σ encoding pairwise correlations between modes [80, 76].

The core computational difference between BS and GBS lies in the complexity of their respective matrix functions. The permanent used in BS is known to be #P-hard to compute, even approximately, a complexity that underpins the classical intractability of simulating BS [42]. Similarly, the Hafnian function in GBS is computationally intensive, scaling exponentially with the size of the system, and encodes the pairing correlations unique to bosonic Gaussian states. The Hafnian of a symmetric $2n \times 2n$ matrix A is defined as:

$$\mathrm{Haf}(A) = \sum_{\mu \in \mathrm{PMP}} \prod_{j=1}^{n} A_{\mu(2j-1),\mu(2j)} \tag{18}$$

where PMP denotes the set of perfect matching permutations [81]. This difference emphasizes how GBS generalizes BS while preserving its complexity-theoretic hardness.

Despite their theoretical similarities, BS and GBS diverge significantly in terms of **experimental practicality** and **detection strategies**. BS demands indistinguishable

single photons and photon-number-resolving detectors, which are technologically challenging at scale. GBS, in contrast, can employ threshold detectors, with the Torontonian function replacing the Hafnian for probability calculations:

$$P(\mathbf{k}) = \frac{\mathrm{Tor}(O)}{\sqrt{\det \sigma}} \tag{19}$$

providing a simplification while retaining computational hardness [82]. Moreover, both BS and GBS has demonstrated practical advantages in real-world applications such as simulating molecular vibronic spectra [77], solving dense subgraph problems [78], and improving the scalability of quantum photonic experiments [83, 84].

4.3 Quantum Simulation Algorithms

Using a quantum computer to simulate natural systems would always be the most practical use of such computers, as the one-to-one analogy of the Hilbert Space of the natural system to the qubits in the quantum system allows linear scaling of the quantum resources required, as opposed to an exponential scaling of classical resources to perform the simulation to similar levels of accuracy.

1. **Initialization of Quantum State** : To begin with, the quantum system is initialized in a quantum state which is analogous to the solution of a quantum chemistry method. Usually the initial state is the solution obtained from Hartree-Fock method which is a single Slater Determinant or basis state.

2. **Circuit Preparation** : In this step, the quantum circuit which will drive the initial state towards the expected state is prepared. In the case of FTQC algorithms like QPE [35] and Linear Combination of Unitaries [45], the process of encoding the information of the Hamiltonian into the quantum circuit (acting the unitaries), and in the case of NISQ-era algorithms like VQE [13] and SQD [19], this process is the creation of the parameterized ansatze.

3. **Basis Transformation and (or) Quantum Measurement**: Prior to the quantum measurement, for some algorithms either the final result is stored in a different basis (Fourier Basis in the case of QPE) or the quantum measurement is to be done in a different basis (Pauli basis measurement, in the case of algorithms like VQE and PQE). Hence, the appropriate basis transformation is performed and then the quantum system is measured.

4. **Optional Classical optimization** : In the case of variational algorithms, the parameters of the ansatze are treated as input and the result from the quantum measurement (expectation value) is treated as the output of a cost function. The output of this cost function is iteratively minimized by tweaking the parameters of the ansatze.

To obtain the Pauli Hamiltonian from molecular geometry data suitable for quantum computers, a series of steps are required. These steps are quintessential for any simulation algorithm.

Starting with the molecular geometry, defined by the Cartesian coordinates of the atomic nuclei, a Hartree-Fock calculation is performed to compute molecular orbitals and the one- and two-electron integrals, h_{pq} and h_{pqrs}, which describe electron interactions.

The molecular Hamiltonian is then expressed in second quantized form using creation and annihilation operators. This Hamiltonian is mapped to a qubit Hamiltonian via the Jordan-Wigner transformation [85], converting fermionic operators into Pauli operators.

The resulting Pauli Hamiltonian, a linear combination of Pauli operator tensor products, is optimized on a quantum computer using a variational approach, where a parameterized quantum state is prepared, and a classical optimizer iteratively updates the parameters to minimize the energy expectation value, typically to find the ground-state energy of the molecule.

After applying the Born-Oppenheimer approximation, the molecular Hamiltonian in second quantized form is written as:

$$\hat{H} = \sum_{i,j} h_{ij} \hat{a}_i^\dagger \hat{a}_j + \sum_{p,q,r,s} h_{pqrs} \hat{a}_p^\dagger \hat{a}_q^\dagger \hat{a}_r \hat{a}_s \tag{20}$$

where the coefficients h_{ij} and h_{pqrs} are calculated via a Hartree-Fock calculation:

$$\left.\begin{aligned} h_{pq} &= \int dr \chi_p(r)^* \left(-\frac{1}{2}\nabla^2 - \sum_\alpha \frac{Z_\alpha}{|r_\alpha - r|} \right) \chi_q(r) \\ h_{pqrs} &= \int dr_1 dr_2 \frac{\chi_p(r_1)^* \chi_q(r_2)^* \chi_r(r_1) \chi_s(r_2)}{|r_1 - r_2|} \end{aligned}\right\} \tag{21}$$

To simulate on a quantum computer, the Hamiltonian is mapped to a qubit Hamiltonian using transformations like Jordan-Wigner [85]:

$$\left.\begin{aligned} \hat{a}_i &\longrightarrow \bigotimes_{k=1}^{i-1} Z_k \otimes \left(\frac{X_i + iY_i}{2} \right) \bigotimes_{l=i+1}^{N-i} \mathcal{I}_l \\ \hat{a}_i^\dagger &\longrightarrow \bigotimes_{k=1}^{i-1} Z_k \otimes \left(\frac{X_i - iY_i}{2} \right) \bigotimes_{l=i+1}^{N-i} \mathcal{I}_l \end{aligned}\right\} \tag{22}$$

The outcome is a qubit Hamiltonian written as a sum of tensor-product Pauli operator terms, utilizing the Pauli operators $\hat{\sigma}_x$, $\hat{\sigma}_y$, $\hat{\sigma}_z$ and identity $\mathcal{I}$:

$$\hat{H} = \sum_i \otimes_{k=0}^{n-1} h_i \hat{P}_k^i \ ; \hat{P}_k \in \{\hat{\sigma}_x, \hat{\sigma}_y, \hat{\sigma}_z, \mathcal{I}\} \tag{23}$$

4.3.1 Variational Quantum Eigensolver

The Variational Quantum Eigensolver, proposed by Peruzzo et al. in 2014 [13], was introduced as a hybrid quantum-classical algorithm tailored to the limitations of NISQ devices. Unlike algorithms like QPE, which demand fault-tolerant quantum hardware, VQE efficiently operates on current quantum processors by combining quantum state preparation with classical optimization.

At its core, VQE relies on the variational theorem(see Sec.2.2.1), ensuring that the expected energy of any trial state provides an upper bound to the true ground-state energy. The quantum computer prepares a parametrized quantum state (also called an **ansatz**), and the classical optimizer iterates over the parameter values, updating them each time to reach the energy expectation value.

As outlined in [13], the workflow(Figure 7) consists of the following steps:

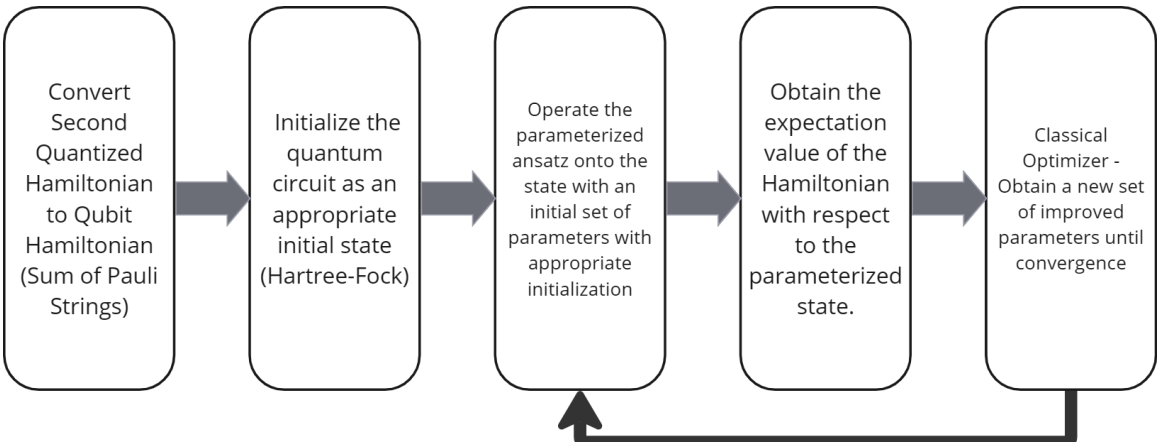

Figure 7: Simple Flowchart for VQE

1. Design a parameterized quantum circuit (ansatz) using unitaries $\hat{U}_j(\theta_j)$ controlled by parameters $\{\theta_j\}$ to prepare the trial state:

$$\left|\psi\{\theta_j^n\}\right\rangle = \hat{U}_j(\theta_j^n)\left|0\right\rangle^{\otimes n}$$

The loss function (energy expectation) is:

$$J(\{\theta_j^n\}) = \left\langle\psi(\{\theta_j^n\})\right| H \left|\psi(\{\theta_j^n\})\right\rangle \tag{24}$$

2. Iterate until convergence:

   (a) Use the **Quantum Expectation Estimation algorithm** to compute the energy by measuring the expectation values of each Pauli string and summing them using 23:

$$J(\{\theta_j^n\}) = \sum_i h_i \left\langle\psi\{\theta_j^n\}\right| \otimes_k \hat{P}_k^i \left|\psi\{\theta_j^n\}\right\rangle \tag{25}$$

   (b) **Classical Minimization:** Feed the computed loss to a classical optimizer to obtain updated parameters $\{\theta_j^{n+1}\}$.

The choice of ansatz is a critical determinant of the performance and scalability of variational quantum algorithms, as it directly influences factors such as circuit depth, expressibility, and the efficiency of Hilbert space exploration. The **Hardware-Efficient(HEA) Ansatz** [86] leverages the native gate set of the underlying quantum hardware, enabling high expressibility and straightforward implementation tailored to specific architectures. In contrast, problem-inspired constructions such as the **Unitary Coupled Cluster Singles and Doubles (UCCSD) ansatz** [87], derived from Coupled Cluster Theory [88], employ chemically/physically motivated excitation operators to capture electron correlations more systematically. More recent developments, such as the **Local Unitary Cluster Jastrow (LUCJ) ansatz** [89], seek to integrate hardware efficiency with chemical intuition, achieving improved accuracy for correlated electronic systems in a hardware-compatible manner. A comprehensive review of the theoretical and practical aspects of VQE can be found in [90].

4.3.2 Projective Quantum Eigensolver

The Projective Quantum Eigensolver (PQE)[20] works by constructing a parameterized trial wavefunction $|\Psi(\boldsymbol{\theta})\rangle$ through the action of a unitary ansatz $\hat{U}(\boldsymbol{\theta})$ on a reference state, typically the Hartree-Fock state $|\phi_0\rangle$. Specifically, $|\Psi(\boldsymbol{\theta})\rangle = \hat{U}(\boldsymbol{\theta})\,|\phi_0\rangle$ where the ansatz is often chosen as the disentangled Unitary Coupled Cluster (dUCC) form:$\hat{U}(\boldsymbol{\theta}) = \prod_\mu e^{\hat{\kappa}_\mu(\theta_\mu)}$ with $(\hat{\kappa}_\mu(\theta_\mu) = \hat{\tau}_\mu(\theta_\mu) - \hat{\tau}_\mu^\dagger(\theta_\mu))$ being an anti-Hermitian cluster operator. PQE then projects the Schrödinger equation onto a basis of excited determinants to minimize the residuals:

$$r_\mu(\boldsymbol{\theta}) \equiv \langle\phi_\mu|\,\hat{U}^\dagger(\boldsymbol{\theta})\hat{H}\hat{U}(\boldsymbol{\theta})\,|\phi_0\rangle \tag{26}$$

where (r_μ) is the residual element that quantifies the deviation from the exact eigenstate. These residuals are minimized iteratively using a coupled-cluster-like quasi-Newton technique until convergence is achieved.

While both PQE and VQE are hybrid quantum-classical algorithms used for finding the ground state energy of quantum systems, they differ significantly in their approach. VQE employs a variational approach by parameterizing a trial wavefunction and minimizing the energy expectation value, $E(\boldsymbol{\theta}) = \langle\phi_0|\,\hat{U}^\dagger(\vec{\theta})\hat{H}\hat{U}(\vec{\theta})\,|\phi_0\rangle$ using classical optimizers. In contrast, PQE utilizes a projective strategy to minimize the residuals associated with the parameterized ansatz, which can lead to faster convergence due to its coupled-cluster-like optimization. PQE is generally more robust to noise compared to VQE as it requires fewer gradient evaluations, which is particularly beneficial on NISQ devices. However, PQE necessitates the inclusion of higher-rank excitation operators for strongly correlated systems, leading to increased quantum resource utilization. This makes PQE potentially more resource-intensive than VQE for complex systems. Nonetheless, PQE often demonstrates superior noise resilience and faster convergence compared to VQE under equivalent parameterizations.

4.3.3 Quantum Imaginary Time Evolution

Quantum Imaginary Time Evolution (QITE) [91] is a powerful technique for simulating quantum systems and finding ground states of Hamiltonians. Unlike real-time evolution, which follows the unitary dynamics dictated by the Schrödinger equation, imaginary time evolution transforms a quantum state towards its ground state by evolving according to a non-unitary process. This approach is particularly useful in variational quantum algorithms and near-term quantum computing applications.

The imaginary time evolution of a quantum state is governed by the equation

$$\frac{d}{d\tau}\,|\psi(\tau)\rangle = (\hat{H} - \langle H(\tau)\rangle)\,|\psi(\tau)\rangle \tag{27}$$

where τ represents imaginary time and H is the system Hamiltonian. The formal solution to this equation [91] is given by

$$\psi(\tau) = A(\tau)e^{-\hat{H}\tau}\,|\psi(0)\rangle\,; A(\tau) = \frac{1}{\sqrt{\langle\psi(0)|\,e^{-2\hat{H}\tau}\,|\psi(0)\rangle}} \tag{28}$$

This evolution suppresses the excited-state components of the initial state, leading to convergence towards the ground state as $\tau \to \infty$.

