A Survey on Quantum Machine Learning: Basics, Current Trends, Challenges, Opportunities, and the Road Ahead

Kamila Zaman[1,2], Alberto Marchisio[1,2], Muhammad Abdullah Hanif[1,2], and Muhammad Shafique[1,2]
[1]eBrain Lab, Division of Engineering, New York University Abu Dhabi, UAE
[2]Center for Quantum and Topological Systems (CQTS), NYUAD Research Institute, New York University Abu Dhabi, UAE
(e-mail: kz2137@nyu.edu, alberto.marchisio@nyu.edu, mh6117@nyu.edu, muhammad.shafique@nyu.edu)

***Abstract*—Quantum Computing (QC) claims to improve the efficiency of solving complex problems, compared to classical computing. When QC is integrated with Machine Learning (ML), it creates a Quantum Machine Learning (QML) system. This paper aims to provide a thorough understanding of the foundational concepts of QC and its notable advantages over classical computing. Following this, we delve into the key aspects of QML in a detailed and comprehensive manner.**

In this survey, we investigate a variety of QML algorithms, discussing their applicability across different domains. We examine quantum datasets, highlighting their unique characteristics and advantages. The survey also covers the current state of hardware technologies, providing insights into the latest advancements and their implications for QML. Additionally, we review the software tools and simulators available for QML development, discussing their features and usability.

Furthermore, we explore practical applications of QML, illustrating how it can be leveraged to solve real-world problems more efficiently than classical ML methods. This paper serves as a valuable resource for readers seeking to understand the current state-of-the-art techniques in the QML field, offering a solid foundation to embark on further exploration and development in this rapidly evolving area.

***Index Terms*—Quantum Computing, Quantum Computer, Quantum Machine Learning, Quantum Neural Networks, Machine Learning, Neural Networks, Quantum Supremacy, Qubit, Superposition, Quantum Correlation, Entanglement, Quantum Gate, Quantum Circuit, Quantum Noise, Noisy Intermediate-Scale Quantum, Fault-Tolerant Quantum Computing, Parametrized Quantum Circuits, Quantum Annealing, Quantum Kernels, Variational Quantum Eigensolver, Quantum Data, Quantum Encoding, Quantum Datasets, Quantum Hardware, Quantum Simulator, Qiskit, PennyLane, Quantum Applications.**

I. Introduction

Machine Learning (ML) systems are well-established tools for identifying patterns in data and generalizing complex, nonlinear problems. These systems have found applications in various domains, including computer vision, healthcare, finance, and the automotive industry. However, ML practitioners must navigate the substantial computing resources required to run large ML models. Despite employing numerous hardware-aware optimizations such as compression and approximations [1]–[5], the limitations of current computing infrastructures and technologies constrain the computational capabilities of ML systems.

The high computational demands of modern ML models necessitate advanced hardware development. According to Moore's law, the number of transistors on an integrated circuit doubles approximately every two years. However, this trend has reached its limits with current high-end CPUs and GPUs [6]. This physical saturation restricts computational power, leading to delays in processing, developmental, and scientific discovery within the ML community. For instance, training Large Language Models with hundreds of billions of parameters and trillions of tokens is extremely compute-intensive [7]. Such tasks require massive investments in time and hardware resources, which only a few high-end companies can afford. To overcome these physical limits and support further discoveries, there is an urgent need to explore new technological avenues of hardware systems that can enhance the computational efficiency for solving given real-world problems by closely simulating and comprehending them.

One of the most promising solutions to this bottleneck is Quantum Computing (QC). Initially proposed by Feynman [31] and further developed by Preskill [32], [33], QC systems exploit quantum mechanical phenomena to significantly improve performance and information processing compared to classical systems. Unlike classical systems that struggle to encompass natural processes, quantum computers operate on similar principles to those found in nature. This similarity suggests that QC could help us better understand natural phenomena. Moreover, QC has the potential to solve problems more cost-efficiently through optimizations, time savings, and environmentally friendly designs.

The Quantum Machine Learning (QML) paradigm represents an excellent opportunity for researchers and industries to achieve remarkable discoveries and design efficient solutions for complex real-world problems. QML systems, driven towards practicality and improved performance over classical systems, open new avenues for the community to discover, build, and align their designs across different levels of the quantum stack [34]. This integration of QC and ML could lead to groundbreaking advancements and a deeper understanding of the world around us.

TABLE I: Qualitative comparison between our work and related QML surveys.

QML Survey	Year	# Pages	# Refs.	QC Details	QC Advantages	QML Algs.	Datasets/Encoding	HW Techs.	SW Tools	Applications
[8]	2006	12	24	✓	✗	✓	✓	✗	✗	✗
[9]	2014	19	71	✗	✓	✓	✗	✗	✗	✗
[10]	2015	38	97	✗	✗	✓	✓	✗	✗	✓
[11]	2016	17	100	✗	✓	✓	✗	✓	✗	✓
[12]	2017	106	309	✓	✓	✓	✓	✗	✗	✓
[13]	2018	33	187	✓	✓	✓	✗	✗	✗	✓
[14]	2019	15	74	✓	✗	✓	✗	✗	✗	✓
[15]	2019	29	309	✓	✓	✓	✗	✓	✗	✓
[16]	2020	8	71	✓	✗	✓	✗	✗	✗	✗
[17]	2020	6	47	✗	✗	✓	✗	✗	✗	✓
[18]	2020	17	165	✓	✗	✓	✗	✗	✗	✓
[19]	2021	9	75	✗	✓	✓	✓	✗	✗	✗
[20]	2021	4	41	✓	✗	✓	✗	✗	✗	✓
[21]	2021	14	73	✗	✓	✓	✗	✗	✗	✗
[22]	2022	6	21	✗	✗	✓	✗	✗	✗	✗
[23]	2022	17	154	✓	✗	✓	✗	✗	✗	✓
[24]	2022	12	59	✗	✗	✓	✗	✗	✓	✗
[25]	2023	19	116	✗	✗	✓	✓	✓	✓	✓
[26]	2023	49	199	✓	✓	✓	✓	✗	✗	✓
[27]	2023	337	851	✓	✓	✓	✓	✗	✗	✓
[28]	2023	59	301	✗	✓	✓	✓	✗	✓	✓
[29]	2023	21	87	✓	✓	✓	✓	✗	✗	✓
[30]	2024	44	42	✓	✓	✓	✓	✗	✗	✗
Our work	2024	35	187	✓	✓	✓	✓	✓	✓	✓

A. Scope and Contributions

In this work, we explore the extent to which the overlap between ML and QC has been investigated and identify the future potential of this intersection. This paper presents a systematic and critical discussion of the current status and future perspectives in the field of QML. Our aim is to provide readers with a comprehensive understanding of QML by detailing the currently available algorithms, frameworks, technologies, and tools.

The primary goal of this survey is to introduce and provide substantial knowledge on QML, laying a solid foundation for solving new and advanced problems. In contrast to prior QML surveys [8]–[30], we provide a well-structured and thoroughly comprehensive collection of information and concepts that summarize the current state of development of QML. Table I illustrates the comparison between this work and a selection of related QML surveys, highlighting the features and topics each covers. It becomes evident from the table that previous surveys have at least one missing feature, whereas this work delivers a detailed review encompassing all key aspects of QML. **The contributions of this paper are summarized below.**

- **Detailed Overview of Quantum Computers:** We present an in-depth analysis of quantum computers, addressing current challenges, existing techniques, and ongoing efforts to develop viable solutions.
- **Review of State-of-the-Art QML Algorithms:** We review the latest QML algorithms, encoding techniques, datasets, hardware technologies, software tools, and their applications, providing a comprehensive understanding of the field's current capabilities.
- **Critical Discussion of Open Research Challenges:** We offer a critical examination of the open research challenges and future directions in the QML field, discussing its practical potential and the steps needed to advance the technology further.

By encompassing these areas, this paper serves as a valuable resource for researchers and practitioners, offering a thorough overview of the current landscape and future possibilities in QML.

B. Outline

This paper is organized into different sections and subsections. Section I introduces the problem, the scope, and the contributions of this paper. Section II presents foundational theoretical concepts of general quantum computers, the challenges of the current technologies, and the state-of-the-art methodologies to overcome these issues. Section III provides an overview of the claimed and proven advantages of QC compared to classical computing. This section discusses the motivations that are driving researchers and industries to design QC and QML systems. Section IV discusses the plethora of the most common QML algorithms that are present in the literature and provides an overview of their applicability. Section V provides the necessary information to understand the existing methods for manipulating quantum data in a way that it can be used in the QML workflow, along with a curated list of dataset resources from the QML perspective. Section VI introduces the existing tools and technologies that are available for QML practitioners to experiment with and investigate further. Section VII presents the current and potential applications of QML that majorly highlight their benefits over classical computing. Section VIII concludes the paper and provides future outlooks in the QML field.

II. Quantum Computing Preliminaries

This section provides an overview of the fundamental concepts in the field of quantum systems. It explains the core features and characteristics of QC systems to build a strong conceptual model of quantum computers, with necessary details for readers to understand the rest of the paper.

A. *Understanding the Qubit*

The bit forms the fundamental processing unit of a classical computing system. As shown in Figure 1a, it can hold one of the two mutually exclusive values, i.e., either 0 or 1 at one instance of time, representing classical computation and classical information. On the other hand, a quantum computer's fundamental unit for quantum computation and quantum information is a *quantum bit* or *qubit*, which can exist in a continuum of states (basis states).

Understanding the qubit as an abstract mathematical object is crucial for the development of quantum circuits. Although the physical realization of qubits is important, the fundamental properties and specifications of qubits are dictated by quantum mechanical principles and remain consistent regardless of their physical form. Unlike a classical bit, which can only be in one of two states (0 or 1) at any given time, a qubit can exist in a superposition of both 0 and 1 simultaneously. This means that a qubit can exist in a superposition of both 0 and 1 states simultaneously (see Figure 1b). This quantum mechanical property of representing any qubit state as a linear combination of basis states, called *superposition*, allows a qubit to embody a range of probabilities for being in either state, reflecting its inherent quantum nature. This abstract representation of a qubit is consistent across any physical realization of the qubit, whether it is implemented using superconducting circuits, trapped ions, or any other technology.

On the other hand, the resultant state of the quantum computation system's processing is dictated by the probabilistic occurrence of the collective system's states as a solution over multiple experiment sample runs (called *shots*). For each shot, the resulting solution over the *super-positioned* states, at the final computational stage of the system circuit after state manipulation (quantum processing) using a set of gates, passes through the measurement operation to form the measured state. It is represented through the basis states $|0\rangle$ and $|1\rangle$, followed by the final probabilistic distribution over possible measured state outcomes as solutions to the problem [35].

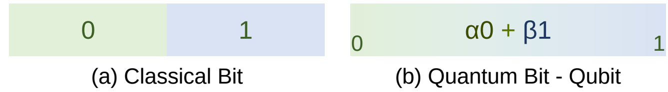

Fig. 1: **(a)** Classical bit vs. **(b)** Quantum Bit with α and β as respective amplitude values of a state-vector, creating a superposition state where $|\alpha|^2 + |\beta|^2 = 1$ as per the Max-Born Rule, satisfying the completeness equation which states to have all probabilities summing up to 1 [36]. $|\alpha|^2$ and $|\beta|^2$ give the respective probability for each state.

The Bloch sphere is an intuitive visual representation of a two-level (two states per qubit) single qubit system. As shown in Figure 2, it represents the pure and mixed states of a qubit. The pure states are distributed over the surface of the sphere, and all the internal states within the sphere are referred to as mixed-qubit states. It has three axes with their respective basis states on opposite ends.

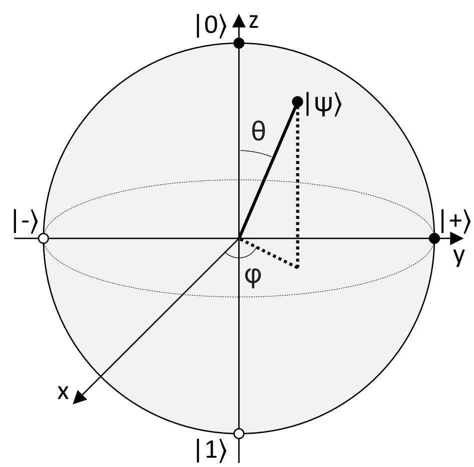

Fig. 2: Bloch sphere for a single two-level qubit system.

B. *Superposition*

Before delving into the details of the superposition principle, we discuss two fundamental concepts of quantum mechanics, the *Wave–Particle duality* and the *Max-Born rule*.

The **Wave-Particle duality** theory posits that waves can exhibit particle-like properties and particles can exhibit wave-like properties. This duality is a cornerstone of quantum mechanics, contrasting sharply with the principles of classical mechanics or Newtonian physics [37]. In classical mechanics, waves and particles are distinct entities with no overlap in their properties. Quantum mechanics, however, reveals that entities such as photons and electrons can behave both as particles and as waves, depending on the experimental setup. This dual nature is exemplified in phenomena such as the double-slit experiment, where particles like electrons create an interference pattern characteristic of waves when not observed, but act like particles when observed.

The **Max-Born rule** provides a probabilistic framework for quantum mechanics. It states that the probability density of finding a quantum system in a given state is proportional to the square of the amplitude of the system's wave function at that state. This is usually represented within a state-vector formalism in Hilbert space. The rule is fundamental to the measurement process in quantum mechanics [38]. When a qubit is measured, the probability of finding it in a particular state is determined by the square of the amplitude of its wave function for that state.

The **superposition** principle is one of the most intriguing and counterintuitive aspects of quantum mechanics. It states that a quantum system can exist in multiple states simultaneously until it is measured. The space in which these states exist is called *Hilbert space*, a mathematical framework that allows the description of quantum states as vectors. A common physical interpretation of the superposition principle involves the spin of a particle, which can be in a superposition of "spin up" and "spin down" states or any combination thereof. Similarly, the energy levels of free electrons in an atom illustrate superposition. An electron can exist in a superposition of the ground state ($|0\rangle$) and an excited state ($|1\rangle$), with possible intermediate states.

In the context of qubits, superposition means that a qubit can represent both 0 and 1 simultaneously, allowing quantum computers to process a vast amount of information in parallel. This property is what gives quantum computers their potential to solve certain complex problems much more efficiently than classical computers.

While the physical realization of qubits is key for the development of quantum circuits, it is more important to understand qubits as abstract mathematical objects. These objects adhere to quantum mechanical principles, which remain consistent regardless of the physical form used for qubit realization. A qubit can be mathematically described as a linear combination (or superposition) of its basis states, $|0\rangle$ and $|1\rangle$. This abstract representation is crucial for designing and understanding quantum algorithms and systems, as it encapsulates the fundamental quantum properties such as superposition and entanglement.

The wave-particle duality and the Max-Born Rule are foundational concepts that lead to a deeper understanding of the superposition principle. The superposition principle, in turn, is essential for grasping the unique capabilities of qubits, which are the building blocks of quantum computing. Understanding these principles not only clarifies the theoretical underpinnings of quantum mechanics but also informs the practical development of quantum technologies.

1) Single Qubit Superposition: As illustrated in Figure 3, the state of the qubit is composed of a mixed presence of state $|0\rangle$ and state $|1\rangle$, i.e., with superposition, in which the amplitude values can be obtained after measurement. For each independent quantum system, the probabilities associated with the measured states are illustrated as separate blocks. This behavior is dictated quantitatively by the amplitude value in context to the wave function driven by the underlying quantum mechanics of the system. Using the Max-Born rule, we sum the squared norm of all amplitudes of the superposition states, which in our example are α and β. Converting the states into probabilities makes the system more intuitive and interpretable.

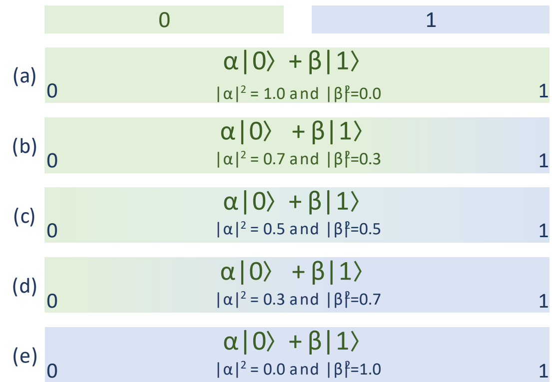

Fig. 3: A few examples for single qubit Superposition cases. Each block represents an independent quantum system visualizing the resultant measured state of the system with gradient depicting probabilities computed from amplitude values α and β of respective superpositions. A superposition of a single qubit **(a)** fully measured to be state $|0\rangle$. **(b)** with 70% probability to be measured in state $|0\rangle$ and 30% for state $|1\rangle$. **(c)** Equal probability to be measured in state $|0\rangle$ and state $|1\rangle$. **(d)** with 30% probability to be measured in state $|0\rangle$ and 70% for state $|1\rangle$. **(e)** Fully measured to be in state $|1\rangle$.

Quantum physics uses state-vectors to describe the state of the system. Figure 3 illustrates various superposition states possible for a single qubit system resulting in probabilities. Currently, there are 5 different arbitrarily chosen superposition states shown for a single qubit, where each block is an independent qubit system with its own corresponding state-vector containing amplitude values α and β. We can observe that the amplitudes of α and β can hold any value from the complex number space defining the quantum mechanical system. However, for practical use of holding and processing information, some rules apply to the amplitude values in the state-vector. According to the Max-Born rule, all amplitude values in the state-vector are such that the sum of their squared norm is always equal to 1. In other words, we observe from the Max-Born rule that amplitudes and probabilities are related to each other. To use the probabilities, we should satisfy the property that their sum is equal to 1. To meet these conditions, the amplitude values of the state-vector must be normalized.

2) Two-Qubit Superposition: Let us now take one step forward and visualize the superposition samples for a two-qubit system. In Figure 4 and Figure 5, all the principles involved in the system are the same as those applied in a single qubit. Still, now the quantum system can exist in 4 (in general, 2^n for an n-qubit system) superpositioned states with 2 qubits at once in the system with respective amplitudes for each state until the measurement operation is performed to determine the solution state. Each figure shows one independent quantum system with various examples of superpositions to elaborate the idea. Equation (1) also illustrates that every state has a corresponding amplitude value (α, β, γ, η), whose squared value corresponds to the probability of that state being the measured state.

$$system\ state = \alpha\,|00\rangle + \beta\,|01\rangle + \gamma\,|10\rangle + \eta\,|11\rangle \quad (1)$$

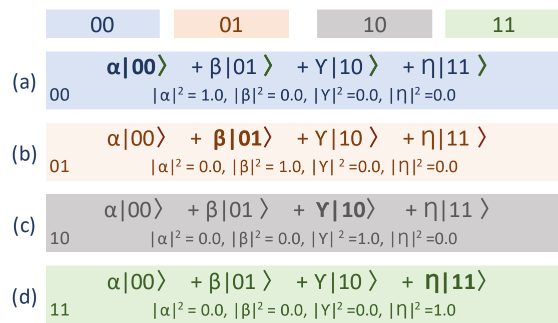

Fig. 4: A few examples of two-qubit superposition cases depicting measured outcomes. **(a)** State $|00\rangle$ as measured state with 100% probability. **(b)** State $|01\rangle$ as measured state with 100% probability. **(c)** State $|10\rangle$ as measured state with 100% probability. **(d)** State $|11\rangle$ as measured state with 100% probability.

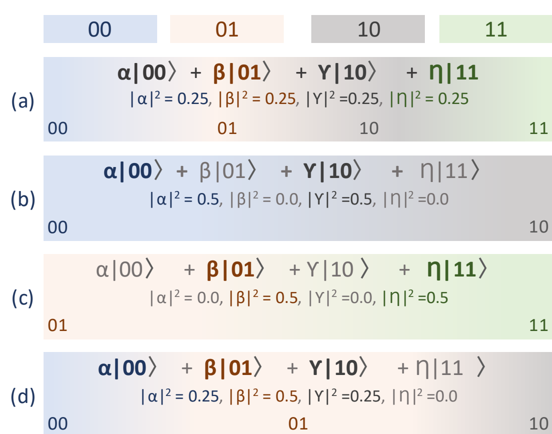

Fig. 5: Other examples of two-qubit superposition cases. **(a)** 25% equal probability to be measured in states $|00\rangle$, $|01\rangle$, $|10\rangle$, and $|11\rangle$. **(b)** 50% probability to be measured in state $|00\rangle$ and 50% for state $|10\rangle$. **(c)** 50% probability to be measured in state $|01\rangle$ and 50% for state $|11\rangle$. **(d)** 25% probability to be measured in state $|00\rangle$, 50% for state $|01\rangle$, and 25% for state $|10\rangle$.

To understand the basic difference between classical and quantum systems, let us consider an analogy that presents a problem to determine the position (A, B, C, or D equivalent to states 00, 01, 10, and 11) of an object at a given instance. Figure 6 shows an example of how the solution approaches differ for classical and quantum systems. Suppose that every state corresponds to the position of the object. In that case, the classical system determines the object's position by sequentially checking each of the possible states of the solution and validating whether the object is present. It ends up with one single state as the final solution, which in the example is position B (state 01), with 100% probability that the object is at that position (state).

On the contrary, a quantum system for the same problem will result in the same solution for identifying the position (state) of the object, but the approach is different. Based on quantum mechanics principles, the quantum system identifies the solution that results in a probabilistic distribution of the possible solutions to the given problem, i.e., in this case, the probabilistic positions of the object's placement. The quantum system's solution is that the object is at position B (state 01). However, it is only 60% confident that it is there and considers other possible outcomes. The mechanisms that quantum systems follow to understand and find the solution are much closer to the real world than classical systems.

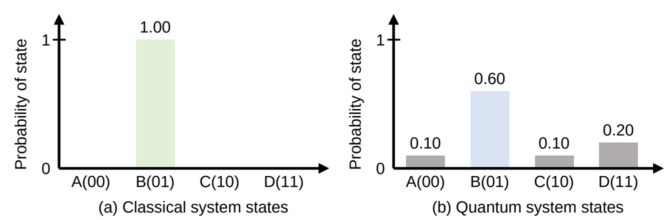

Fig. 6: Example of the probability of states at one point in time for **(a)** a classical system and **(b)** a quantum system (over n experiments/shots). This example helps to understand the differences between the two systems. We assume that each state shows a particular position where an object can be, i.e., positions A, B, C, or D. Both classical and quantum computers give the same solution. The main difference is that the classical system finds the solution after sequential iterations on each possible state. On the other hand, the quantum system gives a probability distribution of where the object is positioned. Both ultimately convey that the position of the object is B.

C. *Quantum Gates*

Quantum logic gates are one of the essential parts of a quantum computer and are the building blocks of all quantum algorithms. These gates are mathematically described by unitary matrices, and their action is always logically reversible. The gates correspond to operators used to manipulate the quantum state of qubits. The only condition on a valid gate (and its corresponding quantum operator) is that it should be unitary. By unitary we mean that it meets the $U^\dagger = U^{-1}$ condition, where U is the gate operation matrix and $U^\dagger$ is the adjoint of U. This property also ensures that it is a reversible gate operation [39].