In practical implementations, QITE is often approximated using a series of short imaginary time steps $\Delta\tau$. A discrete approximation can be expressed as

$$|\psi(\tau + \Delta\tau)\rangle \approx \frac{e^{-H\Delta\tau}|\psi(\tau)\rangle}{\|e^{-H\Delta\tau}|\psi(\tau)\rangle\|}. \tag{29}$$

Since direct application of non-unitary evolution is not feasible on quantum hardware, an effective unitary transformation is approximated using variational techniques or ancilla-assisted methods.

QITE plays a crucial role in quantum algorithms such as the Variational Quantum Imaginary Time Evolution (VQITE) [92] method, where optimization is performed over the parameterized quantum circuit to approximate the evolution. The evolution is enforced by updating circuit parameters iteratively, following an effective classical optimization step. A key advantage of QITE over traditional VQE is its ability to avoid barren plateaus by naturally guiding the state towards the desired eigenstate [50].

4.3.4 Sample-based Quantum Diagonalization

Sample-based Quantum Diagonalization (SQD)[19] is a quantum subspace method designed to leverage quantum circuits and classical computation for solving the Schrödinger equation. The method constructs a subspace spanned by computational basis states sampled from a quantum circuit:

$$\hat{H}_{S^{(b)}} = \hat{P}_{S^{(b)}} \hat{H} \hat{P}_{S^{(b)}}, \tag{30}$$

where the projector $\hat{P}_{S^{(b)}}$ is:

$$\hat{P}_{S^{(b)}} = \sum_{\mathbf{x} \in S^{(b)}} |\mathbf{x}\rangle\langle\mathbf{x}|. \tag{31}$$

These basis states, sampled according to a distribution derived from the quantum circuit, represent Slater determinants in the standard Jordan-Wigner mapping. Efficient computation of matrix elements using the Slater-Condon rules allows for diagonalization within the subspace.

The wavefunction is updated iteratively using:

$$n_{p\sigma} = \frac{1}{K} \sum_{b=1}^{K} \langle\psi^{(b)}|\hat{n}_{p\sigma}|\psi^{(b)}\rangle, \tag{32}$$

where $n_{p\sigma}$ is the occupation number distribution.

The algorithmic workflow of SQD proceeds as follows. First, a trial quantum circuit $|\Phi(\theta_{\text{ini}})\rangle$ is prepared. Then, computational basis states $\{\mathbf{x}_1, \ldots, \mathbf{x}_d\}$ are sampled from the quantum state $|\Phi_{\text{qc}}\rangle$. The basis states which respect the number and spin symmetry are retained, and the erroneous states are treated with an error mitigation technique called **Self-Consistent Configuration Recovery**.Using the set of states obtained after Configuration Recovery (along with the symmetry-maintaining basis states), the Hamiltonian is projected onto the subspace spanned by the sample set, yielding the projected Hamiltonian $\hat{H}_{S^{(b)}} = \hat{P}_{S^{(b)}} \hat{H} \hat{P}_{S^{(b)}}$. This projected Hamiltonian is then diagonalized to obtain eigenvalues $E^{(b)}$ and eigenstates $|\psi^{(b)}\rangle$ for each batch b. The algorithm proceeds iteratively, updating the occupation number distribution and checking for convergence. Finally, the minimum energy across all batches is selected as the SQD estimate: $E_{\text{SQD}} = \min_b E^{(b)}$.

The SQD method offers significant improvements over the variational algorithms by increasing the **efficiency in sampling** because SQD uses a predefined quantum circuit to sample basis states, reducing the optimization burden present in VQE. SQD has higher **noise resilience** because it focuses only on the subspace defined by sampled states, hence SQD minimizes the impact of noise compared to full variational optimization. One other added benefit of SQD is the **flexibility in subspace definition** because SQD allows systematic control over the subspace size d, enhancing its adaptability to specific problems.

4.3.5 Quantum Phase Estimation

Using the framework of Quantum Fourier transform [35](see Sec.2.2.4) as a sub-routine, one of the first algorithmic sub-routines which was given a proper use case by Shor [12], for solving prime factorization, QPE still remains a top contender in the field of FTQC algorithms.

QPE is a cornerstone algorithm in quantum computing, enabling the estimation of the phase (or eigenvalue) of a unitary operator's eigenvector with high precision. QPE underlies multiple quantum algorithms, such as Shor's algorithm and simulation algorithms.

We consider U, an unitary operator acting on a Hilbert space $\mathcal{H}$, with an eigenvector $|\psi\rangle$ such that

$$U|\psi\rangle = e^{i2\pi\phi}|\psi\rangle\,, \tag{33}$$

where $\phi \in [0, 1)$ is the phase to be estimated. In our case, the unitary U is constructed in the following manner, $U = e^{-i\hat{H}t}$ The goal of QPE is to produce an approximation of ϕ with precision determined by the number of ancilla qubits.

The QPE algorithm employs two registers:

- **Ancilla register**: n qubits initialized to $|0\rangle^{\otimes n}$, used to store the phase estimate.
- **System register**: Initialized to the eigenstate $|\psi\rangle$, on which U acts.

The output is an n-bit approximation of ϕ, ideally yielding $\tilde{\phi}$ such that $|\phi - \tilde{\phi}| \leq 2^{-n}$ with high probability.

The QPE circuit consists of three main stages: initialization, controlled unitary application, and inverse QFT.

The ancilla register is prepared in the uniform superposition state using Hadamard gates:

$$|0\rangle^{\otimes n} \xrightarrow{H^{\otimes n}} \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle\,, \tag{34}$$

where $|k\rangle$ denotes the computational basis state corresponding to integer k in binary. The system register is initialized to $|\psi\rangle$, so the total state is

$$|\Psi_0\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle|\psi\rangle\,. \tag{35}$$

For each ancilla qubit j (indexed from 0 to $n-1$), apply the controlled unitary U^{2^j}, where the ancilla qubit $|x_j\rangle$ controls whether U^{2^j} is applied to $|\psi\rangle$. Since $|\psi\rangle$ is an eigenstate,

$$U^{2^j}|\psi\rangle = e^{i2\pi\phi 2^j}|\psi\rangle\,. \tag{36}$$

The controlled operation on state $|k\rangle |\psi\rangle$, where $k = \sum_{j=0}^{n-1} k_j 2^j$, yields:

$$\prod_{j=0}^{n-1} \text{C-}U^{2^j} |k_j\rangle |\psi\rangle = |k\rangle \otimes e^{i2\pi\phi \sum_{j=0}^{n-1} k_j 2^j} |\psi\rangle . \tag{37}$$

Thus, the total state becomes:

$$|\Psi_1\rangle = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} e^{i2\pi\phi k} |k\rangle |\psi\rangle . \tag{38}$$

This resembles the quantum Fourier transform (QFT) of a state encoding ϕ.

Performing IQFT on the above state, and subsequent measurement will yield the phase of the unitary (or in other words the eigen-value of the Hamiltonian) in binary base.

4.3.6 Trotterization

Trotterization is a fundamental technique used for approximating the time evolution operator of a quantum system governed by a Hamiltonian. Given a Hamiltonian H that is decomposed into a sum of non-commuting terms,$H = \sum_j H_j$, its time evolution is given by the unitary operator:$U(t) = e^{-iHt}$. It has been used as a **sub-routine in QPE** for approximation of the time-evolution operator.

Since the individual terms H_j do not necessarily commute (in the event that the individual terms do commute, one uses the *product formula* to directly act the product of the unitaries on the quantum system), direct exponentiation is computationally challenging. The Trotter-Suzuki formula approximates the evolution by breaking it into small time steps:

$$U(t) \approx \left(\prod_j e^{-iH_j t/n} \right)^n \tag{39}$$

where n is the number of Trotter steps. As $n \to \infty$, the approximation becomes exact [93].

Trotterization is widely used in quantum simulation for efficiently simulating molecular systems and condensed matter physics [94]. It also plays a fundamental role in quantum computation by serving as the basis for variational quantum algorithms and hybrid quantum-classical methods. Additionally, it contributes to error mitigation by enabling controlled approximations that help reduce errors in quantum gate implementations [95].

Despite its effectiveness, the Trotter method introduces an approximation error, which scales with $\mathcal{O}(t^2/n)$. Higher-order Suzuki expansions can improve accuracy but require additional quantum gates [93].

4.3.7 Linear Combination of Unitaries

The Linear Combination of Unitaries (LCU) technique is a fundamental quantum algorithmic primitive that enables the efficient realization of non-unitary operators by expressing them in terms of a collection of unitary matrices[45], and can also be used as a **sub-routine with QPE** for efficient implementation of the time-evolution operator. This method has been widely used in quantum algorithms for Hamiltonian simulation, quantum linear algebra, and solving differential equations [40].

The LCU technique expresses a target operation A as a weighted sum of unitary matrices:

$$A = \sum_k \alpha_k U_k, \tag{40}$$

where α_k are complex coefficients, and U_k are unitary matrices. The challenge is to implement this non-unitary A using quantum operations.

To achieve this, an ancilla qubit system is introduced to encode the coefficients α_k in a superposition state:

$$\sum_k \sqrt{p_k} \left|k\right\rangle \otimes U_k \left|\psi\right\rangle, \tag{41}$$

where $p_k = \frac{|\alpha_k|}{\lambda}$ and $\lambda = \sum_k |\alpha_k|$ ensures normalization. By performing a controlled unitary operation and post-selecting on the correct ancilla measurement outcome, the operation A is effectively applied [96].

The Linear Combination of Unitaries (LCU) method has found applications in a variety of quantum algorithms. In **Hamiltonian simulation**, LCU facilitates efficient time evolution of quantum states by decomposing a Hamiltonian into a sum of unitaries [97]. It is also employed in **quantum linear system solvers** such as the HHL algorithm, enabling exponential speedup over classical approaches for solving linear systems [39]. Furthermore, in the context of quantum signal processing, LCU provides a systematic framework for designing quantum circuits that implement a broad range of transformations [40].

As quantum technology advances, the LCU framework is expected to play a critical role in unlocking new computational advantages.

4.4 Factorization - Shor's Algorithm

Shor's algorithm, proposed by Peter Shor in 1994 [36], is one of the most significant breakthroughs in quantum computing. It efficiently factors large integers, a problem that is intractable for classical computers due to its super-polynomial complexity. The algorithm demonstrates exponential speedup over the best-known classical factoring algorithms such as the General Number Field Sieve (GNFS) [98].

At its core, Shor's algorithm relies on the QFT[35] to find the period of a modular function. Given an integer N to be factored, the algorithm chooses a random integer $a < N$ and computes the order r, which is the smallest integer satisfying:

$$a^r \equiv 1 \mod N. \tag{42}$$

Using quantum parallelism and QFT, the algorithm efficiently determines r, which is then used to find a nontrivial factor of N. If r is even, one computes:

$$\gcd(a^{r/2} - 1, N) \tag{43}$$

which, with high probability, yields a factor of N.

The power of Shor's algorithm lies in its quantum subroutine, which estimates the period of a function using QFT. This component is exponentially faster than classical approaches based on trial division or sieving techniques[36].

The impact of Shor's algorithm extends beyond number theory. It threatens widely used cryptographic systems, such as RSA encryption [99], which rely on the difficulty of integer factorization. The ability of quantum computers to efficiently solve this problem has driven the search for post-quantum cryptographic schemes [100].

While Shor's algorithm is theoretically sound, its practical implementation depends on large-scale FTQCs. Current quantum devices are not yet capable of factoring large numbers relevant to cryptographic applications, but ongoing advancements in quantum hardware bring this closer to reality [101].

4.5 Quantum Arithmetic-based Algorithms

Quantum computing has introduced a revolutionary approach to solving problems in linear algebra and optimization. At the core of these advancements lie algorithms that exploit quantum mechanics to achieve exponential or polynomial speedups over classical methods. Many quantum algorithms rely on amplitude amplification, phase estimation, and block encoding techniques to efficiently process large-scale linear systems, eigenvalue problems, and matrix transformations. These techniques are fundamental in quantum linear algebra and play a crucial role in practical applications such as quantum recommendation systems and machine learning.

One of the most significant breakthroughs in quantum linear algebra is the **HHL Algorithm** [39], which provides an exponential speedup for solving linear systems compared to classical methods. The HHL algorithm builds upon the QPE algorithm [35], a fundamental subroutine used to approximate eigenvalues of a given unitary operator. Further developments in this field have led to the **Quantum Linear System Algorithm** (QLSA) [54], which generalizes HHL and improves its efficiency under certain conditions. Notably, the Adiabatic QLSA [55] and Optimal QLSA [102] have refined these approaches by leveraging adiabatic evolution and optimal precision estimation.

QSVT [43] provides a unified framework for manipulating matrix functions efficiently. This technique extends block encoding, which represents matrices within a larger unitary framework, enabling quantum algorithms to apply functions such as matrix inversion and exponentiation in a quantum setting. QSVT serves as a foundational tool for advanced quantum algorithms in optimization and machine learning. Additionally, techniques like the **Hadamard Test**[38] and **Hilbert-Schmidt Test** [103] facilitate inner product estimation and density matrix comparisons, which are essential for quantum state verification and QML applications.

Quantum Amplitude Estimation (QAE) [37] and its iterative variant, **Iterative Quantum Amplitude Estimation (IQAE)** [104], enhance probabilistic inference in quantum algorithms. These techniques are fundamental in finance, chemistry, and machine learning, where accurate estimation of expectation values is crucial. Quantum subroutines for variance estimation further extend these capabilities, allowing efficient risk analysis and uncertainty quantification in quantum simulations. Moreover, as a theoretical concept, **Quantum Random Access Memory (QRAM)** [105] enables efficient data retrieval in quantum algorithms, an essential requirement for large-scale quantum data processing.

Given such efficient methods are available for data encoding and data storage in the future, sub-routines can be devised to perform more complex tasks like **matrix multiplication**[56], **variance estimation** [57] and image processing[106]. Algorithms like **Quantum Singular Value Decomposer(QSVD)** [107] to perform diagonalization of two qubit bi-partite quantum states have also been proposed as an initial step towards the use of these quantum algorithms for real-life applications involving larger data.

These advancements collectively contribute to practical applications such as the **Quantum Recommendation Systems**[53], which utilizes quantum linear algebra techniques to provide faster and more efficient personalized recommendations.

Furthermore, **Bayesian QPE** [44] integrates Bayesian inference with phase estimation to achieve optimal precision in estimating eigenvalues of quantum operators. The synergy of these quantum techniques continues to drive research and development in quantum computing, promising novel solutions to complex computational challenges.

4.6 Quantum Machine Learning Algorithms

Based on the broader classifications of machine-learning algorithms, QML algorithms can also be classified into the following sections. Table 4 discusses some of the popular QML algorithms, their advantage over their classical counter-parts and their applications briefly.

4.6.1 Clustering Algorithms

Clustering falls under unsupervised machine learning. It aims to group similar data points into clusters. In QML, several quantum-enhanced clustering algorithms have been developed to exploit the unique properties of quantum computing. These methods often aim to leverage quantum speedups for high-dimensional data clustering or improve clustering quality via quantum-inspired representations. Key quantum approaches include quantum k-means, quantum hierarchical clustering, and quantum-assisted density-based clustering [108, 109].

The **quantum k-means algorithm**[110] is a prominent quantum adaptation of the classical k-means clustering technique. By utilizing quantum distance calculations and state preparation, the quantum k-means algorithm can achieve polynomial speedup over its classical analog for specific cases. Quantum superposition enables the calculation of the distance between data points and centroids, enabling simultaneous evaluation of multiple distances. Moreover, QAOA has been employed for clustering problems, particularly in graph-based clustering, where it optimizes the graph cut to form coherent clusters [111]. These advancements demonstrate that quantum computing could significantly enhance clustering efficiency for large datasets.

In addition to k-means, hybrid quantum-classical hierarchical clustering [112] leverages quantum principles such as amplitude amplification to optimize the construction of dendrograms. This approach can reduce the computational overhead associated with evaluating pairwise similarities, a bottleneck in classical hierarchical clustering. Meanwhile, density-based clustering algorithms like DBSCAN[113] benefit from quantum subroutines for nearest-neighbor searches, speeding up cluster formation for datasets with complex density structures. In addition to being used as sub-routine for DBSCAN, recently a quantum Mutual MinPts-nearest neighbor graph (MMNG)-based DBSCAN algorithm has also been proposed [114].

Quantum clustering algorithms represent an exciting frontier in QML. Although such methods demand reliable quantum hardware, their capacity for speedups and enhanced clustering performance positions them as promising directions for future research and practical deployment. As quantum technologies mature, these algorithms are expected to address challenges in fields ranging from bioinformatics to social network analysis [115].

4.6.2 Regression Algorithms

Quantum regression algorithms are designed to fit models to data points efficiently, leveraging quantum speedups in solving linear systems and performing matrix inversions. Among these, quantum linear regression stands out as a cornerstone technique.

By utilizing the HHL algorithm [39], quantum linear regression computes the regression coefficients in time logarithmic to the matrix size, under suitable conditions like sparsity and well-conditioned matrices. **Quantum data fitting**[116] extends the linear regression framework to accommodate complex relationships between variables. By solving optimization problems in the quantum domain, it minimizes the least-squares error between the predicted and observed data. For example, the coefficients β in data fitting are derived using $\beta = (X^T X)^{-1} X^T y$ where $X^T X$ is efficiently inverted using quantum solvers. This enables rapid convergence and scalability to large datasets [108].

Lastly, **Quantum Principal Component Analysis** (QPCA)[21] complements regression tasks by reducing dimensionality. QPCA is pivotal for preprocessing data in regression models, ensuring computational efficiency and retaining the most informative features.

Table 4: Comparison of QML Algorithms, their effective advantage and applications

Algorithm	**Quantum Advantage**	**Applications**
Quantum k-means[110]	Polynomial speedup in distance calculations	High-dimensional datasets
QAOA based Clustering[111]	Optimization of graph cuts	Community detection in graphs
Quantum DBSCAN [114]	Faster nearest-neighbor searches	Non-linear cluster structures
Hierarchical Clustering [112]	Efficient similarity evaluation	Data visualization
Quantum Linear Regression [117]	Speedup in solving least-squares problems via HHL algorithm	Predictive modeling in physics and finance
Quantum Support Vector Machines (QSVM) [22]	Quadratic speedup in kernel computation with quantum-enhanced feature spaces	Image recognition, spam filtering
Quantum Neural Networks (QNN) [47]	Enhanced expressivity and representation through quantum circuits	Function approximation, generative modeling
Quantum Principal Component Analysis (QPCA) [21]	Efficient extraction of principal components in high-dimensional data	Feature reduction, data visualization

4.6.3 Classification Algorithms

Classification tasks in QML leverage quantum principles to achieve superior performance in distinguishing between classes. The Quantum Support Vector Machine (QSVM)[22] employs quantum kernels to map data into high-dimensional Hilbert spaces, enabling linear separability of complex datasets.

Another noteworthy method is **Quantum Linear Regression** [118], which clusters data points by iteratively minimizing the Euclidean distance between data points and cluster centroids. Particularly the implementation by Schuld et al., [117] quantum implementation achieves quadratic speedup in distance calculations.

4.6.4 Neural Network Algorithms

Quantum neural networks (QNNs)[47] are hybrid models that integrate quantum circuits into traditional neural network architectures. By replacing classical layers with parameterized quantum circuits (PQCs), QNNs exploit quantum parallelism for enhanced data processing.[109].

A key advantage of QNNs is their ability to implement quantum gradients, facilitating efficient training of deep architectures. Quantum variational algorithms further enhance QNNs by optimizing hybrid quantum-classical models [119]. These algorithms minimize the variational loss function using stochastic gradient descent and other classical optimization techniques. Such architectures are well-suited for applications like image recognition and generative modeling.

Finally, **Quantum Deep Learning (QDL)** [120] extends QNNs by incorporating multiple quantum layers and advanced optimization strategies. By leveraging the expressiveness of deep architectures and the computational power of quantum circuits, QDL offers promising avenues for solving complex machine learning tasks.

4.7 Quantum Key Distribution

The development of Quantum Key Distribution (QKD) began with the BB84 protocol proposed by Bennett and Brassard in 1984, which utilized polarized photons and quantum principles like the no-cloning theorem [121] and state collapse upon measurement [23]. E91, introduced by Ekert in 1991 [17], used entangled particles and Bell's theorem to detect eavesdropping. This was followed by BBM92 [122], which adapted BB84 into an entanglement-based framework. The 2000s saw the introduction of Continuous-Variable QKD protocols like GG02 [123], leveraging Gaussian-modulated coherent states compatible with standard telecom systems.

Further advancements in Continuous-Variable QKD protocols include included Differential Phase-Shift Keying (DPSK) [58] and Coherent One-Way (COW) [124], which improved robustness and communication rates. In 2012, Measurement-Device-Independent QKD (MDI-QKD) [125] was introduced to remove detector side-channel vulnerabilities. The Micius satellite mission in 2016 demonstrated global-scale QKD [126], signaling a major leap from theoretical protocols to practical, secure quantum communication systems.

The QKD protocols being invented can be categorized into two major types based on the variable type (whereas a more general taxonomy is shown in Figure 8 and Table 5):

When the number of distinct basis states that can be used for the transmission is restricted by the discrete nature of the Hilbert Space, these fall under the category of **Discrete Variable QKD**:

- **BB84 Protocol** [23] The BB84 protocol, introduced by Bennett and Brassard in 1984, is the first and most widely studied QKD protocol. It employs polarized photons as qubits and uses two complementary bases, typically the rectilinear (horizontal and vertical) and diagonal (45^o and 135^o) bases. Alice randomly selects a basis to encode her qubits and Bob randomly selects a basis to measure them. After transmission, they publicly compare bases and keep only the results where their bases match. Privacy amplification and error correction are then applied to establish a secure key. This protocol's security is based on the no-cloning theorem and the ability to detect eavesdropping through increased error rates.

- **E91 Protocol** [17] Proposed by Artur Ekert in 1991, the E91 protocol utilizes entanglement to establish secure keys. Alice and Bob share entangled photon pairs, and each measures their photons using randomly chosen bases. The correlations between their measurements, as predicted by quantum mechanics, are used to generate the key. The E91 protocol also employs Bell's theorem to detect eavesdropping, ensuring the security of the key. This protocol highlights the fundamental role of entanglement in quantum cryptography.

- **B92 Protocol** [127] The B92 protocol, proposed by Bennett in 1992, is a simplified QKD scheme. Alice sends randomly polarized photons, and Bob uses randomly chosen measurement bases (where the bases belong to one each of two Mutually Unbiased Basis Sets, say 0^o and 45^o polarized photons). After public communication, the shared key is established from the photons measured with matching bases. While simpler than BB84, the B92 protocol is less robust against certain types of eavesdropping [16].

- **BBM92 Protocol** [122] The BBM92 protocol, developed by Bennett, Brassard, and Mermin, adapts the principles of the BB84 protocol to an entanglement-based framework. Alice and Bob share entangled photon pairs and measure them in randomly chosen bases. The results are compared to detect eavesdropping and establish a secure key. This protocol's entanglement-based nature provides additional security against potential passive attacks.

- **Differential Phase-Shift Keying Protocol(DPSK)** [58] The DPSK protocol uses a train of coherent pulses with encoded phase differences between consecutive pulses to transmit information. Bob measures the relative phase using an interferometer. The simplicity of this protocol and its tolerance to photon-number splitting attacks make it suitable for practical implementations, especially in telecom networks.

- **Coherent One-Way (COW) Protocol** [124] The Coherent One-Way protocol uses weak coherent pulses to transmit information. Alice sends two pulses per time slot, where one pulse contains the information bit and the other is a reference. Bob measures the time of arrival and phase of the pulses to detect eavesdropping. The simplicity of the COW protocol makes it a practical choice for high-rate QKD, but it requires careful synchronization and monitoring to maintain security.

- **Discrete-Variable Measurement-Device-Independent QKD (MDI-QKD)** [125]: MDI-QKD, proposed by Lo, Curty, and Qi, addresses the vulnerabilities of measurement devices by making the measurement process independent of trust. Alice and Bob each prepare quantum states and send them to an untrusted third party for Bell-state measurement. The third party announces the measurement results, which Alice and Bob use to generate a secure key. This approach eliminates all detector side-channel attacks, significantly enhancing security. The paper [18] proposes a variant of MDI-QKD which employs a Bell test[128] to enhance security further. By using entanglement and performing Bell-state measurements, the protocol strengthens its resilience to side-channel attacks and ensures the security of the key generation process.

- **Twin-Field QKD (TF-QKD)** [129] The Twin-Field Quantum Key Distribution (QKD) protocol, developed by Lucamarini et al. in 2018, is a pioneering method

in quantum cryptography that extends the reach of secure key distribution without relying on quantum repeaters. In this protocol, two distant parties, Alice and Bob, generate phase-randomized optical fields and transmit them to a central measuring station, Charlie. Charlie performs single-photon interference measurements on the combined fields, using the interference pattern to extract a secure key. This approach enables the protocol to achieve a secure key rate that scales with the square root of the channel transmittance, significantly surpassing the rate-distance limit of traditional QKD protocols. Additionally, the protocol is measurement-device-independent, ensuring security even if the measurement devices are compromised.

Table 5: A tabular taxonomy of the QKD protocols, as done in Figure 8.

Protocol	**Variable**	**Device**	**Entanglement -based**
BB84 [23]	DV	Device Dependent	No
E91 [17]	DV	Device Dependent	Yes
B92 [127]	DV	Device Dependent	No
BBM92 [122]	DV	Device Dependent	Yes
DPSK [58]	DV	Device Dependent	No
COW [124]	DV	Device Dependent	No
Coh. State [123]	CV	Device Dependent	No
No-Switching [130]	CV	Device Dependent	No
Discrete-encoding [131]	CV	Device Dependent	No
Squeezed-state [132]	CV	Device Dependent	No
GG02 [123]	CV	Device Dependent	No
TF-QKD [129]	DV	Semi-Device Independent	No
MDI-DV [125]	DV	Semi-Device Independent	Yes
MDI-CV [133]	CV	Semi-Device Independent	Yes

In the other case, where the number of basis-states "accessible" for the transmission is not restricted due to the size of the Hilbert Space(due to the infinite dimension of the Hilbert Space), these QKD protocols fall under the category of **"Continuous Variable QKD"**:

- **Gaussian Modulated Coherent State Protocol** [123] The GMCS protocol demonstrates that coherent states, which are easier to create experimentally in comparison to other Gaussian states, are sufficient for secure quantum key distribution (QKD). In this continuous-variable protocol, Alice encodes two real variables drawn from a Gaussian distribution onto a coherent state. Bob then randomly measures either the position or momentum quadrature using *homodyne detection*. Security is ensured by the nonorthogonality of coherent states, leveraging the no-cloning theorem. After the quantum communication phase, classical postprocessing involves basis reconciliation and key extraction based on correlated measurement outcomes.

- **No-Switching Protocol** [130] This protocol enhances coherent-state QKD by allowing both quadratures to contribute to the key, rather than discarding one after sifting. Proposed by Weedbrook et al., it replaces Bob's homodyne detection with *heterodyne detection*, enabling simultaneous measurement of both position and momentum quadratures. Although this introduces additional noise due to

the uncertainty principle, it eliminates the need for random basis switching and simplifies the experimental setup. As a result, the protocol achieves higher secret-key rates and is compatible with all known continuous-variable QKD schemes.