D. Quantum Circuits

A quantum circuit is a collection of quantum gates interconnected by quantum wires. The actual structure of a quantum circuit, the number and the types of gates, as well as the interconnection scheme, are dictated by the unitary transformation, U, executed by the circuit [40]. A circuit is an independent module composed of gates, wires, and qubits arranged in a certain way to perform a task. Hence, its functionality is similar to what we refer to as an algorithm. A collection of circuits are independent functional quantum units using the output of a circuit as the input of another one to form a complete algorithm for the solution of the given problem [41] (see Figure 7).

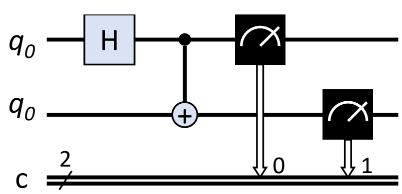

Fig. 7: An example of a quantum circuit depicting input qubits, gates, measurement, and output.

E. Quantum Correlations

Correlation, a term derived from statistics, measures how much knowledge of one part of a system can predict the behavior of another part. In classical mechanics, this correlation is deterministic: knowing the state of one part allows precise prediction of the other part's state. Any observable in a classical system has only one possible outcome, and deviations from this outcome are attributed to measurement errors or inaccuracies.

In contrast, quantum mechanics introduces a probabilistic nature to correlation. Observables in a quantum system do not have a single, definite outcome [42]. Instead, they have a range of possible values, known as *eigenvalues*, which are associated with certain probabilities. These eigenvalues are the result of the system being represented by a *Hermitian matrix*. Thus, even without any perturbation, measurements of a quantum system can yield different results over time. These varying outcomes are described by *eigenfunctions* that exist within a Hilbert space, a mathematical framework that encompasses all possible states of the system.

1) Quantum Entanglement: Quantum entanglement is a central concept in quantum information theory and represents one of the most counterintuitive aspects of quantum mechanics. Entanglement occurs when two particles become linked such that the state of one particle instantaneously influences the state of the other, regardless of the distance separating them [43]–[45]. This phenomenon, which Albert Einstein famously referred to as a "spooky action at a distance", defies the classical notion of local realism [46].

The mechanism of quantum entanglement is illustrated in Figure 8, showing how changes in one particle affect its entangled partner. While entanglement itself is nondeterministic, meaning the specific outcomes cannot be predicted with certainty, achieving deterministic control over entangled states could lead to groundbreaking applications across various fields. Quantum development tools like Qiskit [47] and PennyLane [48] enable researchers to create and manipulate entangled states for solving real-world problems.

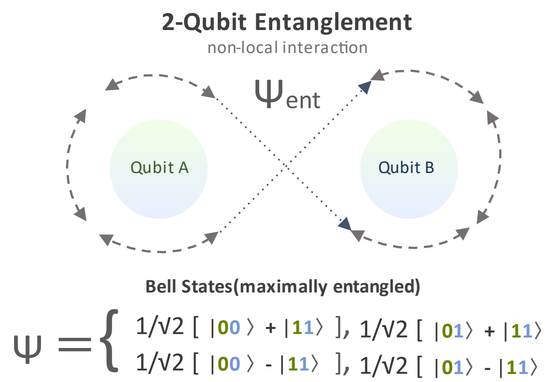

Fig. 8: Overview of two-qubit entanglement. The bell states are four specific maximally entangled (shared) quantum states of two qubits.

2) Quantum Decoherence: A quantum system exhibits *coherence* when there is a well-defined phase relationship between its different states. Coherence is crucial for quantum computing because it determines how long a qubit can maintain its state without being disturbed by external factors. High coherence allows for extended computations, as the qubit can retain its information over the necessary duration for processing.

Quantum *decoherence* is the process by which a quantum system loses its coherence due to interactions with the external environment. This loss of coherence manifests as the system's transition from a pure quantum state to a mixed state, effectively causing a collapse of the wave function [49]. Decoherence can be viewed as the leakage of information from the quantum system into the environment, or as environmental interference disrupting the quantum state.

Decoherence presents a significant challenge in developing scalable and stable quantum computers. It limits the duration for which qubits can reliably perform computations, thereby affecting the overall performance and viability of quantum systems. Addressing and mitigating quantum decoherence is a primary focus in the quest to build practical quantum computers.

F. Quantum Noise

Leading from the idea of coherence and decoherence, *quantum noise* makes the implementation of large-scale and reliable quantum computers with low error rates extremely challenging. Therefore, it is imperative to properly characterize the noise in quantum systems and devise mitigation techniques to detect and correct the errors [50].

1) Uncertainty Principle: Quantum noise can be partially understood through the lens of Heisenberg's uncertainty principle. This principle asserts that certain pairs of physical properties, such as position and momentum, cannot both be known to arbitrary precision simultaneously. The more precisely one property is known, the less precisely the other can be determined. This inherent uncertainty in quantum mechanics leads to quantum noise, which manifests as fluctuations in the quantum state of a system.

Heisenberg's uncertainty principle states that it is fundamentally impossible to simultaneously measure the exact value of certain pairs of related physical quantities, such as position (x) and momentum (p), with infinite precision [51]. Mathematically, this principle is expressed as in Equation (2):

$$\Delta x \cdot \Delta p \geq \frac{\hbar}{2} \tag{2}$$

where Δx is the uncertainty in position, Δp is the uncertainty in momentum, and $\hbar$ is the reduced Planck's constant. This principle highlights the intrinsic limitations in our ability to measure and predict the behavior of quantum particles accurately.

Heisenberg's uncertainty principle provides a fundamental explanation for the presence of quantum noise, which arises from the inherent uncertainties in quantum systems. Qubits, as quantum mechanical entities, are particularly vulnerable to this noise. Addressing these challenges is essential for advancing quantum computing and harnessing its full potential.

2) Noisy Intermediate-Scale Quantum Era: The long-term goal for QC is to develop functional quantum algorithms that can solve problems despite the noisy environmental systems they work in. Completely eliminating the noise errors is extremely difficult. Hence, the community has defined a set of achievable intermediate goals to evolve from Noisy Intermediate-Scale Quantum (NISQ)[1] to Fault Tolerant Quantum Computers (FTQC); see Figure 9.

The current state of quantum computing is referred to as the NISQ era [32]. Current quantum processors only support 50-100's of qubits but are not advanced enough to guarantee complete fault tolerance. Despite that, they represent a valid infrastructure for experimenting and improving the designs towards ideal systems. In the NISQ era, near-term hybrid quantum-classical algorithms are designed and applied to various fields, such as quantum chemistry, QML, and combinatorial optimization.

3) Error Correction: Due to the need for algorithms that are capable of handling quantum noise in the NISQ era, several techniques have been proposed for error correction and mitigation. In classical systems, error correction mechanisms are based on redundancy. If the copies do not retain the same value, a majority vote determines the correct value. This process works well in systems with a sufficiently low error probability, since it is most likely that only single errors appear.

[1]NISQ is a term coined by John Preskill [32].

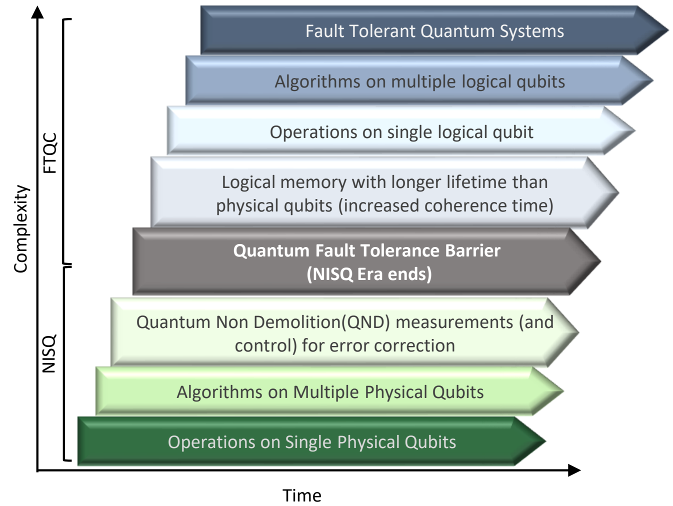

Fig. 9: Quantum Information Processing Development Stages highlighting NISQ and FTQC era phases.

Similarly, error correction techniques for quantum systems do not correct 100% of the errors but help to reduce the effect of the noise. Unlike classical bits that can only be affected by a flip between 0 and 1, qubits can also experience phase errors, i.e., errors appearing when a qubit state changes its phase. Moreover, quantum errors are continuous since the rotation angle can assume any value.

Due to the no-cloning theorem [52], creating a perfect copy of a quantum state is impossible. Hence, Quantum Error Correction (QEC) codes introduce redundancy by spreading the information of a single qubit onto an entangled state of multiple qubits. With this approach, it is possible to perform multi-qubit (syndrome) measurements to extract the information about the error without altering the quantum information. Typical QEC schemes are the three-qubit bit flip code [53], the three-qubit phase flip code [54] and Shor's nine qubit code [55].

As shown in Figure 10, it is currently not possible to engineer systems with noise rates lower than 10^{-2} or 10^{-3} per gate, but the quantum community is confident that this threshold can be overcome in the next 5-10 years.

4) Error Mitigation: Despite sharing the same goal as QEC to reduce the impact of quantum noise, Quantum Error Mitigation (QEM) techniques operate differently. While QEC aims to restore the correct value after the error occurs, QEM aims to reduce or suppress the errors that occur during computation. Moreover, as shown in Figure 11, QEC completely removes the noise, while QEM keeps a small amount of noise under control. A scenario where QEC is applied, called 'hard regime', is theoretically ideal but difficult to realize. The other scenario where QEM is employed, called 'easy regime', is experimentally easier to implement but at the price of lower qubit protection. For QEM deployment,

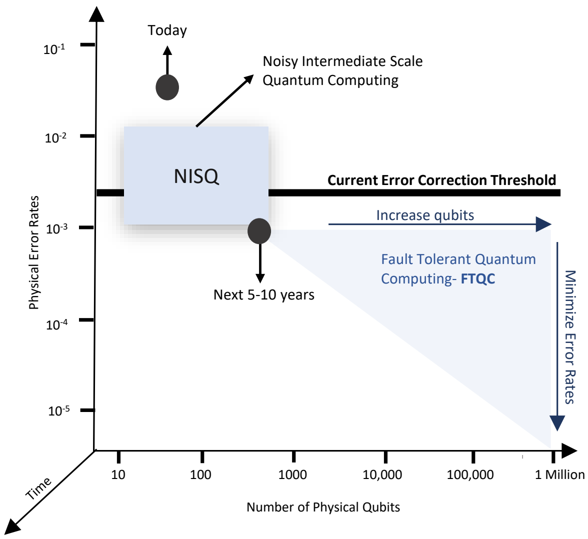

Fig. 10: Current state of QEC development. The figure shows the physical error rate threshold per gate that is currently not possible to overcome in the NISQ era.

Mitiq [56] is an open-source error mitigation package in Python that implements QEM techniques like zero-noise extrapolation [57] and probabilistic error cancellation [58] for quantum computers.