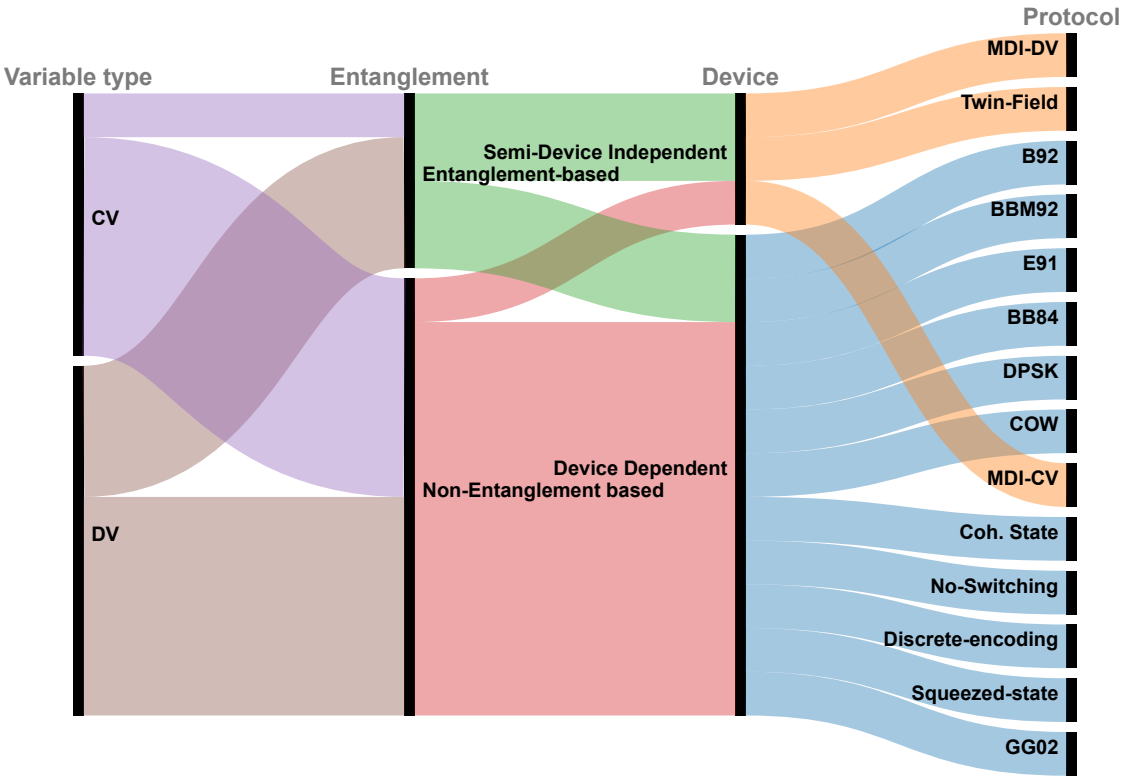

Figure 8: Taxonomy of quantum key distribution (QKD) protocols categorized by encoding type (discrete-variable vs continuous-variable), entanglement usage (entanglement-based vs non-entanglement-based), and device-independence.

- **GG02 Protocol** [123] The GG02 protocol, named after Grosshans and Grangier, is a continuous-variable QKD (CV-QKD) protocol that uses Gaussian-modulated coherent states. Alice encodes information in the quadratures of coherent states and sends them to Bob through a quantum channel. Bob measures the states using homodyne [134] or heterodyne detection. This protocol's compatibility with standard telecom infrastructure and its potential for higher key rates make it a practical choice for many applications.

- **Squeezed-State protocol** [132] The first Gaussian-modulated CV-QKD protocol using Gaussian states and measurements was proposed by Cerf, Levy, and van Assche. Alice encodes a random bit u and a real variable a (drawn from a Gaussian distribution) into a *squeezed vacuum state*, displacing it by a and applying a random phase $\phi = u\pi/2$ to select the quadrature. Averaging over the modulation produces a thermal state with fixed variance, concealing the quadrature choice from Eve. Bob measures a randomly selected quadrature, and both parties retain only the correlated data. A variant by García-Patrón and Cerf introduced heterodyne detection at Bob's end, using reverse reconciliation while discarding one quadrature post-measurement.

- **Discrete-Modulation CV-QKD protocol** [131] Early continuous-variable quantum key distribution (CV-QKD) protocols employed discrete encoding of Gaussian states. With the emergence of Gaussian modulated coherent states, interest in discrete encoding waned. However, recent studies have revived this approach due to its ease of experimental implementation and superior error correction capabilities, enabling longer-distance communication. In discrete-modulation protocols, an N-letter alphabet of coherent states $|k\rangle = |\alpha e^{i2\pi k/N}\rangle$ encodes the key, with Bob applying homodyne or heterodyne detection. These multiletter schemes offer higher key rates over lossy channels. While postprocessing typically involves postselection or reverse reconciliation, the main limitation remains the nascent stage of their security analysis, though notable progress has been achieved.

4.8 Quantum-Inspired Classical Algorithms and Classical Simulation of Quantum Systems

4.8.1 Quantum-Inspired Classical Algorithms

Quantum-inspired algorithms have garnered significant attention across various domains, leveraging principles from quantum mechanics to enhance classical computational techniques. For instance, reinforcement learning has been advanced through quantum-inspired methods, notably in synthesizable drug discovery [135]. Surveys highlight the practical utility of arithmetic-based quantum-inspired algorithms [136], with early interest documented even before quantum computing's rise [137]. Applications span diverse problems, such as solving join order optimization using quantum-inspired digital annealing [138], combinatorial optimization via quantum-inspired evolutionary algorithms [139], and benchmarking heuristic solvers [140]. Further, quantum-inspired techniques have been applied to singular value decomposition [141], virtual screening of ligand molecules [142], recommendation systems [143], and multi-objective optimization for dynamic wireless electric vehicle charging stations [144], demonstrating their versatility and potential.

Quantum-Inspired Machine Learning encompasses a wide range of classical algorithms inspired by quantum mechanical concepts but implemented without the need for quantum hardware [145]. One of the most prominent families are dequantized algorithms, which aim to replicate the performance of quantum algorithms using classical resources under similar assumptions, often leveraging state preparation analogs like L2-norm sampling [140, 146]. Major QiML is directed towards tensor network approaches [147], which model high-dimensional data via quantum-inspired factorizations, mimicking wavefunction representations. Additionally, quantum-inspired metaheuristics such as quantum-inspired evolutionary algorithms (QIEA) [139], quantum-inspired particle swarm optimization (QPSO) [148], and quantum-inspired ant colony optimization (QACO) [149][150] apply quantum analogues like Q-bits and Q-gates to guide optimization. Despite the variety of approaches, what unifies QiML is the use of quantum concepts to inspire improvements in classical learning, offering scalable, hardware-independent alternatives to QML [145].

4.8.2 Classical Simulation of Quantum Systems

Classical simulation plays a crucial role in quantum computing by aiding hardware validation, quantum algorithm testing, and quantum advantage demonstrations [151, 152]. Various simulation methods exist, each suited for different applications and computational constraints. These methods can be broadly classified into exact and approximate

approaches, based on their accuracy. State-vector simulation represents quantum states explicitly as vectors and applies unitary operations using matrix multiplications [153]. While highly accurate, its exponential memory requirement limits feasibility to small systems. Tensor networks, including matrix product states (MPS) and tree tensor networks, efficiently represent low-entanglement quantum states [154, 155]. They are widely used for approximate simulations of quantum circuits with limited entanglement growth. Stabilizer methods efficiently track quantum states using Pauli operator updates for circuits composed mostly of Clifford gates [156, 157]. Extensions, such as stabilizer-rank and quasiprobability methods, allow simulation of near-Clifford circuits [158, 159].

Decision diagram methods, such as tensor decision diagrams, exploit redundancies in quantum circuits to enable efficient simulations [160]. Sampling-based approaches, such as Markov-chain Monte Carlo methods, are useful for estimating quantum system behavior efficiently [154]. These methods are particularly relevant for noisy quantum circuits and near-term quantum applications. Classical simulations of quantum systems continue to evolve, pushing the boundaries of feasibility and efficiency.

5 Current Applications and Industry Relevance

5.1 Healthcare

Quantum computing holds transformative potential for the healthcare sector, primarily through enhanced computational capabilities in molecular simulations, optimization problems, and machine learning [161, 162]. Quantum algorithms such as the VQE [13], QPE [49] and SQD [19] have already demonstrated significant promise in simulating complex molecular systems.

Table 6: Quantum Computing Applications in Healthcare and Drug Discovery

Application	**Algorithms Used**
Molecular Simulation [13, 49, 163, 164, 165]	Ground-state energy computation using VQE, QPE, quantum embedding and Quantum Monte Carlo
Protein Ligand Interactions [166, 167, 168]	VQE, SQD for interaction energy predictions
Drug Discovery Optimization [169, 170]	Hybrid algorithms (QAOA, QUTIE) and QGANs for generative chemistry
Molecular Docking [171, 172, 173]	Quantum Annealing, DCQO, Variational Quantum Adiabatic Algorithm for molecular docking problem.
QML [174, 175, 176, 170]	QSAR modeling, virtual screening, toxicity prediction via hybrid QNNs and QGANs
Protein Folding [177, 178]	Counterdiabatic and resource-efficient quantum algorithms for protein structure prediction
Quantum Bioinformatics [179, 180]	Quantum algorithms for sequence similarity and solvent configuration prediction

Classical approaches, including Density Functional Theory (DFT) [181] and Hartree-Fock [182], often fail to capture strong electron correlations in large molecules. Quantum methods can offer more accurate and scalable solutions. Additional approaches, such as

quantum-assisted simulators [183], quantum-classical hybrid neural networks for binding affinity prediction [184], and network science frameworks [185], continue to expand the horizons of quantum computing in biomedical domains. While the potential is vast, challenges such as hardware noise, limited qubit counts, and short coherence times limit current scalability. Still, hybrid quantum-classical methods are increasingly seen as practical options for the near term. Continued research in error correction, qubit fidelity, and application-specific algorithms is essential to fully leverage quantum computing in healthcare.

Table 6 categorizes prominent applications of quantum computing in healthcare-related domains.

5.2 Finance

Quantum computing has emerged as a promising technology in the financial industry, offering the potential to solve complex problems more efficiently than classical methods [186, 187]. The key areas where quantum algorithms can provide advantages include portfolio optimization, risk assessment, and derivative pricing. Table 7 discusses various quantum algorithms and their potential use-cases in finance.

Table 7: Quantum Computing Applications in Finance

Application	**Algorithms Used**
Portfolio Optimization [188]	Use of QAOA and Quantum Interior Point Methods for asset allocation under constraints
Risk Assessment [189, 190]	Quantum Amplitude Estimation (QAE) for computing VaR and CVaR, improving simulation efficiency
Derivative Pricing [191, 192, 193]	Quantum-enhanced Monte Carlo techniques and QAE for simulating stochastic processes
Fraud Detection [194, 195, 196]	Quantum annealing and clustering for anomly detection in transactional data
Financial Modeling [197, 198, 199, 200]	QGANs, QCBMs, and SSQW-based random state preparation for market data and pricing models, QLSS for numerical solutions of differential equations
NLP in Finance [201]	QNLP for analysis of unstructured financial documents and market news sentiment
Machine Learning [202, 195, 196]	QSVMs, QNNs, and clustering for credit scoring, fraud detection, and market trend analysis

Quantum computing is rapidly transforming the financial industry by offering efficient solutions to complex problems traditionally handled by classical methods [186, 187, 203]. Key applications include portfolio optimization, risk assessment, and derivative pricing. For portfolio optimization, quantum algorithms such as **QAOA** and **Quantum Interior Point Methods (QIPM)** provide advantages over classical techniques like quadratic and mixed-integer programming [188]. In risk assessment, **QAE** enables more efficient computation of risk measures such as **Value-at-Risk (VaR)** and **Conditional VaR (CVaR)**, offering a quadratic speed-up over classical Monte Carlo simulations [189, 190]. Derivative pricing, which involves simulating stochastic processes, also benefits from QAE

and quantum-enhanced Monte Carlo methods [191, 192, 193]. Additionally, **quantum annealing** has been used for fraud detection by identifying anomalous transactions in large datasets [194]. Despite hardware limitations, ongoing research and technological advancements are making quantum algorithms increasingly viable for financial decision-making.

Advances in QML further enhance financial modeling through generative approaches and probabilistic methods. Quantum generative models such as **QGANs** and **Quantum Circuit Born Machines** (QCBMs) have been applied to financial data generation and risk analysis [197], while mutual information-maximizing QGANs extend classical Info-GAN concepts for improved financial modeling [199]. **Quantum Natural Language Processing (QNLP)** enables better understanding of unstructured financial data [201], and **split-step quantum walks (SSQW)** have been used for option pricing through parameterized quantum circuits [198]. Additional developments include quantum portfolio optimization integrated with blockchain [204], quantum Boltzmann machines for risk forecasting using D-Wave annealers [205], and solving option pricing PDEs via their equivalence to the Schrödinger equation [206]. In fraud detection and credit scoring, QSVM, QNN and quantum clustering techniques enhance classification and anomaly detection in financial datasets [202].

5.3 Defence and Communication

Quantum Key Distribution (QKD) is a major advancement in secure communications, particularly in financial transactions, military networks, and critical infrastructure protection. Protocols such as BB84 [16], E91 [17], and Measurement-Device-Independent QKD (MDI-QKD) [125] ensure secure key exchange against potential quantum adversaries. These techniques leverage quantum mechanics principles like entanglement and the no-cloning theorem to prevent eavesdropping. QKD has been successfully implemented in fiber-optic and free-space communication networks, enhancing encryption techniques beyond classical cryptographic methods [207].

Table 8: Quantum Key Distribution Implementations

Protocol	**Distance**	**Country**	**Year**	**Medium**
Decoy-state BB84 [126]	1,200 km	China	2017	Satellite-to-ground
TF-QKD[208]	1,002 km	China	2023	Fiber
TF-QKD[209]	600 km	Japan	2023	Fiber
TF-QKD[210]	511 km	China	2021	Fiber
Decoy-state BB84 [211]	421 km	Switzerland, US	2018	Fiber
DPSK [212]	380 km	India	2021	Fiber

In defense applications, QKD provides an additional layer of security for military communications, intelligence sharing, and satellite-based encrypted links. Satellite-based QKD systems, such as those pioneered by China's Micius satellite, have demonstrated the feasibility of long-range quantum-secure communication [126]. Governments and defense

organizations are actively investing in QKD infrastructure to future-proof secure communications against threats posed by quantum computing advancements in cryptanalysis.

Owing to high costs in integration with existing communication infrastructure the large-scale deployment of QKD still remains a challenge. Research into quantum repeaters [213] and next-generation QKD protocols aims to enhance the scalability and practicality of quantum-secure networks. Table 8 highlights the milestone achievements in QKD implementations.

5.4 Optimization

Quantum computing holds transformative potential for optimization across industries like logistics, finance, and healthcare by leveraging algorithms such as **QAOA** and **Quantum Annealing** to tackle complex combinatorial problems intractable for classical methods [214]. In logistics, these techniques optimize supply chain management, delivery routes, and scheduling, addressing challenges like the traveling salesman problem, vehicle routing, and network flow with Quantum Annealing for routing and cargo loading [215, 216, 217], **DC Quantum Optimization (DCQO)** and Quantum Neural Networks for complex routing [218, 219], QAOA for production scheduling [220], Hybrid Quantum-Classical Annealing for logistic network design [221].

However, hardware limitations and noise susceptibility pose challenges, necessitating hybrid quantum-classical approaches that combine quantum algorithms with traditional methods to enable near-term advancements. Ongoing research in quantum error correction and algorithmic refinements will be critical to fully realizing quantum optimization's potential in industrial applications. Table 9 details some of the applications in ogistics for which quantum algorithms can be potentially used.

Table 9: Quantum Computing Applications in Logistics

Application	**Algorithms Used**
Routing [218, 216, 219, 215]	Quantum Annealing, Digitized Counterdiabatic Quantum Optimization (DCQO), Quantum Neural Networks
Cargo Load [217]	Quantum Annealing
Network Design [221]	Hybrid Quantum-Classical Annealing
Scheduling [220]	QAOA

6 Future Directions

Quantum computing is advancing rapidly, with significant progress in the NISQ era and promising developments toward FTQC. Current devices, with tens to hundreds of qubits, are demonstrating potential in practical applications like pharmaceutical research, optimization, and search techniques, while the transition to FTQC is expected to unlock unprecedented computational power. Despite hardware limitations, ongoing improvements in algorithms and error correction are paving the way for transformative impacts across industries. Below is a summary of key advancements and challenges in this evolving field:

- Quantum computing is poised to revolutionize fields such as pharmaceuticals, logistics, and cryptography, leveraging NISQ-era algorithms like VQE and QAOA,

with FTQC promising exponential speedups in tasks like molecular simulations and factorization [5, 36, 222].

- NISQ algorithms, including VQEs and sample-based methods like SQD for molecular ground state approximation in drug discovery and QAOA for combinatorial optimization, are showing independent utility despite noise and limited qubit connectivity, with potential to outperform classical methods as hardware improves. [168, 166, 167, 14].

- Quantum annealing is effectively addressing complex optimization and simulation problems, demonstrating competitive performance in real-world applications like supply chain optimization, even within the constraints of current hardware limitations [223, 224, 225, 226, 227].

- Improvements in early-FTQC, such as experiments with logical qubits and lower-depth, hardware-friendly algorithms with rapidly progressing research in error correction and mitigation, are blurring the line between NISQ and early-FTQC, enabling more robust applications in optimization and simulation [228, 229].

- FTQC algorithms like Shor's for factorization and quantum linear system solvers offer super-polynomial speedups but require substantial number of qubits with high coherence times; further investigation is needed to demonstrate genuine quantum advantage [230], and simulation algorithms face challenges in eliminating the need for good initial states [36, 39, 101, 222].

The journey from NISQ to FTQC is a nascent field with significant challenges, including hardware scalability and error correction, but continuous advancements in quantum neural networks, post-quantum cryptography, simulation, optimization algorithms signal a bright future where quantum computing will solve complex problems, transforming industries [231, 100, 19, 226].

7 Summary

This work presents a comprehensive overview of quantum computing algorithms, beginning with its foundational principles and the quantum systems underpinning algorithms development. We discussed key quantum algorithms and emphasized their theoretical significance and current industrial applications. Through a systematic classification of these algorithms, the paper highlights the growing maturity of quantum computing as a scientific discipline.

The inclusion of QML underscores the interdisciplinary nature of quantum computing and its potential to revolutionize data analysis and optimization tasks. Similarly, the discussion of classical algorithms for simulating quantum systems bridges the gap between quantum mechanics and practical computational tools. These methods remain essential for validating quantum systems and exploring the performance of quantum algorithms, especially as experimental quantum hardware continues to evolve.

Recent advances, such as improvements in error handling techniques, and the scaling of quantum processors, mark significant milestones in the field. The integration of classical and quantum computing frameworks is fostering hybrid approaches that are redefining computational boundaries. As quantum technologies continue to evolve, overcoming scalability barriers and advancing algorithmic design will be pivotal in realizing their full transformative potential.

Acknowledgements

The authors express their sincere gratitude to Prof. Alok Shukla of Ahmedabad University for his timely assistance, invaluable guidance, and continued encouragement, which contributed significantly to shaping this review. We also extend our appreciation to the advisors of Qclairvoyance for their support, constructive discussions, and inspiration throughout the preparation of this work.

Funding

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Conflict of Interest

The authors declare that they have no competing interests.

References

[1] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information, Section 1.2.2, Section 2, Section 5.1.* Cambridge University Press, 2000.

[2] A. Einstein, B. Podolsky, and N. Rosen. Can Quantum-Mechanical Description of Physical Reality Be Considered Complete? *Physical Review*, 47(10):777–780, May 1935.

[3] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, October 1997.

[4] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996.

[5] John Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018.

[6] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. Quantum computation by adiabatic evolution. *arXiv preprint quant-ph/0001106*, 2000.

[7] Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman. Exponential algorithmic speedup by a quantum walk. In *Proceedings of the thirty-fifth annual ACM symposium on Theory of computing*, STOC03, page 59–68. ACM, June 2003.

[8] R.P. Feynman. Simulating physics with computers. *International Journal of Theoretical Physics*, 21(6/7):467–488, 1982.

[9] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 400:97–117, 1985.

[10] David Deutsch and Richard Jozsa. Rapid solution of problems by quantum computation. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 439(1907):553–558, 1992.

[11] Daniel R. Simon. On the power of quantum computation. *SIAM Journal on Computing*, 26(5):1474–1483, 1997.

[12] P.W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[13] Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1):4213, 2014.

[14] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

[15] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. The quest for a quantum neural network. *Quantum Information Processing*, 13(11):2567–2586, August 2014.

[16] Charles H. Bennett. Quantum cryptography using any two nonorthogonal states. *Phys. Rev. Lett.*, 68:3121–3124, May 1992.

[17] Artur K. Ekert. Quantum cryptography based on Bell's theorem. *Phys. Rev. Lett.*, 67:661–663, Aug 1991.

[18] Feihu Xu, Marcos Curty, Bing Qi, and Hoi-Kwong Lo. Practical aspects of measurement-device-independent quantum key distribution. *New Journal of Physics*, 15(11):113007, November 2013.

[19] Javier Robledo-Moreno, Mario Motta, Holger Haas, Ali Javadi-Abhari, Petar Jurcevic, William Kirby, Simon Martiel, Kunal Sharma, Sandeep Sharma, Tomonori Shirakawa, Iskandar Sitdikov, Rong-Yang Sun, Kevin J. Sung, Maika Takita, Minh C. Tran, Seiji Yunoki, and Antonio Mezzacapo. Chemistry beyond exact solutions on a quantum-centric supercomputer. *arXiv preprint arXiv:2405.05068*, 2024.

[20] Nicholas H. Stair and Francesco A. Evangelista. Simulating many-body systems with a projective quantum eigensolver. *PRX Quantum*, 2:030301, Jul 2021.

[21] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9):631–633, July 2014.

[22] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113:130503, Sep 2014.

[23] Charles H. Bennett and Gilles Brassard. Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560:7–11, 2014.

[24] John H. Reina, Luis Quiroga, and Neil F. Johnson. Decoherence of quantum registers. *Phys. Rev. A*, 65:032326, Mar 2002.

[25] Adriano Barenco, Todd A. Brun, Rüdiger Schack, and Timothy P. Spiller. Effects of noise on quantum error correction algorithms. *Physical Review A*, 56(2):1177–1188, August 1997.

[26] Alexander M. Dalzell, Sam McArdle, Mario Berta, Przemyslaw Bienias, Chi-Fang Chen, András Gilyén, Connor T. Hann, Michael J. Kastoryano, Emil T. Khabiboulline, Aleksander Kubica, Grant Salton, Samson Wang, and Fernando G. S. L. Brandão. Quantum algorithms: A survey of applications and end-to-end complexities, 2023.

[27] Alessandro Berti and Francesco Ghisoni. Efficient quantum state preparation with bucket brigade qram. *arXiv preprint arXiv:2510.16149*, 2025.

[28] Daniel F. V. James, Paul G. Kwiat, William J. Munro, and Andrew G. White. Measurement of qubits. *Phys. Rev. A*, 64:052312, Oct 2001.

[29] J. J. Sakurai. *Modern Quantum Mechanics (Revised Edition) Chapter 5.4*. Addison Wesley, 1 edition, September 1993.

[30] Jun Zhang, Jiri Vala, Shankar Sastry, and K. Birgitta Whaley. Geometric theory of nonlocal two-qubit operations. *Physical Review A*, 67(4), April 2003.

[31] Adriano Barenco, Charles H. Bennett, Richard Cleve, David P. DiVincenzo, Norman Margolus, Peter Shor, Tycho Sleator, John A. Smolin, and Harald Weinfurter. Elementary gates for quantum computation. *Phys. Rev. A*, 52:3457–3467, Nov 1995.

[32] Joaquín Ossorio-Castillo, Ulises Pastor-Díaz, and José M. Tornero. A generalisation of the phase kick-back, 2022.

[33] Andrew Lucas. Ising formulations of many NP problems. *Frontiers in Physics*, 2:5, 2014.

[34] D. Coppersmith. An approximate fourier transform useful in quantum factoring. *arXiv preprint quant-ph/0201067*, 2002.

[35] A. Yu. Kitaev. Quantum measurements and the abelian stabilizer problem. *arXiv preprint quant-ph/9511026*, November 1995.

[36] Peter W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings of the 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994.

[37] Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Information*, 305:53–74, 2002.

[38] Dorit Aharonov, Vaughan Jones, and Zeph Landau. A polynomial quantum algorithm for approximating the jones polynomial. *arXiv preprint quant-ph/0511096*, 2006.

[39] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

[40] Guang Hao Low and Isaac L. Chuang. Optimal hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.*, 118:010501, Jan 2017.

[41] Guang Hao Low and Isaac L. Chuang. Hamiltonian Simulation by Qubitization. *Quantum*, 3:163, July 2019.

[42] Scott Aaronson and Alex Arkhipov. The computational complexity of linear optics. In *Proceedings of the Forty-Third Annual ACM Symposium on Theory of Computing*, STOC '11, page 333–342, New York, NY, USA, 2011.

[43] András Gilyén, Yuan Su, Guang Hao Low, and Nathan 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*, STOC '19. ACM, June 2019.

[44] Joseph G. Smith, Crispin H. W. Barnes, and David R. M. Arvidsson-Shukur. Adaptive bayesian quantum algorithm for phase estimation. *Phys. Rev. A*, 109:042412, Apr 2024.

[45] Andrew M. Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Info. Comput.*, 12(11–12):901–924, November 2012.

[46] Pablo Arnault, Pablo Arrighi, Steven Herbert, Evi Kasnetsi, and Tianyi Li. A typology of quantum algorithms. *arXiv preprint arXiv:2407.05178*, 2024.

[47] Sanjay Gupta and R.K.P. Zia. Quantum neural networks. *Journal of Computer and System Sciences*, 63(3):355–383, 2001.

[48] Alán Aspuru-Guzik, Anthony D. Dutoi, Peter J. Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, September 2005.

[49] Alán Aspuru-Guzik, Anthony D. Dutoi, Peter J. Love, and Martin Head-Gordon. Simulated quantum computation of molecular energies. *Science*, 309(5741):1704–1707, September 2005.

[50] Tyson Jones, Suguru Endo, Sam McArdle, Xiao Yuan, and Simon C. Benjamin. Variational quantum algorithms for discovering hamiltonian spectra. *Phys. Rev. A*, 99:062304, Jun 2019.

[51] Byungjoo Kim, Kang-Min Hu, Myung-Hyun Sohn, Yosep Kim, Yong-Su Kim, Seung-Woo Lee, and Hyang-Tag Lim. Qudit-based variational quantum eigensolver using photonic orbital angular momentum states. *Science Advances*, 10(43):3472, 2024.

[52] William J. Huggins, Sam McArdle, Thomas E. O'Brien, Joonho Lee, Nicholas C. Rubin, Sergio Boixo, K. Birgitta Whaley, Ryan Babbush, and Jarrod R. McClean. Virtual distillation for quantum error mitigation. *Phys. Rev. X*, 11:041036, Nov 2021.

[53] Iordanis Kerenidis and Anupam Prakash. Quantum recommendation systems. *arXiv preprint arXiv:1603.08675*, 2016.

[54] Dominic W. Berry, Andrew M. Childs, Aaron Ostrander, and Guoming Wang. Quantum algorithm for linear differential equations with exponentially improved dependence on precision. *Communications in Mathematical Physics*, 356(3):1057–1081, October 2017.

[55] Dong An and Lin Lin. Quantum linear system solver based on time-optimal adiabatic quantum computing and quantum approximate optimization algorithm. *ACM Transactions on Quantum Computing*, 3(2):1–28, March 2022.

[56] Anna Bernasconi, Alessandro Berti, Gianna Maria del Corso, and Alessandro Poggiali. Quantum subroutine for efficient matrix multiplication. *IEEE Access*, 12:116274–116284, 2024.

[57] Anna Bernasconi, Alessandro Berti, Gianna M. Del Corso, Riccardo Guidotti, and Alessandro Poggiali. Quantum subroutine for variance estimation: algorithmic design and applications. *Quantum Machine Intelligence*, 6(2):78, 2024.