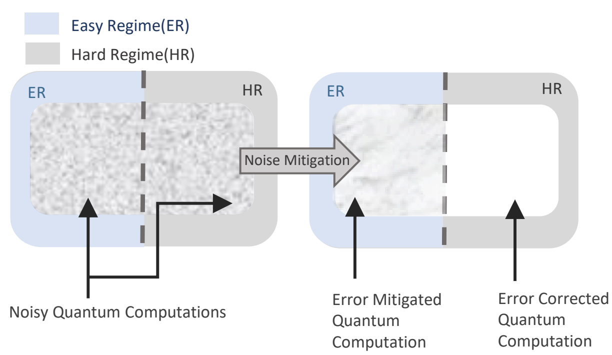

Fig. 11: Error correction vs. error mitigation.

The NISQ era is characterized by relatively small-sized quantum circuits. Moreover, quantum noise affects state preparation, gate operations, and measurement. Due to the many qubits and large circuit depth required by QEC codes, it is impossible to implement QEC on NISQ devices. On the other hand, QEM techniques offer low-overhead solutions to implement quantum circuits in an accurate and reliable way [50] [54].

5) Fault-Tolerant Quantum Computing: Fault tolerance is a property that enables a system to continue operating not only in normal conditions but also in the presence of hardware or software failures. While there is not often a direct comparison between classical and quantum information processing, the principles of fault tolerance in classical systems are easily transferable.

In a quantum computer, the basic gates are much more vulnerable to noise than classical transistors, since qubits' implementations depend on manipulating single electron spins, photon polarization, or similar fragile subatomic particle systems. Currently, the main challenge for developing efficient and reliable quantum hardware is the ability to maintain qubit states long enough to perform useful computations without requiring redundancy efforts that cause high compute inefficiency and overhead due to the duplication of states. However, it might not be possible to engineer systems with lower error rates. An important ongoing research direction dictates investigating alternative technologies for qubit materials to build quantum systems that achieve reasonable coherence times.

Additionally, the phenomenon of entanglement makes quantum systems inherently fragile. The interactions between qubits during the execution of the quantum circuit lead to errors. If several errors appear in an uncontrolled manner, the QEC or QEM is overwhelmed, and the computation will fail. Therefore, the main goal of FTQC is to control the propagation of faults between processes.

G. Quantum Stack

Quantum computers operate under a fundamentally different set of rules than classical computers and can solve certain problems faster by exploiting quantum effects. The development stack in Figure 12 shows an abstraction of the quantum stages formulating a generalized quantum stack. Starting from the Hardware or qubit level, the abstractions go up to the application level composed of Quantum Algorithms. Since QC is at an early development stage, industries primarily focus on the Quantum Hardware level, where the reliability of measurements plays a crucial role in designing and choosing the physical mediums. While simulated quantum hardware can deal with perfect qubits, realistic quantum hardware can be implemented with superconducting qubits, ion trapped qubits, quantum dots, and neutral atom qubits [59].

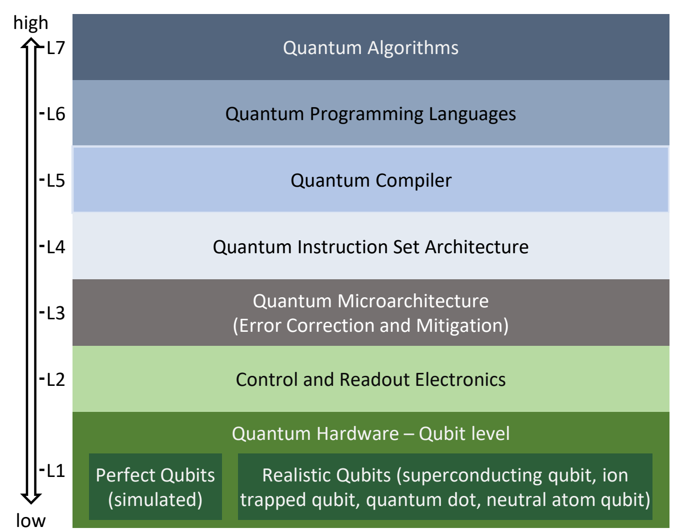

Fig. 12: Quantum stack - from Hardware to Applications.

III. Benefits of Quantum Computing

This section presents an overview of the benefits of QC over classical systems. It provides the necessary motivations to conduct further research and development of quantum computers.

A. Quantum Dimensionality Reduction

A common way to reveal properties of an unknown quantum state, having many copies of a system in that state, is to measure different observables and to analyze the results statistically. The unknown quantum state can play an active role in this analysis. Given multiple copies of a quantum system with its respective density matrix, it is possible to perform the unitary transformation. Its result generates quantum coherence among different copies of the system. Popular techniques for dimensionality reduction are the Principal Component Analysis (PCA) [60], where the eigenvectors corresponding to the unknown state's eigenvalues state are revealed in a very fast manner, and the Quantum Linear Discriminant Analysis Dimensionality Reduction (QLDADR) [61], where the vectors having a high-dimensional feature space are projected onto a low-dimensional feature space.

B. Quantum Supremacy for Solving Complex Problems

The inherent problems' difficulty, or hardness, is a fundamental concept in computational complexity theory (see Appendix A for more details). The hardness quantifies the resources required by different models of computation to solve problems, such as the number of steps of a deterministic Turing machine [62]. Various models and resources are often considered, including deterministic, nondeterministic, and probabilistic models, time and space constraints, and interactions across different models. "Quantum Supremacy" is a concept that indicates that a programmable quantum device can solve a problem that no classical computer can solve in a feasible amount of time. For instance, Google's Sycamore processor [63] performs in 200 seconds a task that would require $10,000$ years using a classical computer. Such an acclaimed advantage of QC compared to classical computers demands a modification of the complexity theory for problems that are solvable on a quantum computer. Hence, the classical complexity classification diagram has been redefined to incorporate the quantum perspective and evaluate the complexity of the issues, considering their ability to be solvable on a quantum computer. Figure 13 shows that a set of problems outside the P space that can be solved in Bounded-error, Quantum, and Polynomial-time (BQP) [64]. Identifying BQP problems is crucial to prove the overwhelming ability of quantum computers compared to classical computers.

C. Quantum Enhancement to Classical Problems

In the NISQ era, as important as it is to lead the direction to showcase QC's ability to outperform classical computers to solve problems with unattainable solutions within the classical paradigm. It is also necessary to pave a direction where we put both paradigms together to complement each other. It is a key step to comprehend how to use quantum and classical paradigms in an amalgamated approach to solutions that utilize the strengths of each framework in their own aspects and practicality. In contrast to the criteria set for achieving quantum advantage, the quantum enhancement's goal is to show QC solving a hard problem in a realistic time frame, irrespective of the reliability of the solution. It is also important to present a stringent version of the passing criteria that dictate and drive the practical usefulness of Quantum Computers. Frameworks and solutions designed under such a lens can be seen as quantum enhancements to existing but not limited to current classical systems and designs.

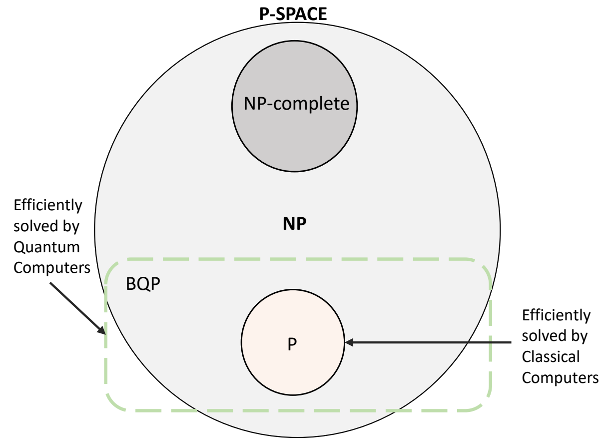

Fig. 13: Quantum Computing complexity theory. The classical complexity theory is extended with the class of Bounded-error, Quantum, and Polynomial-time (BQP) problems.

Demonstrating such aspects of QC is essential to gain confidence and value in using such systems. Hence, works like [65] that aim to practically demonstrate enhancements like quantum speedup for a computational problem are good examples of paths leading to practical quantum usability. Similar works investigate how the generic quantum advantage passing criteria can be tuned slightly towards strict reasoning to generate useful results and efforts.

To be critical of the current state of quantum practicality, existing methods are either heuristic-based or depend largely on the encoding of data in a quantum mechanical way that causes the change in performance and are the sole aspects making the difference from classical performance. In light of the hybrid quantum-classical systems, the work in [66] discusses important aspects that should be changed to obtain meaningful developments and revolves around how heuristic-based algorithms do not provide formal evidence that showcases their genuine and consistent advantage over classical algorithms. Moreover, the variational circuits can only implement linear classifiers on input quantum mechanically encoded to improve feature extraction, which can be replaced by classical support vector machines if the encoding is classically tractable.

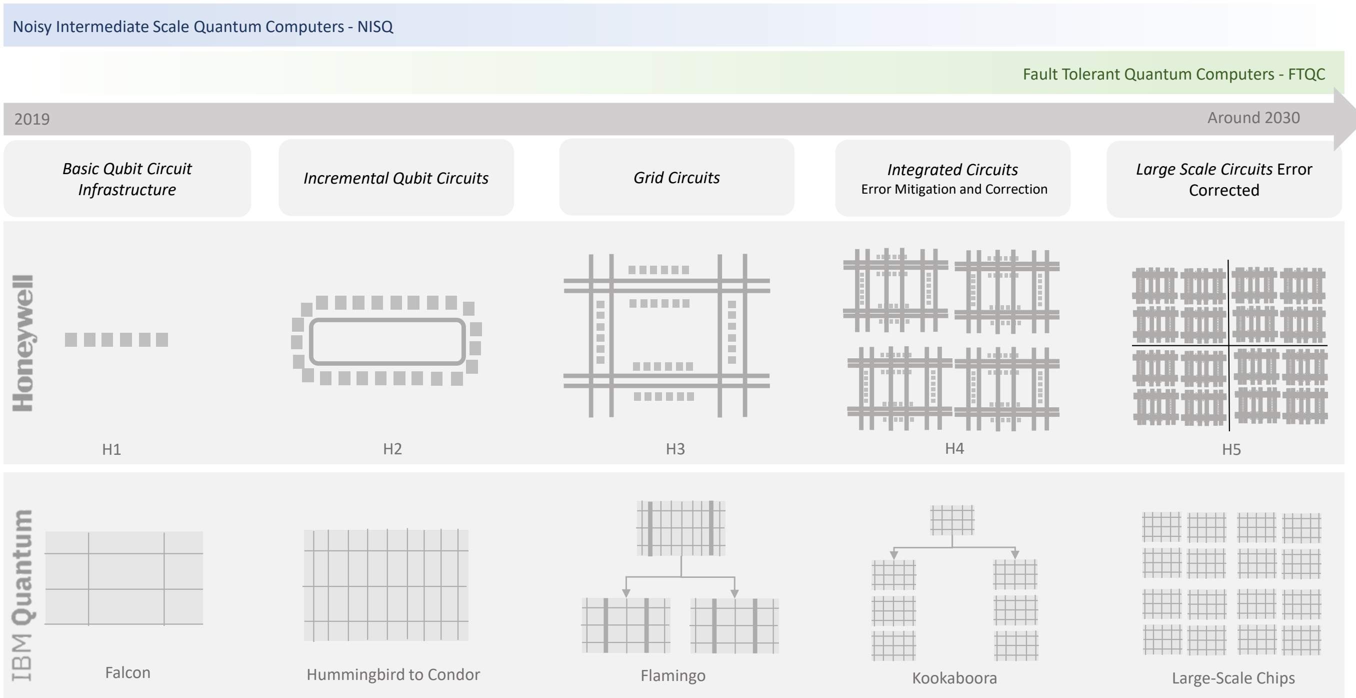

Fig. 14: Roadmap to design large-scale quantum chips by Honeywell and IBM. Starting from the basic qubit circuit infrastructure, the systems incrementally evolve to support more qubits. The development roadmap includes grid circuits and integrated circuits with error correction and mitigation, to ultimately design large-scale circuits that are envisioned to be released by 2030.

Such valid critical questions arise and should be addressed to fully comprehend and utilize the potential of quantum systems. Our survey aims to benefit researchers on these trails of thought as it is necessary to dive deeper into the field to come up with answers to such questions. Hence, our goal is to enable researchers using our survey to understand, comprehend, and compare the types of methodologies and tools they have to derive practical solutions and answers to their problems.

D. Preserving Maximum Information

Quantum computers are machines that execute calculations using the characteristics of quantum mechanics. What is occurring at the quantum scale follows the different laws of physics from what is known empirically from classical physics. Due to the superposition phenomenon, a qubit can represent the dual state of 0 and 1 simultaneously, while a classic binary bit can only take either 0 or 1 at a time. Hence, to express n-bit combinations using a classical computer, we need 2^n combinations, which can be checked sequentially before the classical computer can find the solution. On the other hand, a quantum computer takes advantage of its wave nature so that the wave interference increases the probability of the desired state and decreases the other to reach a correct solution effectively all at once.

E. Large-Scale Infrastructures and Design Automation

Infrastructure is an asset to any economy. Its development and maintenance, despite being difficult, have an enormous contribution to economic development and prosperous future growth. Continuous needs for humanitarian survival and improved quality of life are attracting necessary investments in transportation, civil engineering, and health sectors. Developing systems that provide smooth processing for these domains is challenging, given the burden of data required to process and handle with the aspects of time criticality, user safety, and reliability. The potential of QC to provide extensive grounds to optimize these processes defines new out-of-the-box ways for infrastructural development that are entirely new and creatively solve the problems at hand in a way that was impossible with classical computers.

The realization of large-scale quantum systems will take several years. However, it is the ultimate target for all significant quantum hardware companies. Their roadmaps aim to have such large-scale systems available for industrial and practical use around the end of the NISQ era. Figure 14 shows the development roadmaps provided by IBM [67] and Honeywell [68] for their current progress and plans towards achieving large-scale quantum infrastructures. These milestones demonstrate the continuous focus on developing quantum chips with more qubits every year, like the 432-qubit Osprey machine by IBM [69]. Similarly, Honeywell's H-series [70] has stages planned from H1 to H5 as their progress

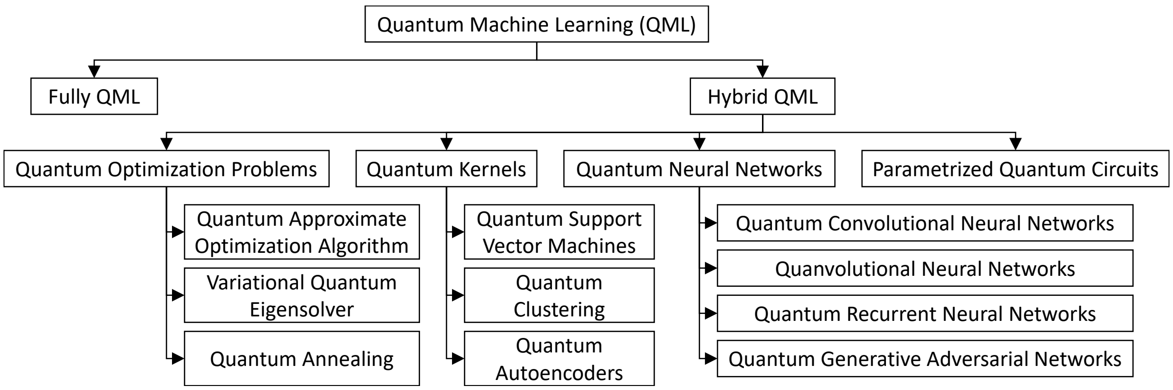

Fig. 15: Overview of QML algorithms.

from 10 qubit systems to larger scale qubit quantum systems envisioned to be designed around the year 2030.

IV. Quantum Machine Learning Algorithms

This section provides an overview of QML algorithms and their domain applicability. ML is a class of advanced algorithms that perform a certain task. Given a large number of inputs and desired outputs, an ML model can be trained to make predictions on unseen data. If it is executed on quantum computers, it becomes a quantum ML algorithm. An overview of the existing QML algorithms is shown in Figure 15.

A. *Categorization of QML Approaches*

Before diving into the details of QML algorithms, it is important to characterize different approaches based on the type of data and type of processor used to solve the problem [8]. The four categories (see Figure 16) are formed based on whether the data is classical (C) or quantum (Q) and whether the algorithm runs on a classical (C) or quantum (Q) computer.

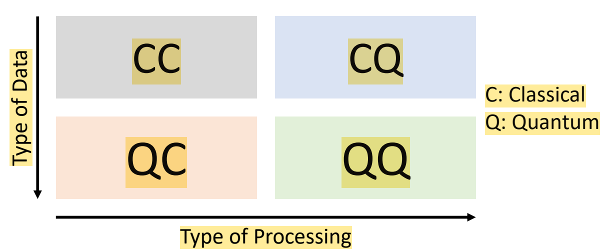

Fig. 16: Categories of QML based on types of data and processor.

- **CC** refers to processing *Classical data using Classical computers*, but using algorithms inspired by quantum computing, such as the recommendation system algorithm.
- **CQ** refers to processing *Classical data using Quantum machine learning algorithms* and will be the main focus of this chapter.
- **QC** refers to processing *Quantum data using Classical machine learning algorithms*. This is an active area of investigation, with classical machine learning algorithms used in many quantum computing areas, such as qubit characterization, control, and readout
- **QQ** refers to processing *Quantum data using Quantum machine learning algorithms*. It is also known as Fully Quantum Machine Learning (FQML). It can be considered a future investigation area that can be developed during a more mature stage of quantum computing.

B. *Parametrized Quantum Circuits*

Variational or Parametrized Quantum Circuits (PQCs) are specific types of quantum algorithms that depend on free parameters. PQCs allow us to utilize the existing quantum computers to their full extent. In the context of QML, PQCs are used either to encode the data, where the parameters are determined by the data being encoded, or as a quantum model, where the parameters are determined by an optimization process. PQCs can be interpreted as ML models, considering a variational quantum classifier that uses two variational circuits (see Figure 17). The first circuit associates the gate parameters with fixed data inputs, while the second circuit depends on free and trainable parameters. This setup provides the basic building blocks to build QML algorithms on NISQ devices [71].

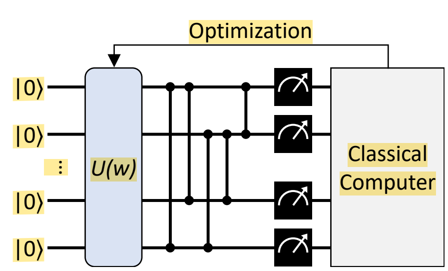

Fig. 17: Example of a parametrized quantum circuit.

A Variational Quantum Algorithm (VQA) combines a classical optimizer with a PQC. VQAs represent a promising approach to achieving quantum advantage over classical systems, even when using NISQ devices that come with various constraints. These constraints include a limited number of qubits, restricted qubit connectivity, and decoherence errors that limit the depth of quantum circuits [72]. Despite these challenges, VQAs can leverage the strengths of both classical and quantum computing to tackle problems that are intractable for classical algorithms alone. Examples of applications in which VCAs excel are quantum chemistry, material science, financial modeling, and cryptography.

C. Quantum Optimization Problems

An optimization problem aims to find the best solution to a given challenge. For instance, the goals of a business are to minimize the production cost or maximize its revenue. The solution can either be discrete if the variable to optimize is determined from a countable set or continuous if the optimized value is found from a continuous function. It can also be either single-objective or multi-objective. Applying QML to solve optimization problems enables us to employ quantum classification and regression programs on QML infrastructure (see Figure 18). The intrinsic parallelism of quantum computing can speed up the optimization computation to compute the global minimum (or maximum) faster than classical computing [73].

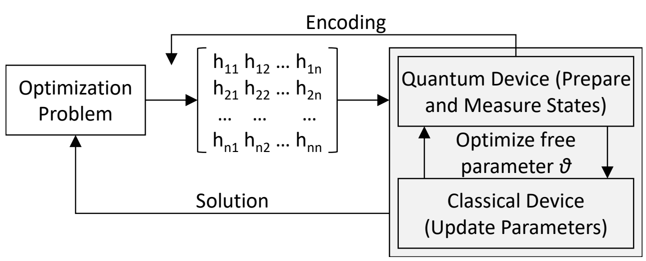

Fig. 18: Solving a combinatorial Optimization Problem with hybrid quantum models. Hamiltonians (H) are always square matrices with lengths equal to 2^n, where n is the number of qubits in the Hamiltonian.

The primary goal and advantage of all efforts behind quantum algorithms is to motivate and identify the practical advantage over classical in practical scenarios. Similarly, in gate-based quantum computers, which are currently adopted in the industry, and in shared future directions, it is important and useful to define beneficial use-cases of a quantum algorithm, to showcase its efficient utilization of the quantum hardware. Such efforts are now making their way into the research work involving techniques like Filtering Variational Quantum Eigensolvers [74]. Driven by the idea of filtering operators while processing to increase speed and reliability towards converging towards an optimal solution, such efforts are necessary to make QML systems impactful.

1) Variational Amplitude Amplification and Estimation:

Variational Amplitude Amplification

Amplitude amplification is a technique in quantum computing algorithms derived by generalizing the working principle of Grover's Search Algorithm [75]. Initially discovered by Brassad and Hoyer [76], the approach to finding a solution using "amplitude amplification" gives rise to a new family of quantum algorithms with the advantage of obtaining a quadratic speedup over several classical algorithms in comparison to a brute-force search. However, to observe its best potential impact when applied to real-world problems that embody larger search spaces for solutions, we need heuristics along with "amplitude amplification" to make it practically useful [77].

Variational Amplitude Estimation

Quantum Amplitude Estimation (QAE) is the task of finding an estimate for the amplitude of the solution state using the process of amplitude amplification described above. It can be generalized as a way of finding a solution to a combinatorial search problem, where components of interest in the space are projected in sub-spaces as a superposition of the required solution and other possibilities. Given the problem in this context, the process of "amplitude estimation" is the algorithm for estimating the probability that the final measurement after applying amplification will yield the desired state of the solution for the respective problem. The advantage of using this algorithm lies in the fact that it provides a quadratically faster solution than expected by a classical algorithm, provided the condition that the solution to the problem is known to be unique [75], [77], [78]. The first implementation of the algorithm by Brassad and Hoyer [76] for amplitude estimation employs a strategy involving phase estimation, which makes the algorithm compute-intensive. Another variant employs an iterative strategy to avoid phase estimation, which represents a useful evolution of the amplitude estimation strategy [79].

2) Quantum Approximate Optimization Algorithm: The Quantum Approximate Optimization Algorithm (QAOA) [80] is a technique that finds approximate solutions to combinatorial optimization problems. It is based on PQCs that approximate the adiabatic evolution from an initial Hamiltonian. A set of parameters are tweaked to optimize a cost function out of the quantum circuit output. The QAOA method is a promising candidate for achieving quantum advantage over classical systems in the NISQ era. Potential applications of the QAOA in the real world span from the logistics field, such as designing air/ground traffic and shipping routes, to the finance domain, where it can be used to maximize profits and minimize risks for a given portfolio.