[58] Kyo Inoue, Edo Waks, and Yoshihisa Yamamoto. Differential phase shift quantum key distribution. *Phys. Rev. Lett.*, 89:037902, Jun 2002.

[59] M. Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 99–108, New York, NY, USA, 1996.

[60] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the thirty-seventh annual ACM symposium on Theory of computing (STOC '05)*, pages 84–93. ACM, Jan 2005.

[61] Dorit Aharonov, Andris Ambainis, Julia Kempe, and Umesh Vazirani. Quantum walks on graphs. In *Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing*, STOC '01, page 50–59, New York, NY, USA, 2001.

[62] Y. Aharonov, L. Davidovich, and N. Zagury. Quantum random walks. *Phys. Rev. A*, 48:1687–1690, 1993.

[63] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997.

[64] Alok Shukla and Prakash Vedula. An efficient implementation of a quantum search algorithm for arbitrary n. *The European Physical Journal Plus*, 140(6):575, 2025.

[65] Andris Ambainis. Quantum walks and their algorithmic applications. *International Journal of Quantum Information*, 1(4):507–518, 2003.

[66] Frederic Magniez, Ashwin Nayak, Peter C. Richter, and Miklos Santha. Search via quantum walk. *SIAM Journal on Computing*, 37(1):413–424, 2007.

[67] Kostas Blekos, Dean Brand, Andrea Ceschini, Chiao-Hui Chou, Rui-Hao Li, Komal Pandya, and Alessandro Summer. A review on quantum approximate optimization algorithm and its variants. *Physics Reports*, 1068:1–66, June 2024.

[68] Rebekah Herrman, Phillip C. Lotshaw, James Ostrowski, Travis S. Humble, and George Siopsis. Multi-angle quantum approximate optimization algorithm. *Scientific Reports*, 12(1):6781, April 2022.

[69] P. Chandarana, N. N. Hegade, Koushik Paul, F. Albarrán-Arriagada, Enrique Solano, A. del Campo, and Xi Chen. Digitized-counterdiabatic quantum approximate optimization algorithm. *Phys. Rev. Research*, 4(1):013141, July 2021.

[70] Linghua Zhu, Ho Lun Tang, George S. Barron, F. A. Calderon-Vargas, Nicholas J. Mayhall, Edwin Barnes, and Sophia E. Economou. Adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer. *Phys. Rev. Res.*, 4:033029, Jul 2022.

[71] Daniel J. Egger, Jakub Marecek, and Stefan Woerner. Warm-starting quantum optimization. *Quantum*, 5:479, June 2021.

[72] Andreas Bärtschi and Stephan Eidenbenz. Grover mixers for qaoa: Shifting complexity from mixer design to state preparation. In *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 72–82, October 2020.

[73] Sergey Bravyi, Alexander Kliesch, Robert Koenig, and Eugene Tang. Obstacles to variational quantum optimization from symmetry protection. *Physical Review Letters*, 125(26):260505, 2020.

[74] M. Born and V. Fock. Beweis des adiabatensatzes. *Zeitschrift für Physik*, 51(3):165–180, March 1928.

[75] Philipp Hauke, Helmut G. Katzgraber, Wolfgang Lechner, Hidetoshi Nishimori, and William D. Oliver. Perspectives of quantum annealing: Methods and implementations. *Reports on Progress in Physics*, 83:054401, March 2020.

[76] C. S. Hamilton, R. Kruse, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex. Gaussian boson sampling. *Phys. Rev. Lett.*, 119:170501, 2017.

[77] J. Huh, G. G. Guerreschi, B. Peropadre, J. R. McClean, and A. Aspuru-Guzik. Boson sampling for molecular vibronic spectra. *Nature Photonics*, 9:615, 2015.

[78] J. M. Arrazola and T. R. Bromley. Using gaussian boson sampling to find dense subgraphs. *Phys. Rev. Lett.*, 121:030503, 2018.

[79] S. Scheel, K. Nemoto, W. J. Munro, and P. L. Knight. Measurement-induced nonlinearity in linear optics. *Phys. Rev. A*, 68:032310, 2003.

[80] R. Kruse, C. S. Hamilton, L. Sansoni, S. Barkhofen, C. Silberhorn, and I. Jex. Detailed study of gaussian boson sampling. *Phys. Rev. A*, 100:032326, 2019.

[81] G. Bressanini, H. Kwon, and M. S. Kim. Noise thresholds for classical simulability of nonlinear boson sampling. *Phys. Rev. A*, 106:042413, 2022.

[82] N. Quesada, J. M. Arrazola, and N. Killoran. Gaussian boson sampling using threshold detectors. *Phys. Rev. A*, 98:062322, 2018.

[83] L. S. Madsen, F. Laudenbach, M. F. Askarani, F. Rortais, T. Vincent, J. F. Bulmer, F. M. Miatto, L. Neuhaus, L. G. Helt, M. J. Collins, et al. Quantum computational advantage with a programmable photonic processor. *Nature*, 606:75, 2022.

[84] H.-S. Zhong, Y.-H. Deng, J. Qin, H. Wang, M.-C. Chen, L.-C. Peng, Y.-H. Luo, D. Wu, S.-Q. Gong, H. Su, Y. Hu, P. Hu, X.-Y. Yang, W.-J. Zhang, H. Li, Y. Li, X. Jiang, L. Gan, G. Yang, L. You, Z. Wang, L. Li, N.-L. Liu, J. J. Renema, C.-Y. Lu, and J.-W. Pan. Phase-programmable gaussian boson sampling using stimulated squeezed light. *Phys. Rev. Lett.*, 127:180502, 2021.

[85] P. Jordan and E. Wigner. Über das paulische Äquivalenzverbot. *Zeitschrift für Physik*, 47(9):631–651, September 1928.

[86] Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549(7671):242–246, 2017.

[87] Panagiotis Kl. Barkoutsos, Jerome F. Gonthier, Igor Sokolov, Nikolaj Moll, Gian Salis, Andreas Fuhrer, Marc Ganzhorn, Daniel J. Egger, Matthias Troyer, Antonio Mezzacapo, Stefan Filipp, and Ivano Tavernelli. Quantum algorithms for electronic structure calculations: Particle-hole hamiltonian and optimized wave-function expansions. *Physical Review A*, 98(2):022322, 2018.

[88] Rodney J. Bartlett and Monika Musiał. Coupled-cluster theory in quantum chemistry. *Reviews of Modern Physics*, 79(1):291–352, 2007.

[89] Mario Motta, Kevin J. Sung, K. Birgitta Whaley, Martin Head-Gordon, and James Shee. Bridging physical intuition and hardware efficiency for correlated electronic states: the local unitary cluster jastrow ansatz for electronic structure. *Chem. Sci.*, 14:11213–11227, 2023.

[90] Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson. The variational quantum eigensolver: A review of methods and best practices. *Physics Reports*, 986:1–128, November 2022.

[91] Mario Motta, Chong Sun, Adrian T. K. Tan, Matthew J. O'Rourke, Erika Ye, Austin J. Minnich, Fernando G. S. L. Brandão, and Garnet Kin-Lic Chan. Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution. *Nature Physics*, 16(2):205–210, November 2019.

[92] Hao-En Li, Xiang Li, Jia-Cheng Huang, Guang-Ze Zhang, Zhu-Ping Shen, Chen Zhao, Jun Li, and Han-Shi Hu. Variational quantum imaginary time evolution for matrix product state ansatz with tests on transcorrelated hamiltonians. *The Journal of Chemical Physics*, 161(14):144104, Oct 2024.

[93] Masuo Suzuki. General theory of fractal path integrals with applications to many-body theories and statistical physics. *Journal of Mathematical Physics*, 32(2):400–407, 02 1991.

[94] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.

[95] Andrew M. Childs, Yuan Su, Minh C. Tran, Nathan Wiebe, and Shuchen Zhu. Theory of trotter error with commutator scaling. *Physical Review X*, 11(1), February 2021.

[96] Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by uniform spectral amplification. *arXiv preprint arXiv:1707.05391*, 2017.

[97] Dominic W. Berry, Andrew M. Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, page 792–809. IEEE, October 2015.

[98] Arjen K. Lenstra and Hendrik W. Lenstra. The development of the number field sieve. *Lecture Notes in Mathematics*, 1554:11–42, 1993.

[99] Ronald L. Rivest, Adi Shamir, and Leonard Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

[100] Daniel J. Bernstein and Tanja Lange. Post-quantum cryptography. *Nature*, 549(7671):188–194, September 2017.

[101] Ryan Babbush, Jarrod R. McClean, Michael Newman, Craig Gidney, Sergio Boixo, and Hartmut Neven. Focus beyond quadratic speedups for error-corrected quantum advantage. *PRX Quantum*, 2(1):010103, March 2021.

[102] Pedro C.S. Costa, Dong An, Yuval R. Sanders, Yuan Su, Ryan Babbush, and Dominic W. Berry. Optimal scaling quantum linear-systems solver via discrete adiabatic theorem. *PRX Quantum*, 3:040303, Oct 2022.

[103] Sumeet Khatri, Ryan LaRose, Alexander Poremba, Lukasz Cincio, Andrew T. Sornborger, and Patrick J. Coles. Quantum-assisted quantum compiling. *Quantum*, 3:140, May 2019.

[104] Dmitry Grinko, Julien Gacon, Christa Zoufal, and Stefan Woerner. Iterative quantum amplitude estimation. *npj Quantum Information*, 7(1):52, 2021.

[105] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum random access memory. *Phys. Rev. Lett.*, 100:160501, Apr 2008.

[106] Alok Shukla and Prakash Vedula. A hybrid classical-quantum algorithm for digital image processing. *Quantum Information Processing*, 22(1):3, 2022.

[107] Carlos Bravo-Prieto, Diego García-Martín, and José I. Latorre. Quantum singular value decomposer. *Phys. Rev. A*, 101:062310, Jun 2020.

[108] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549(7671):195–202, 2017.

[109] Junyu Liu, Fabio Tacchino, Jeffrey R. Glick, Liang Jiang, and Alberto Mezzacapo. Representation learning via quantum neural tangent kernels. *PRX Quantum*, 3(3):030323, September 2022.

[110] Alessandro Poggiali, Alessandro Berti, Anna Bernasconi, Gianna M. Del Corso, and Riccardo Guidotti. Quantum clustering with k-means: A hybrid approach. *Theoretical Computer Science*, 992:114466, April 2024.

[111] Sayantan Pramanik and M. Girish Chandra. Quantum-assisted graph clustering and quadratic unconstrained d-ary optimisation. *arXiv preprint arXiv:2004.02608*, 2021.

[112] I. D. Lazarev, Marek Narozniak, Tim Byrnes, and A. N. Pyrkov. Hybrid quantum-classical unsupervised data clustering based on the self-organizing feature map. *Phys. Rev. A*, 111:012416, Jan 2025.

[113] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, and et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 226–231, 1996.

[114] Xuming Xie, Longzhen Duan, Taorong Qiu, and Junru Li. Quantum algorithm for mmng-based dbscan. *Scientific Reports*, 11(1):15559, 2021.

[115] H.-Y. Huang, Richard Kueng, and John Preskill. Information-theoretic bounds on quantum advantage in machine learning. *Physical Review Letters*, 126(19):190505, May 2021.

[116] Nathan Wiebe, Daniel Braun, and Seth Lloyd. Quantum algorithm for data fitting. *Phys. Rev. Lett.*, 109:050505, Aug 2012.

[117] Maria Schuld, Ilya Sinayskiy, and Francesco Petruccione. Prediction by linear regression on a quantum computer. *Physical Review A*, 94:022342, 2016.

[118] Guoming Wang. Quantum algorithm for linear regression. *Phys. Rev. A*, 96:012335, Jul 2017.

[119] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, August 2021.

[120] Nathan Wiebe, Ashish Kapoor, and Krysta M. Svore. Quantum deep learning. *arXiv preprint arXiv:1412.3489*, 2015.

[121] W. K. Wootters and W. H. Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.

[122] Charles H. Bennett, Gilles Brassard, and N. David Mermin. Quantum cryptography without Bell's theorem. *Phys. Rev. Lett.*, 68:557–559, Feb 1992.

[123] Frédéric Grosshans and Philippe Grangier. Continuous variable quantum cryptography using coherent states. *Phys. Rev. Lett.*, 88:057902, Jan 2002.

[124] D. Stucki, N. Brunner, N. Gisin, V. Scarani, and H. Zbinden. Fast and simple one-way quantum key distribution. *Applied Physics Letters*, 87:011102, 2005.

[125] Hoi-Kwong Lo, Marcos Curty, and Bing Qi. Measurement-device-independent quantum key distribution. *Phys. Rev. Lett.*, 108:130503, Mar 2012.

[126] S.-K. Liao, W.-Q. Cai, W.-Y. Liu, L. Zhang, and et al. Li, Y. Satellite-based entanglement distribution over 1200 kilometers. *Science*, 356:1140–1144, 2017.

[127] Cătălin Anghel, Adrian Istrate, and Mihai Vlase. A comparison of several implementations of b92 quantum key distribution protocol. In *2022 26th International Conference on System Theory, Control and Computing (ICSTCC)*, pages 374–379, 2022.

[128] Gopalan Raghavan. Device independence and the quest towards physical limits of privacy. In Sergio Curilef and Angel Ricardo Plastino, editors, *Topics on Quantum Information Science*, chapter 7. IntechOpen, London, 2021.

[129] M. Lucamarini, Z. L. Yuan, J. F. Dynes, and A. J. Shields. Overcoming the rate–distance limit of quantum key distribution without quantum repeaters. *Nature*, 557(7705):400–403, May 2018.

[130] Christian Weedbrook, Andrew M. Lance, Warwick P. Bowen, Thomas Symul, Timothy C. Ralph, and Ping Koy Lam. Quantum cryptography without switching. *Phys. Rev. Lett.*, 93:170504, Oct 2004.

[131] A Leverrier, E Karpov, P Grangier, and N J Cerf. Security of continuous-variable quantum key distribution: towards a de finetti theorem for rotation symmetry in phase space. *New Journal of Physics*, 11(11):115009, November 2009.