3) Variational Quantum Eigensolver: A Variational Quantum Eigensolver (VQE) is a hybrid algorithm that uses both classical computers and quantum computers to find the ground state of a given physical system. The VQE algorithm is versatile and can be applied to a wide range of tasks, making it a promising candidate for use in many applications of quantum computers, including machine learning and control theory [81]. One particularly useful application of the VQE algorithm is finding the ground state energy (i.e., the minimum eigenvalue) of a Hamiltonian. This is accomplished by minimizing a cost function based on the expectation value of the Hamiltonian. An additional term in this algorithm accounts for the overlap between the excited and ground states.

An instance of VQE requires the definition of two algorithmic sub-components, which are a quantum trial state (also called *Ansatz* [82]) and a *classical optimizer*. The optimizer varies the parameters of the Anzatz, such that it works towards a state, as determined by its parameters, that results in the minimum expectation value being measured by the input operator (Hamiltonian). Additionally, the VQE can be used to model complex wave functions in polynomial time, making it one of the most promising NISQ applications for quantum computing. In practical applications, the VQE is used in chemistry for simulations of molecules, as well as logistics and network design.

4) Quantum Annealing: The Quantum Annealing (QA) method is an algorithm to solve combinatorial optimization problems. Instead of using the temperature to explore the problem space, the QA uses the laws of quantum mechanics to measure the energy state. Starting from the qubits' dual state, where all the possible combinations of the values are equally likely to happen, the quantum mechanical effects are gradually reduced through a process called quantum fluctuation [83] to reach the most optimal point where the lowest energy state is achieved. In practical use-cases, the QA is efficient for combinatorial optimization problems whose search space has several local minima, such as the travelling salesman problem.

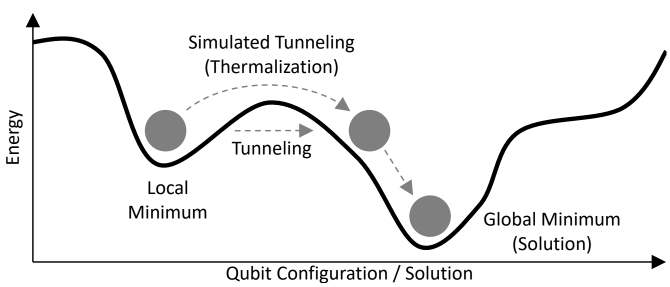

Fig. 19: Example of quantum annealing.

QA techniques require an objective function to be defined in the *Ising model* or *Quadratic Unconstrained Binary Optimisation (QUBO)*. The Ising model is a mathematical model in statistical mechanics [84]. The QUBO [85] is mathematically equivalent to the Ising model and can be used to formulate a problem in a more simple way.

In physics, the minimum energy principle [86] states that the internal energy decreases and approaches the minimum values. If we can formulate our problems as an energy minimization problem, the QA method can search for the best possible solution by utilizing quantum methods over an energy landscape.

D. Quantum Neural Networks

Quantum Neural Networks (QNNs) are computational Artificial Neural Network (ANN) models that are based on the principles of quantum mechanics [87]. As shown in Figure 20, typically in QNNs, the input data is loaded with classical data inputs. The quantum circuit contains a feature map module with input parameters and an Ansatz module with trainable weights. Measurements are conducted to obtain the outputs.

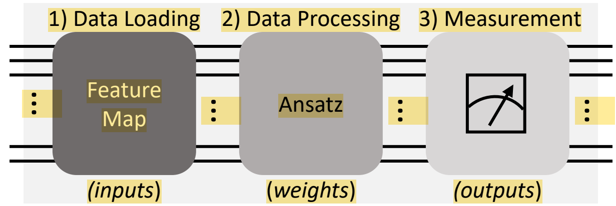

Fig. 20: Example of a Quantum Neural Network.

What makes QNNs exciting compared to traditional ANNs is the differentiable nature of the quantum circuits. Quantum computers can compute the changes in the control parameters needed to make the QNN better at a given task. While common applications of QNNs are in the area of image recognition and object detection since they excel in detecting specific features from images, QNNs can be used for a plethora of applications, as discussed in Section VII.

During the NISQ era, the main focus is on Hybrid Quantum Neural Networks (HQNNs). The generic structure of an HQNN is illustrated in Figure 21, while specialized model types are discussed in the following sections.

1) Quantum Convolutional Neural Networks: Convolutional Neural Networks(CNNs) are established architectures for image classification tasks in the computer vision domain [88]. Quantum Convolutional Neural Networks (QCNNs) are HQNN architectures whose structure is inspired by classical CNNs (see Figure 22). The QCNN circuit model proposed by Cong et al. [89] extends fundamental properties of CNNs to the quantum domain by using only $O(log(N))$ variational parameters for input sizes of N qubits. Therefore, it enables efficient training and implementation of QCNNs on realistic, near-term quantum devices.

The structure of a classical CNN consists of applying alternating convolutional layers (with an activation function) and pooling layers, typically followed by fully-connected layers before the output is generated. In QCNNs, the convolution operations are performed as parameterized unitary rotations based on PQCs, executed on neighboring pairs of qubits. These convolutions are followed by pooling layers. The scope of pooling layers is to down-sample the results of the previous layer by reducing the data dimensions while inherently extracting the most relevant features. The dimensionality reduction with pooling layers reduces the dimensions of the quantum circuits, which in turn reduces the number of qubits in the circuit while retaining the maximum information possible from predecessor layers. Consequently, it reduces the computational cost of the complete circuit since the number of learnable parameters of the QCNN is equal to the qubits in the circuit. However, in quantum physics, it is not possible to directly remove qubits from the circuit. Hence, the pooling layers in quantum circuits are deployed by measuring a subset of the qubits and using these measurement results to control subsequent operations. The fully-connected layers

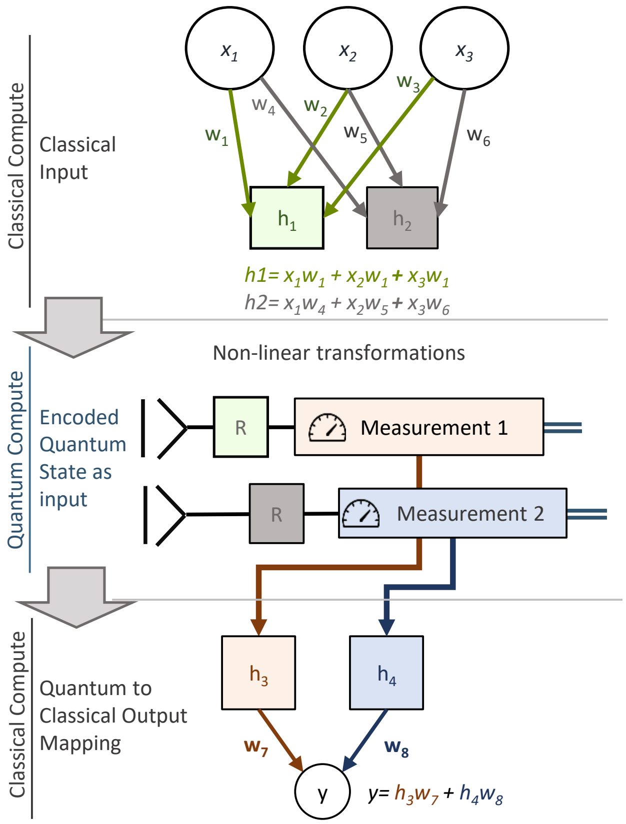

Fig. 21: Functionality of a Hybrid Quantum Deep Neural Network. The classical input is converted into the quantum domain by encoding the data as a quantum state. After measurements, a quantum to classical output mapping is required to obtain the final results.

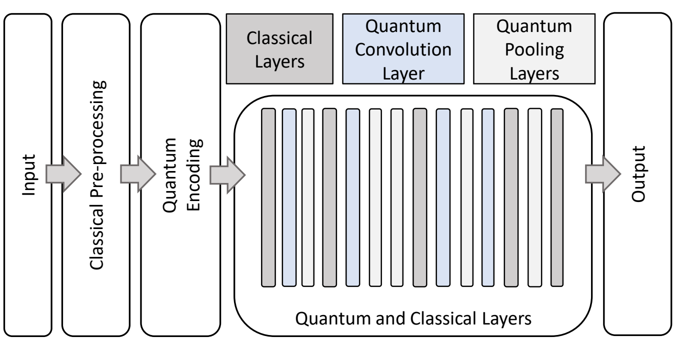

Fig. 22: Functionality of a Quantum Convolutional Neural Network.

are implemented as a multi-qubit operation on the remaining qubits before the final measurement. All the parameters involved in these operations are learned during training.

While the classification task is the primary purpose for using QCNNs, they can also devise a quantum error correction scheme optimized for a given error model [89]. However, QCNNs with large circuit depths are challenging to implement due to the high coherence required for qubits.

2) Quanvolutional Neural Networks: Motivated by the idea of a convolution layer where, instead of processing the entire input data with a global function, a local convolution is applied, Quanvolutional Neural Networks (QuanCNNs) [90] are based on a "quanvolutional" kernel. Compared to QCNNs, where the complete architecture is implemented with PQCs, QuanCNNs focus only on building efficient convolutional layers using quantum circuits (see Figure 23). Considering that the underlying basic principle of convolutional layers is to extract features from data hierarchically, a quanvolutional layer leverages the ability of quantum systems to contextualize greater information capacity. Therefore, QuanCNNs introduce preliminary layers before the classical CNN to manipulate (i.e., pre-process through transformations) the inputs using a single or a set of consecutive quantum layers. The input of the first classical layer is a quantum output measured classically that preserves a large amount of information for improving the contextual feature extraction in the classical layers. In other words, it aims to encapsulate the input information using a representation that results from the quantum space transformations. This approach complements and facilitates the architecture in identifying correlated features in the subsequent (classical) layers.

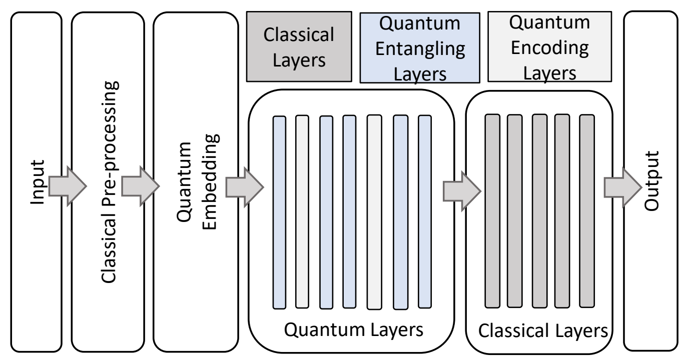

Fig. 23: Functionality of a Quanvolutional Neural Network.

Considering the current limits of the quantum hardware, where the quantum circuit implementation of many qubits is challenging, QuanCNNs represent a valid solution for circumventing this obstacle since the deeper layers are still implemented in classical computing. The advantage of quantum computing is that it is used for implementing convolutional layers, where low-qubit circuits are sufficient to perform the required task well.

3) Qauntum Recurrent Neural Networks: Recurrent Neural Networks (RNNs) are the foundational models for using sequential data to solve temporal problems such as natural language processing. Inspired by the structure of a VQE quantum circuit, a Quantum Recurrent Neural Network (QRNN) replaces the RNN layers with PQCs. While VQE circuits are very dense, QRNN cells are highly structured circuits with fewer parameters that are reused. Each parameter has a higher-level logical unit than the VQE components [91]. Applying multiple QRNN cells iteratively on the input sequence generates a QRNN (see Figure 24) that behaves similarly to classical RNNs. However, due to the decoherence issues, the QRNNs' performance on long sequential data is typically worse than classical RNNs. The coherence constraints can be relaxed if the QRNN architecture is built by stacking the recurrent blocks in a staggered way [92].

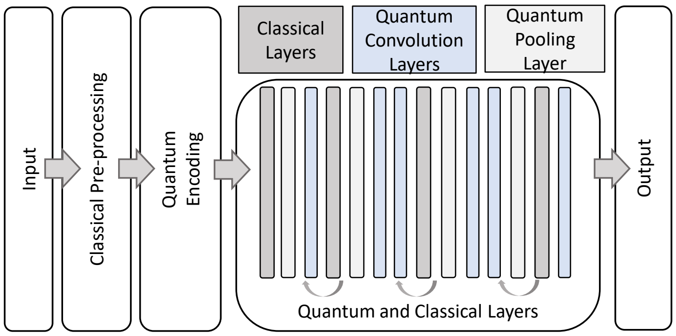

Fig. 24: Functionality of a Quantum Recurrent Neural Network.

4) Quantum Generative Adversarial Networks: A Generative Adversarial Network (GAN) aims to generate data that resembles the original samples used in the training dataset. This is typically implemented using a generator that produces new (fake) data and a discriminator that distinguishes fake from real data. Fully Quantum GANs (QGANs) typically require too large resources that cannot be implemented in near-term quantum devices. To overcome this limitation, it is possible to design hybrid QGANs where either the generator or the discriminator is implemented with classical computing. For example, the QGAN architecture proposed by Huang et al. [93] has a quantum generator and a classical discriminator. The generator is divided into multiple sub-generator blocks, each of them responsible for generating a patch of the full image. As shown in Figure 25, the generated image is formed by concatenating the patches. This approach can be implemented on quantum computers with a limited number of available qubits, where the same device can execute the sub-generator sequentially. This architecture provides guidance for developing advanced QGAN models on near-term quantum devices and opens up an avenue for exploring quantum advantages in various GAN-related tasks.

5) Quantum Autoencoders: Autoencoders (AEs) are self-supervised ML models that reduce the size of the input data by reconstructing it. It can compress and encode information from the input using representation learning. Quantum counterparts of AEs, called Quantum Autoencoders (QAEs) [94], aim to reduce the dimensionality of quantum states. As shown in Figure 26, a QAE is composed of three layers, which are input, bottleneck, and output. Its primary use is for digital compression, where the information can be encoded into a smaller amount of qubits. QAEs enable the transformation and mapping of large problems into smaller quantum circuit equivalents. Moreover, QAEs are useful for denoising since they extract relevant features from the initial quantum state into encoded data while neglecting the additional noise [50].

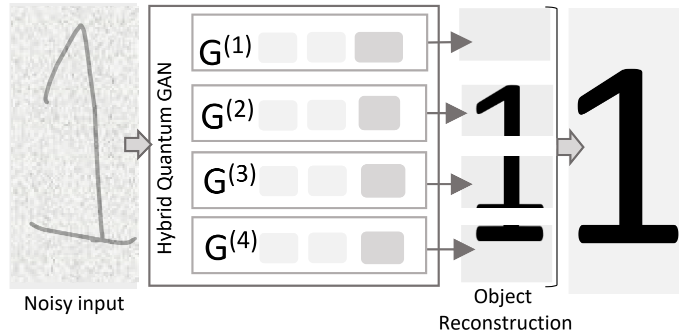

Fig. 25: Functionality of a Quantum Generative Adversarial Network, where the quantum generator is split into multiple sub-generators that craft patches of the full image.

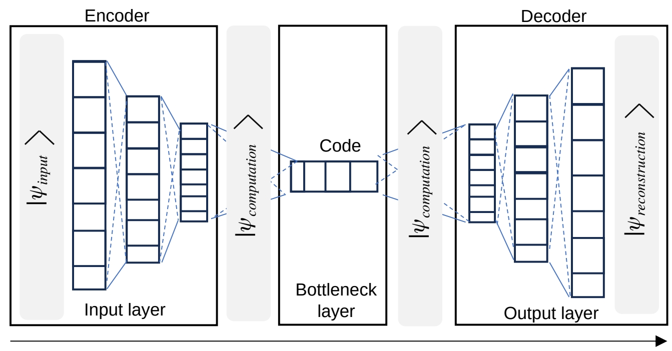

Fig. 26: Functionality of a Quantum Autoencoder, where the input state is compressed through the encoder and then decompressed through the decoder.

6) Binary Neural Networks with Quantum Optimization : Another class of neural networks, whose design is motivated by the need for a compact architecture for increased portability and ubiquitous computing, is the Binary Neural Networks (BNNs). As suggested by their name, the working principle is to replace floating point feature weights and activations with binary values, introducing a binarization process. The redesigned strategy is favorable as it reduces the required storage and computation for resource-constrained devices while leveraging the trade-off with its performance. This process allows compact, lightweight, and power-efficient networks that can simultaneously maintain acceptable accuracy. Methods for improving the performance of BNNs are an active area of research and quantum methods theoretically present themselves as a useful tool for improvement. The work presented in [95] argued theoretically that QCs could train BNNs to their global optimum via quantum search techniques in a shorter time than using brute-force classical optimization. An alternative approach to training BNNs utilizes a quantum superposition of weights [96], [97].

7) Single-Qubit Neural Networks : Single-qubit QNNs are based on the data re-uploading technique [98]. According to this method, each layer receives a copy of the input on the same single qubit line. The measured state of the solution stores the sequentially-encoded processing result carried on from all the sequential layers. The work in [99] presents a theoretical framework formulated to showcase the expressive capability of the employed data re-uploading technique over the quantum neural network layers, each of them comprising interconnected encoding circuits and trainable circuit blocks. It is further highlighted how single-qubit QNNs can approximate any univariate function by mapping the model to a partial Fourier series.While it seems to be a useful direction to be explored, as demonstrated in [100], there are limitations for evaluating multivariate functions by analyzing the frequency spectrum and the flexibility of Fourier coefficients which must be kept in mind while designing architectures based on the single qubit QNN template. [99], [101], [102]

E. Quantum Kernels

Kernel methods are a collection of pattern analysis algorithms that employ kernel functions to operate in a high-dimensional feature space, such as Support Vector Machines (SVMs) and clustering techniques. The main objective of SVMs is to find decision boundaries to group a given set of data points into classes. When these classes' data spaces are not linearly separable, SVMs can benefit from using kernels to find these boundaries.

Similarly to classical kernels, Quantum Kernels (QKs) help to deal with data points in high dimensional feature space and are typically employed to conduct unsupervised classification tasks. A quantum kernel can be defined as in Equation (3), where x and y are n-dimensional inputs and $Q(x, y)$ is the kernel function. f denotes a function that performs the mapping from an n-dimensional space into an m-dimensional space, where $n < m$. $\langle f(x), f(y) \rangle$ denotes the dot product between the functions. This function serves as the foundation for performing clustering and regression algorithms in the quantum domain.

$$Q(x, y) = \langle f(x), f(y) \rangle \tag{3}$$

Constructing QKs for SVMs has been under investigation in recent works. The performance of the resulting quantum SVMs for classification problems with a small number of training points exceeds that of optimized classical models with conventional kernels [103] [104]. Moreover, QKs allow us to interpret the learning process with the help of quantum information tools [105].

Quantum clustering is a class of data clustering techniques that use the mathematical and conceptual tools of quantum mechanics. It belongs to a family of density-based clustering algorithms, where clusters are defined by areas of higher-density data points. Its design involves a scale-space probability function viewed at the lowest eigenstate of a Schrödinger equation, followed by simple analytic operations to derive a potential function whose minima determine cluster centers (see Figure 27). The design proposed by Horn et al. [106] was applied to a two-dimensional system but is also scalable to higher dimensions. As demonstrated by Aïmeur et al. [107], quantum clustering for different algorithms (divisive clustering, k-medians, and construction of a neighborhood graph) can obtain significant speedup compared to classical clustering.

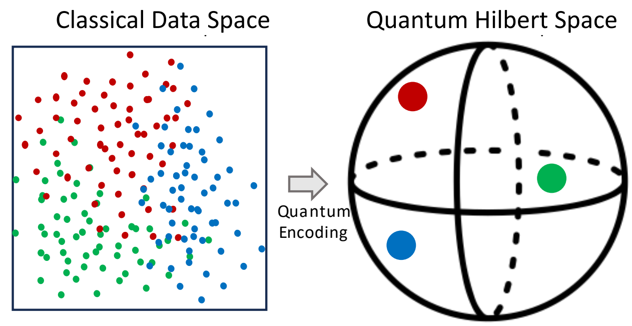

Fig. 27: Overview of Quantum Clustering, where the cluster centers are represented in the Hilbert space.

F. Quantum Computing for Bayesian Machine Learning

Bayesian ML is based on utilizing Bayesian methods, like Gaussian processes, to carry out ML tasks. Providing information on the uncertainty of predictions is one of its greatest strengths [108]. Looking into channeling this aspect of advantage to creatively encompass the uncertainty in a quantum methodology is a promising research direction. Interesting results have been obtained connecting deep feed-forward neural networks with Gaussian processes, which eliminates the need for back-propagation while training [108]. As back-propagation itself represents one of the challenges in QML, there is great potential for the success of methods such as Bayesian ML that can perform ML without back-propagation. This can be further optimized by utilizing quantum components in other useful paradigms within the framework. Specialized algorithms can find their way to implementation based on the Bayesian ML technique inspired by quantum optimizations. Such algorithms effectively exploit the connection between ML and Gaussian processes from the quantum lens to curate quantum-enhanced frameworks [108].

G. Limitations of Current QML Approaches

Despite the recent advancements in QML, there are still some limitations due to their relatively immature technology and relatively fewer studies compared to classical ML. The following list provides the most well-known limitations of QML.

- Most of the existing QML algorithms are not fully quantum but a hybrid combination of classical and quantum operations.
- The accuracy measured by QML algorithms is not yet matured to be on par with the state-of-the-art classical ML algorithms.
- The lack of quantum technology and tool standardization hinders the QML algorithm development due to their different design practices.

- The high-performance computation cost of developing large-scale QML incurs a high demand for access to tools and expensive resource infrastructures.
- The susceptibility to noise of quantum systems dictates instability and reliability concerns.
- For wide QNNs, the well-known Barren plateau problem [109] leads to trainability issues.
- Since the parameter update of QML algorithms relies on classical optimization modules, the quantum technology cannot be utilized to its full extent.
- The design strategies and usage of quantum gates in QML circuits are relatively naive and not well established.

V. Datasets

This section provides an overview of the existing methods for generating and manipulating quantum data to be usable for QML tasks. After discussing the data pre-processing methods and the data encoding techniques, we present a list of quantum datasets.

A. *Quantum Data*

In quantum mechanics, the fundamental principle is the evolution of energy systems over time, i.e., the Hamiltonian of the systems. In the domain of quantum computing, we deal with varying states of the system for computation and processing. Hence, quantum computing systems must be capable of handling data dictated by these transformations, mechanics, and quantum constraints. *Quantum data* can be referred to as relevant information describing the states and evolution of the quantum system over time. These quantum features represent input and output arguments for various quantum operations and functions that transform the system's quantum state from one form to another. The collection of such data is referred to as *quantum datasets*.

Some examples of quantum data can be:

- Factors affecting the relevant characteristics of a quantum system such as the Hamiltonian and other observables.
- A particular stationary state of the quantum system (e.g., the ground state), from where the process of evolution begins encapsulated in a quantum computation process.
- Unitary transformations, the possible quantum operations applicable to the system.
- Result of computation attained by measurement or additional projection operators which allow extraction of resulting distributions and expectation values of the system.
- Noise-relevant data for the systems' evolution and other phenomena collected via the control parameters.