[132] Mark Hillery. Quantum cryptography with squeezed states. *Phys. Rev. A*, 61:022309, Jan 2000.

[133] Samuel L. Braunstein and Stefano Pirandola. Side-channel-free quantum key distribution. *Phys. Rev. Lett.*, 108:130502, Mar 2012.

[134] Ulf Leonhardt. *Measuring the Quantum State of Light, Chapter 4.2.* Cambridge University Press, Cambridge, UK, 1997.

[135] Dannong Wang, Jintai Chen, Zhiding Liang, Tianfan Fu, and Xiao-Yang Liu. Quantum-inspired reinforcement learning for synthesizable drug design. *arXiv preprint arXiv:2409.09183*, 2024.

[136] Juan Miguel Arrazola, Alain Delgado, Bhaskar Roy Bardhan, and Seth Lloyd. Quantum-inspired algorithms in practice. *Quantum*, 4:307, August 2020.

[137] Gexiang Zhang. Quantum-inspired evolutionary algorithms: a survey and empirical study. *Journal of Heuristics*, 17(3):303–351, June 2011.

[138] Manuel Schönberger, Immanuel Trummer, and Wolfgang Mauerer. Quantum-inspired digital annealing for join ordering. *Proc. VLDB Endow.*, 17(3):511–524, November 2023.

[139] Kuk-Hyun Han and Jong-Hwan Kim. Quantum-inspired evolutionary algorithm for a class of combinatorial optimization. *IEEE Transactions on Evolutionary Computation*, 6(6):580–593, 2002.

[140] Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st annual ACM SIGACT symposium on theory of computing*, pages 217–228, 2019.

[141] Chen Ding, Tian-Yi Bao, and He-Liang Huang. Quantum-inspired support vector machine. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12):7210–7222, 2022.

[142] Maritza Hernandez, Guo Liang Gan, Kirby Linvill, Carl Dukatz, Jun Feng, and Govinda Bhisetti. A quantum-inspired method for three-dimensional ligand-based virtual screening. *Journal of Chemical Information and Modeling*, 59(10):4475–4485, October 2019.

[143] Ewin Tang. A quantum-inspired classical algorithm for recommendation systems. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, STOC 2019, page 217–228, New York, NY, USA, 2019.

[144] Dong Hua, Chenzhang Chang, Suisheng Liu, Yiqing Liu, Dunhao Ma, and Hua Hua. Quantum-inspired multi-objective optimization framework for dynamic wireless electric vehicle charging in highway networks under stochastic traffic and renewable energy variability. *World Electric Vehicle Journal*, 16(4):221, 2025.

[145] Maria Schuld and Francesco Petruccione. *Machine learning with quantum computers, Chapter 3*. Springer, 2021.

[146] Nai-Hui Chia, András Gilyén, Tongyang Li, Han-Hsuan Lin, Ewin Tang, and Chunhao Wang. 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*, STOC 2020, page 387–400, New York, NY, USA, 2020.

[147] Juan Carrasquilla. Machine learning for quantum matter. *Advances in Physics: X*, 5(1):1797528, 2020.

[148] Jun Sun, Bin Feng, and Wenbo Xu. Particle swarm optimization with particles having quantum behavior. In *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*, volume 1, pages 325–331. IEEE, 2004.

[149] Ling Wang, Qun Niu, and Minrui Fei. A novel quantum ant colony optimization algorithm. In *Bio-Inspired Computational Intelligence and Applications: International Conference on Life System Modeling and Simulation, LSMS 2007, Shanghai, China, September 14-17, 2007. Proceedings*, pages 277–286. Springer, 2007.

[150] Madhushree Das, Arindam Roy, Samir Maity, and Samarjit Kar. A quantum-inspired ant colony optimization for solving a sustainable four-dimensional traveling salesman problem under type-2 fuzzy variable. *Advanced Engineering Informatics*, 55:101816, 2023.

[151] Thi Ha Kyaw, Tim Menke, Sukin Sim, Abhinav Anand, Nicolas P. D. Sawaya, William D. Oliver, Gian Giacomo Guerreschi, and Alán Aspuru-Guzik. Quantum computer-aided design: Digital quantum simulation of quantum processors. *Physical Review Applied*, 16(4):044042, 2021.

[152] J. Eli Bourassa, Rafael N. Alexander, Michael Vasmer, Ashlesha Patil, Ilan Tzitrin, Takaya Matsuura, Daiqin Su, Ben Q. Baragiola, Saikat Guha, Guillaume Dauphinais, et al. Blueprint for a scalable photonic fault-tolerant quantum computer. *Quantum*, 5:392, 2021.

[153] G. F. Viamontes, I. L. Markov, and J. P. Hayes. *Quantum Circuit Simulation, Chapter III*. Springer Science & Business Media, 2009.

[154] Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J Bremner, John M Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. *Nature Physics*, 14(6):595–600, 2018.

[155] Cupjin Huang, Fang Zhang, Michael Newman, Junjie Cai, Xun Gao, Zhengxiong Tian, Junyin Wu, Haihong Xu, Huanjun Yu, Bo Yuan, et al. Classical simulation of quantum supremacy circuits. *arXiv preprint arXiv:2005.06787*, 2020.

[156] Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70:052328, November 2004.

[157] Craig Gidney. Stim: A fast stabilizer circuit simulator. *Quantum*, 5:497, July 2021.

[158] Sergey Bravyi, Dan Browne, Padraic Calpin, Earl Campbell, David Gosset, and Mark Howard. Simulation of quantum circuits by low-rank stabilizer decompositions. *Quantum*, 3:181, September 2019.

[159] James R. Seddon, Bartosz Regula, Hakop Pashayan, Yingkai Ouyang, and Earl T. Campbell. Quantifying quantum speedups: Improved classical simulation from tighter magic monotones. *PRX Quantum*, 2:010345, March 2021.

[160] Xin Hong, Xiangzhen Zhou, Sanjiang Li, Yuan Feng, and Mingsheng Ying. A tensor network based decision diagram for representation of quantum circuits. *arXiv preprint arXiv:2009.02618*, 2021.

[161] Alberto Baiardi, Matthias Christandl, and Markus Reiher. Quantum computing for molecular biology. *ChemBioChem*, 24(13):e202300120, 2023.

[162] Frederik F. Flöther. The state of quantum computing applications in health and medicine. *Research Directions: Quantum Technologies*, page 1–21, July 2023.

[163] Max Rossmannek, Fabijan Pavošević, Angel Rubio, and Ivano Tavernelli. Quantum embedding method for the simulation of strongly correlated systems on quantum computers. *The Journal of Physical Chemistry Letters*, 14(14):3491–3497, April 2023.

[164] Changsu Cao, Jinzhao Sun, Xiao Yuan, Han-Shi Hu, Hung Q. Pham, and Dingshun Lv. Ab initio quantum simulation of strongly correlated materials with quantum embedding. *npj Computational Materials*, 9(1):78, 2023.

[165] Luning Zhao, Joshua J. Goings, Willie Aboumrad, Andrew Arrasmith, Lazaro Calderin, Spencer Churchill, Dor Gabay, Thea Harvey-Brown, Melanie Hiles, Magda Kaja, Matthew Keesan, Karolina Kulesz, Andrii Maksymov, Mei Maruo,

Mauricio Muñoz, Bas Nijholt, Rebekah Schiller, Yvette de Sereville, Amy Smidutz, Felix Tripier, Grace Yao, Trishal Zaveri, Coleman Collins, Martin Roetteler, Evgeny Epifanovsky, Arseny Kovyrshin, Lars Tornberg, Anders Broo, Jeff R. Hammond, Zohim Chandani, Pradnya Khalate, Elica Kyoseva, Yi-Ting Chen, Eric M. Kessler, Cedric Yen-Yu Lin, Gandhi Ramu, Ryan Shaffer, Michael Brett, Benchen Huang, Maxime R. Hugues, and Tyler Y. Takeshita. Quantum-classical auxiliary field quantum monte carlo with matchgate shadows on trapped ion quantum computers. *arXiv preprint arXiv:2506.22408*, 2025.

[166] Josh J. M. Kirsopp, Cono Di Paola, David Zsolt Manrique, Michal Krompiec, Gabriel Greene-Diniz, Wolfgang Guba, Agnes Meyder, Detlef Wolf, Martin Strahm, and David Muñoz Ramo. Quantum computational quantification of protein–ligand interactions. *International Journal of Quantum Chemistry*, 122(22):e26975, 2022.

[167] Hocheol Lim, Doo Hyung Kang, Jeonghoon Kim, Aidan Pellow-Jarman, Shane McFarthing, Rowan Pellow-Jarman, Hyeon-Nae Jeon, Byungdu Oh, June-Koo Kevin Rhee, and Kyoung Tai No. Fragment molecular orbital-based variational quantum eigensolver for quantum chemistry in the age of quantum computing. *Scientific Reports*, 14(1):2422, 2024.

[168] Akhil Shajan, Danil Kaliakin, Abhishek Mitra, Javier Robledo Moreno, Zhen Li, Mario Motta, Caleb Johnson, Abdullah Ash Saki, Susanta Das, Iskandar Sitdikov, Antonio Mezzacapo, and Kenneth M. Merz Jr. Towards quantum-centric simulations of extended molecules: sample-based quantum diagonalization enhanced with density matrix embedding theory. *arXiv preprint arXiv:2411.09861*, 2024.

[169] Hoang M Ngo, My T Thai, and Tamer Kahveci. Qutie: quantum optimization for target identification by enzymes. *Bioinformatics Advances*, 3(1):112, August 2023.

[170] Po-Yu Kao, Ya-Chu Yang, Wei-Yin Chiang, Jen-Yueh Hsiao, Yudong Cao, Alex Aliper, Feng Ren, Alán Aspuru-Guzik, Alex Zhavoronkov, Min-Hsiu Hsieh, and Yen-Chu Lin. Exploring the advantages of quantum generative adversarial networks in generative chemistry. *Journal of Chemical Information and Modeling*, 63(11):3307–3318, May 2023.

[171] Mathieu Garrigues, Victor Onofre, and Noé Bosc-Haddad. Towards molecular docking with neutral atoms. *arXiv preprint arXiv:2402.06770*, 2024.

[172] Emanuele Triuzzi, Riccardo Mengoni, Francesco Micucci, Domenico Bonanni, Daniele Ottaviani, Andrea Beccari, and Gianluca Palermo. Molecular docking via weighted subgraph isomorphism on quantum annealers. *arXiv preprint arXiv:2405.06657*, 2025.

[173] Qi-Ming Ding, Yi-Ming Huang, and Xiao Yuan. Molecular docking via quantum approximate optimization algorithm. *Phys. Rev. Appl.*, 21:034036, Mar 2024.

[174] Wei-Yin Chiang, Po-Yu Kao, Tzu-Lan Yeh, Ya-Chu Yang, Yen-Chu Lin, and Alex Zhavoronkov. Enhancing drug discovery: Quantum machine learning for qsar prediction with incomplete data. *arXiv preprint arXiv:2501.13395*, 2025.

[175] Stefano Mensa, Emre Sahin, Francesco Tacchino, Panagiotis Kl Barkoutsos, and Ivano Tavernelli. Quantum machine learning framework for virtual screening in drug discovery: a prospective quantum advantage. *Machine Learning: Science and Technology*, 4(1):015023, February 2023.

[176] Anthony M. Smaldone and Victor S. Batista. Quantum-to-classical neural network transfer learning applied to drug toxicity prediction. *Journal of Chemical Theory and Computation*, 20(11):4901–4908, 2024.

[177] Pranav Chandarana, Narendra N. Hegade, Iraitz Montalban, Enrique Solano, and Xi Chen. Digitized counterdiabatic quantum algorithm for protein folding. *Phys. Rev. Appl.*, 20:014024, Jul 2023.

[178] Anton Robert, Panagiotis Kl. Barkoutsos, Stefan Woerner, and Ivano Tavernelli. Resource-efficient quantum algorithm for protein folding. *npj Quantum Information*, 7(1):38, 2021.

[179] Anthony Chagneau, Yousra Massaoudi, Imene Derbali, and Linda Yahiaoui. Quantum algorithm for bioinformatics to compute the similarity between proteins. *IET Quantum Communication*, 5(4):417–442, 2024.

[180] Mauro D'Arcangelo, Louis-Paul Henry, Loïc Henriet, Daniele Loco, Nicolaï Gouraud, Stanislas Angebault, Jules Sueiro, Jérôme Forêt, Pierre Monmarché, and Jean-Philip Piquemal. Leveraging analog quantum computing with neutral atoms for solvent configuration prediction in drug discovery. *Phys. Rev. Res.*, 6:043020, Oct 2024.

[181] W. Kohn and L. J. Sham. Self-consistent equations including exchange and correlation effects. *Phys. Rev.*, 140:A1133–A1138, Nov 1965.

[182] V. Fock. Näherungsmethode zur lösung des quantenmechanischen mehrkörperproblems. *Zeitschrift für Physik*, 61(1):126–148, 1930.

[183] Kishor Bharti and Tobias Haug. Quantum-assisted simulator. *Phys. Rev. A*, 104:042418, Oct 2021.

[184] L. Domingo, M. Djukic, C. Johnson, and F. Borondo. Binding affinity predictions with hybrid quantum-classical convolutional neural networks. *Scientific Reports*, 13(1):17951, 2023.

[185] Sabrina Maniscalco, Elsi-Mari Borrelli, Daniel Cavalcanti, Caterina Foti, Adam Glos, Mark Goldsmith, Stefan Knecht, Keijo Korhonen, Joonas Malmi, Anton Nykänen, Matteo A. C. Rossi, Harto Saarinen, Boris Sokolov, N. Walter Talarico, Jussi Westergren, Zoltán Zimborás, and Guillermo García-Pérez. Quantum network medicine: rethinking medicine with network science and quantum algorithms. *arXiv preprint arXiv:2206.12405*, 2022.

[186] Román Orús, Samuel Mugel, and Enrique Lizaso. Quantum computing for finance: Overview and prospects. *Reviews in Physics*, 4:100028, November 2019.

[187] Daniel J. Egger, Claudio Gambella, Jakub Marecek, Scott McFaddin, Martin Mevissen, Rudy Raymond, Andrea Simonetto, Stefan Woerner, and Elena Yndurain. Quantum computing for finance: State-of-the-art and future prospects. *IEEE Transactions on Quantum Engineering*, 1:1–24, 2020.

[188] Patrick Rebentrost and Seth Lloyd. Quantum computational finance: Quantum algorithm for portfolio optimization. *KI - Künstliche Intelligenz*, 38(4):327–338, 2024.

[189] Stefan Woerner and Daniel J. Egger. Quantum risk analysis. *npj Quantum Information*, 5(1):15, 2019.

[190] Daniel J. Egger, Ricardo Garcia Gutierrez, Jordi Cahue Mestre, and Stefan Woerner. Credit Risk Analysis Using Quantum Computers . *IEEE Transactions on Computers*, 70(12):2136–2145, December 2021.

[191] Patrick Rebentrost, Brajesh Gupt, and Thomas R. Bromley. Quantum computational finance: Monte carlo pricing of financial derivatives. *Phys. Rev. A*, 98:022321, Aug 2018.

[192] Nikitas Stamatopoulos, Daniel Egger, Yue Sun, Christa Zoufal, Raban Iten, Ning Shen, and Stefan Woerner. Option pricing using quantum computers. *Quantum*, 4:291, 2020.

[193] Ana Martin, Bruno Candelas, Ángel Rodriguez-Rozas, José D Martin-Guerrero, Xi Chen, Lucas Lamata, Román Orús, Enrique Solano, and Mikel Sanz. Toward pricing financial derivatives with an ibm quantum computer. *Physical Review Research*, 3(1):013167, 2021.

[194] Olawale Titiloye and Alan Crispin. Quantum annealing of the graph coloring problem. *Discrete Optimization*, 8(2):376–384, May 2011.

[195] David Horn and Assaf Gottlieb. Algorithm for data clustering in pattern recognition problems based on quantum mechanics. *Phys. Rev. Lett.*, 88:018702, Dec 2001.

[196] M. Weinstein, F. Meirer, A. Hume, Ph. Sciau, G. Shaked, R. Hofstetter, E. Persi, A. Mehta, and D. Horn. Analyzing big data with dynamic quantum clustering. *arXiv preprint arXiv:1310.2700*, 2013.

[197] Santanu Ganguly. Implementing quantum generative adversarial network (qgan) and qcbm in finance. *arXiv preprint arXiv:2308.08448*, 2025.

[198] Yen-Jui Chang, Wei-Ting Wang, Hao-Yuan Chen, Shih-Wei Liao, and Ching-Ray Chang. Preparing random state for quantum financing with quantum walks. *arXiv preprint arXiv:2302.12500*, 2023.

[199] Mingyu Lee, Myeongjin Shin, Junseo Lee, and Kabgyun Jeong. Mutual information maximizing quantum generative adversarial network and its applications in finance. *arXiv preprint arXiv:2309.01363*, 2023.

[200] Alok Shukla and Prakash Vedula. A hybrid classical-quantum algorithm for solution of nonlinear ordinary differential equations. *Applied Mathematics and Computation*, 442:127708, April 2023.

[201] Jonas Stein, Ivo Christ, Nicolas Kraus, Maximilian Balthasar Mansky, Robert Müller, and Claudia Linnhoff-Popien. Applying qnlp to sentiment analysis in finance. In *2023 IEEE International Conference on Quantum Computing and Engineering (QCE)*, page 20–25. IEEE, September 2023.

[202] Marco Pistoia, Syed Farhan Ahmad, Akshay Ajagekar, Alexander Buts, Shouvanik Chakrabarti, Dylan Herman, Shaohan Hu, Andrew Jena, Pierre Minssen, Pradeep Niroula, Arthur Rattew, Yue Sun, and Romina Yalovetzky. Quantum machine learning for finance. *arXiv preprint arXiv:2109.04298*, 2021.

[203] Dylan Herman, Cody Googin, Xiaoyuan Liu, Alexey Galda, Ilya Safro, Yue Sun, Marco Pistoia, and Yuri Alexeev. A survey of quantum computing for finance. *arXiv preprint arXiv:2201.02773*, January 2022.

[204] Abha Satyavan Naik, Esra Yeniaras, Gerhard Hellstern, Grishma Prasad, and Sanjay Kumar Lalta Prasad Vishwakarma. From portfolio optimization to quantum blockchain and security: a systematic review of quantum computing in finance. *Financial Innovation*, 11(1):88, 2025.

[205] Cameron Perot. Quantum boltzmann machines: Applications in quantitative finance. *arXiv preprint arXiv:2301.13295*, 2023.

[206] Filipe Fontanela, Antoine Jacquier, and Mugad Oumgari. Short communication: A quantum algorithm for linear pdes arising in finance. *SIAM Journal on Financial Mathematics*, 12(4):SC98–SC114, 2021.

[207] S. Pirandola, U. L. Andersen, L. Banchi, M. Berta, D. Bunandar, R. Colbeck, D. Englund, T. Gehring, C. Lupo, C. Ottaviani, J. L. Pereira, M. Razavi, J. Shamsul Shaari, M. Tomamichel, V. C. Usenko, G. Vallone, P. Villoresi, and P. Wallden. Advances in quantum cryptography. *Advances in Optics and Photonics*, 12(4):1012, December 2020.

[208] Yang Liu, Wei-Jun Zhang, Cong Jiang, Jiu-Peng Chen, Chi Zhang, Wen-Xin Pan, Di Ma, Hao Dong, Jia-Min Xiong, Cheng-Jun Zhang, Hao Li, Rui-Chun Wang, Jun Wu, Teng-Yun Chen, Lixing You, Xiang-Bin Wang, Qiang Zhang, and Jian-Wei Pan. Experimental twin-field quantum key distribution over 1000 km fiber distance. *Phys. Rev. Lett.*, 130:210801, May 2023.

[209] Mirko Pittaluga, Mariella Minder, Marco Lucamarini, Mirko Sanzaro, Robert I. Woodward, Ming-Jun Li, Zhiliang Yuan, and Andrew J. Shields. 600-km repeater-like quantum communications with dual-band stabilization. *Nature Photonics*, 15(7):530–535, July 2021.

[210] Jiu-Peng Chen, Chi Zhang, Yang Liu, Cong Jiang, Wei-Jun Zhang, Zhi-Yong Han, Shi-Zhao Ma, Xiao-Long Hu, Yu-Huai Li, Hui Liu, Fei Zhou, Hai-Feng Jiang, Teng-Yun Chen, Hao Li, Li-Xing You, Zhen Wang, Xiang-Bin Wang, Qiang Zhang, and Jian-Wei Pan. Twin-field quantum key distribution over a 511 km optical fibre linking two distant metropolitan areas. *Nature Photonics*, 15(8):570–575, 08 2021.

[211] Alberto Boaron, Gianluca Boso, Davide Rusca, Cédric Vulliez, Claire Autebert, Misael Caloz, Matthieu Perrenoud, Gaëtan Gras, Félix Bussières, Ming-Jun Li,

Daniel Nolan, Anthony Martin, and Hugo Zbinden. Secure quantum key distribution over 421 km of optical fiber. *Phys. Rev. Lett.*, 121:190502, Nov 2018.

[212] Nishant Kumar Pathak, Sumit Chaudhary, Sangeeta, and Bhaskar Kanseri. Phase encoded quantum key distribution up to 380 km in standard telecom grade fiber enabled by baseline error optimization. *Scientific Reports*, 13(1):15868, 2023.

[213] Koji Azuma, Sophia E. Economou, David Elkouss, Paul Hilaire, Liang Jiang, Hoi-Kwong Lo, and Ilan Tzitrin. Quantum repeaters: From quantum networks to the quantum internet. *Rev. Mod. Phys.*, 95:045006, Dec 2023.

[214] Costantino Carugno, Maurizio Ferrari Dacrema, and Paolo Cremonesi. Evaluating the job shop scheduling problem on a d-wave quantum annealer. *Scientific Reports*, 12(1):6539, 2022.

[215] Sean J. Weinberg, Fabio Sanches, Takanori Ide, Kazumitzu Kamiya, and Randall Correll. Supply chain logistics with quantum and classical annealing algorithms. *Scientific Reports*, 13(1):4770, March 2023.

[216] José Francisco Ariño Sales and Rodrigo Andrés Palacios Araos. Adiabatic quantum computing impact on transport optimization in the last-mile scenario. *Frontiers in Computer Science*, 5:1294564, 2023.

[217] Sebastián V. Romero, Eneko Osaba, Esther Villar-Rodriguez, Izaskun Oregi, and Yue Ban. Hybrid approach for solving real-world bin packing problem instances using quantum annealers. *Scientific Reports*, 13(1):11777, 2023.

[218] Archismita Dalal, Iraitz Montalban, Narendra N. Hegade, Alejandro Gomez Cadavid, Enrique Solano, Abhishek Awasthi, Davide Vodola, Caitlin Jones, Horst Weiss, and Gernot Füchsel. Digitized counterdiabatic quantum algorithms for logistics scheduling. *Physical Review Applied*, 22:064068, 2025.

[219] Randall Correll, Sean J. Weinberg, Fabio Sanches, Takanori Ide, and Takafumi Suzuki. Quantum neural networks for a supply chain logistics application. *Advanced Quantum Technologies*, 6(7):2200183, 2023.

[220] Fristi Riandari, Aisyah Alesha, and Hengki Tamando Sihotang. Quantum computing for production planning. *International Journal of Enterprise Modelling*, 15(3):163–175, Sep. 2021.

[221] Yongcheng Ding, Xi Chen, Lucas Lamata, Enrique Solano, and Mikel Sanz. Implementation of a hybrid classical-quantum annealing algorithm for logistic network design. *SN Computer Science*, 2(2):68, 2021.

[222] Yukun Zhang, Xiaoming Zhang, Jinzhao Sun, Heng Lin, Yifei Huang, Dingshun Lv, and Xiao Yuan. Quantum algorithms for quantum molecular systems: A survey. *WIREs Computational Molecular Science*, 15(3):e70020, 2025.

[223] EJ Crosson and DA Lidar. Prospects for quantum enhancement with diabatic quantum annealing. *Nature Reviews Physics*, 3(7):466–489, 2021.

[224] Helmut G. Katzgraber, Firas Hamze, Zheng Zhu, Andrew J. Ochoa, and H. Munoz-Bauza. Seeking quantum speedup through spin glasses: The good, the bad, and the ugly. *Phys. Rev. X*, 5:031026, Sep 2015.

[225] Troels F. Rønnow, Zhihui Wang, Joshua Job, Sergio Boixo, Sergei V. Isakov, David Wecker, John M. Martinis, Daniel A. Lidar, and Matthias Troyer. Defining and detecting quantum speedup. *Science*, 345(6195):420–424, July 2014.

[226] Andrew D. King, Alberto Nocera, Marek M. Rams, Jacek Dziarmaga, Roeland Wiersema, William Bernoudy, Jack Raymond, Nitin Kaushal, Niclas Heinsdorf, Richard Harris, Kelly Boothby, Fabio Altomare, Mohsen Asad, Andrew J. Berkley, Martin Boschnak, Kevin Chern, Holly Christiani, Samantha Cibere, Jake Connor, Martin H. Dehn, Rahul Deshpande, Sara Ejtemaee, Pau Farre, Kelsey Hamer, Emile Hoskinson, Shuiyuan Huang, Mark W. Johnson, Samuel Kortas, Eric Ladizinsky, Trevor Lanting, Tony Lai, Ryan Li, Allison J. R. MacDonald, Gaelen Marsden, Catherine C. McGeoch, Reza Molavi, Travis Oh, Richard Neufeld, Mana Norouzpour, Joel Pasvolsky, Patrick Poitras, Gabriel Poulin-Lamarre, Thomas Prescott, Mauricio Reis, Chris Rich, Mohammad Samani, Benjamin Sheldan, Anatoly Smirnov, Edward Sterpka, Berta Trullas Clavera, Nicholas Tsai, Mark Volkmann, Alexander M. Whiticar, Jed D. Whittaker, Warren Wilkinson, Jason Yao, T. J. Yi, Anders W. Sandvik, Gonzalo Alvarez, Roger G. Melko, Juan Carrasquilla, Marcel Franz, and Mohammad H. Amin. Beyond-classical computation in quantum simulation. *Science*, 388(6743):199–204, apr 2025.

[227] Finley Alexander Quinton, Per Arne Sevle Myhr, Mostafa Barani, Pedro Crespo del Granado, and Hongyu Zhang. Quantum annealing applications, challenges and limitations for optimisation problems compared to classical solvers. *Scientific Reports*, 15(1):12733, apr 2025.

[228] Dolev Bluvstein, Simon J. Evered, Alexandra A. Geim, Sophie H. Li, Hengyun Zhou, Tom Manovitz, Sepehr Ebadi, Madelyn Cain, Marcin Kalinowski, Dominik Hangleiter, J. Pablo Bonilla Ataides, Nishad Maskara, Iris Cong, Xun Gao, Pedro Sales Rodriguez, Thomas Karolyshyn, Giulia Semeghini, Michael J. Gullans, Markus Greiner, Vladan Vuletić, and Mikhail D. Lukin. Logical quantum processor based on reconfigurable atom arrays. *Nature*, 626(7997):58–65, Feb 2024.

[229] Youngseok Kim, Andrew Eddins, Sajant Anand, Ken Xuan Wei, Ewout van den Berg, Sami Rosenblatt, Hasan Nayfeh, Yantao Wu, Michael Zaletel, Kristan Temme, and Abhinav Kandala. Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618(7965):500–505, Jun 2023.

[230] Jens Eisert and John Preskill. Mind the gaps: The fraught road to quantum advantage. *arXiv preprint arXiv:2510.19928*, 2025.

[231] Maria Schuld and Nathan Killoran. Quantum machine learning in feature hilbert spaces. *Phys. Rev. Lett.*, 122:040504, Feb 2019.