The data collected for solving real-life problems are currently in classical form. To use quantum models for these solutions, we need to understand and process data in its quantum equivalent form. The possible types of classical data required to create quantum state representations for quantum processing are the following:

- Discrete data (binary or integer values).
- Real continuous data (floating-point values).
- Complex continuous data (complex values).

While fully quantum datasets will drive the FTQC era, the goal in the NISQ era is to understand how to encapsulate the current classical information into a quantum state. In this regard, after understanding the quantum pre-processing, we discuss encoding and embedding techniques for classical data into quantum data. Then, we provide a curated list of quantum datasets along with their properties and application domains.

B. *Motivation for QML Datasets*

A valuable lesson from the success of classical ML is that easily accessible and high-quality datasets catalyze the development of new algorithms and the improvement of existing ones. Learning from quantum data is more efficient when executed on quantum computers. Hence, deploying ML algorithms on quantum computers promises efficiency, accuracy, and rich data encapsulation into computing information about real-world systems.

Readily available data drives the efficiency of research quality. High-quality data curated for research and development catalyzes practical and collaborative research across disciplines and fosters insightful growth. Similarly, to experience the full potential of quantum computers to extract information and deeply understand a system through accurate simulations of its real-world counterpart, it is important to have quantum data available to diverse users.

C. *Data Pre-Processing*

Individual independent variables operating as inputs to the ML model are referred to as *features*. They can be thought of as representations or attributes that describe the data. An ML model makes accurate and precise predictions if the algorithm can easily interpret the data's features.

However, applying ML algorithms to noisy data would not give quality results as the system would fail to identify features effectively. Noisy data would introduce factors and patterns in the learned model that are different from the actual distribution of the given problem. Therefore, data pre-processing is important to improve the overall data quality to feed the ML models. The generic data pre-processing pipeline in an ML system includes the steps needed to transform or encode data so that it can be easily parsed by the machine, as shown in Figure 28a. The same principle applies in the quantum domain, but the quantum machines should also interpret the classical data. In this regard, a coherent and structured pre-processing mechanism for QML in the NISQ era is an inevitable need. Despite the generic pre-processing pipeline applicable in any case, Figure 28b illustrates a hybrid pre-processing pipeline for quantum model development. It ensures that the information extraction is maximized using quantum machines and allows quantum computation to apply to any kind of classical data.

Since QML has not yet extensively demonstrated its advantages compared to classical ML, it is more susceptible to adoption by the industry. The current quantum computers have few qubits to test and are noisy, making it difficult to

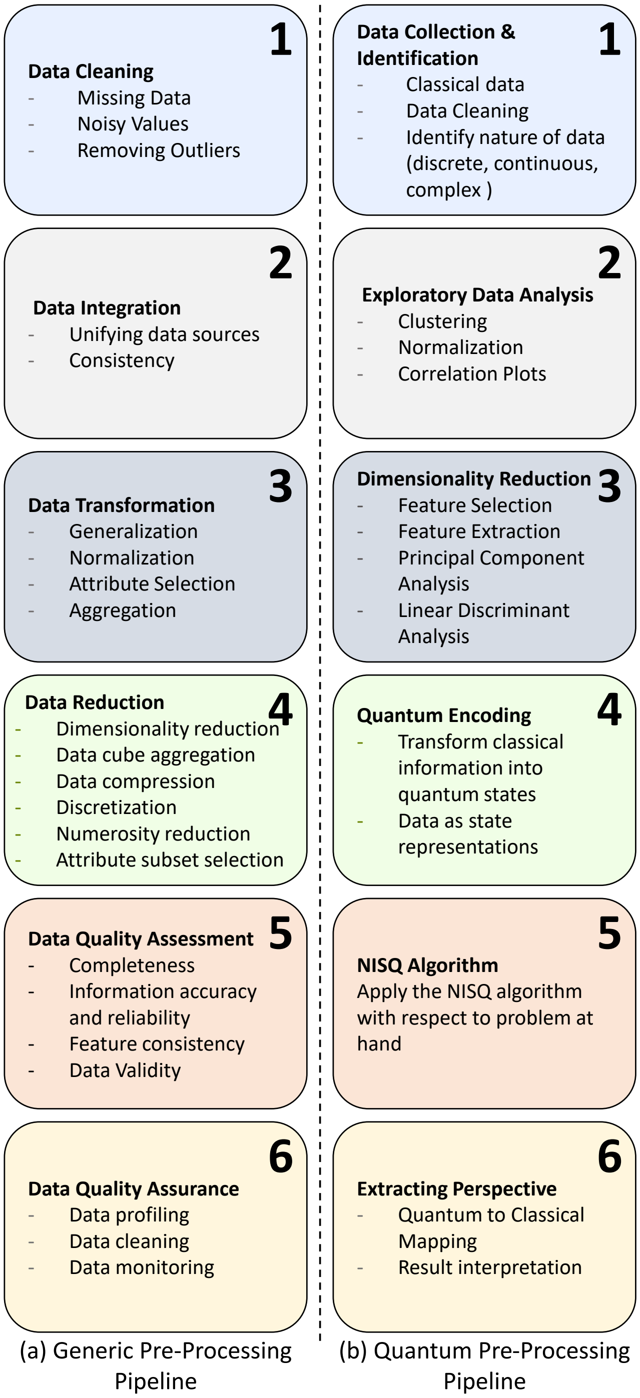

Fig. 28: Data pre-processing pipelines. **(a)** Generic pipeline. **(b)** Quantum pipeline.

demonstrate the current and potential quantum advantage of QML methods. Utilizing quantum methods for ML in the pre-processing phase, it is possible to achieve better classical encoding and performance of quantum classifiers [110]. Since many proposed QML applications rely on using well-known datasets, especially in the context of quantum data with constraints generated due to the underlying quantum mechanics for representing it as quantum states, it is important to formulate and standardize its collection and encoding processes over time to maintain quality [111].

D. Data Encoding

Even though the current quantum resources and methodologies provide close estimations and approximations of real-world systems, they do not always constitute adequate datasets for quantum classification models. The most common approach in the NISQ era is transforming the well-known classical datasets into the corresponding quantum representation and vice versa [111]. The fundamental step in the quantum processing pipeline is referred to as *state preparation*. It allows us to process data on quantum circuits before applying any specific quantum computation. Figure 27 shows an example of how quantum encoding is used to project data from the classical space to the quantum (Hilbert) space.

For tackling the data constraints in QML implementations, data encoding is a fundamental step for representing classical data as quantum states. Encoding layers largely influence the QML model expressivity since the data encoding strategy defines and drives the relevant QML parameters, e.g., the features the QML model can represent [25]. Knowing the various encoding techniques is necessary to choose the most suitable one for solving the particular QML task. Each encoding method has constraints due to the unstable quantum mechanical properties that hinder the complete data encapsulation in quantum representation. However, despite that, they successfully encode the information into quantum states and have large margins of improvement over time [112]–[114]. Figure 29 shows a generic overview of the quantum state preparation stage. The process of embedding/encoding generates a quantum representation of the classical data governed by the selected encoding technique within the Hilbert space dimension.

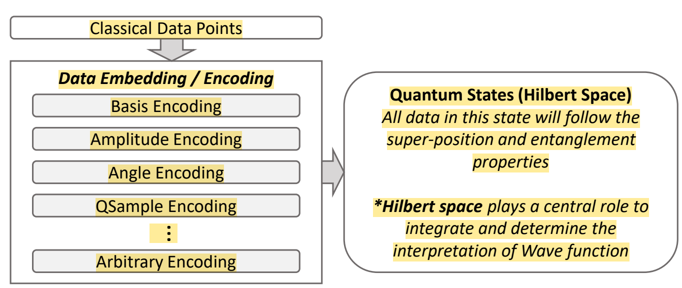

Fig. 29: Quantum state preparation model in the data processing pipeline.

Data encoding patterns describe a particular encoding as a tradeoff between three major objectives:

- The number of qubits needed for the encoding should be minimal because current quantum devices only support a limited number of qubits.
- The number of parallel operations needed to realize the encoding should be minimal to minimize the width of the

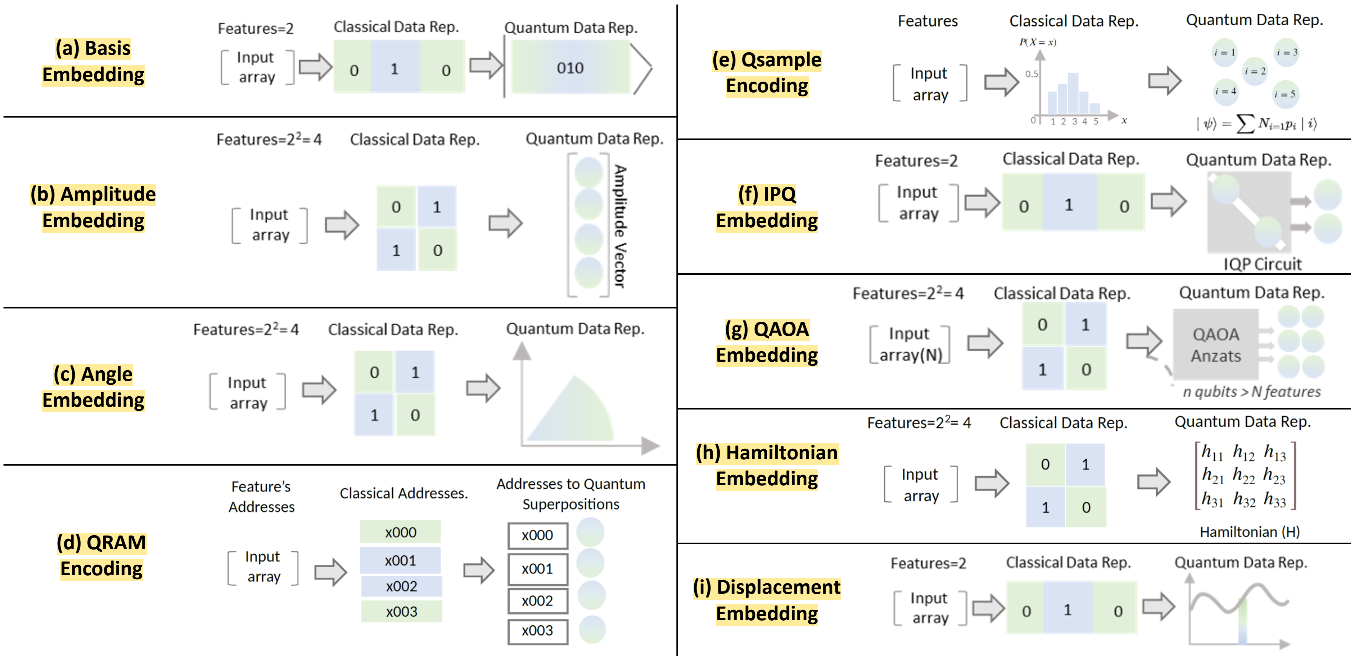

Fig. 30: Overview of quantum data encoding techniques. **(a)** Basis Embedding. **(b)** Amplitude Embedding. **(c)** Angle Embedding. **(d)** Quantum Random Access Memory (QRAM) Encoding. **(e)** Qsample Encoding. **(f)** Instantaneous Quantum Polynomial-time (IPQ) Embedding. **(g)** Quantum Approximate Optimization Algorithm (QAOA) Embedding. **(h)** Hamiltonian Embedding. **(i)** Displacement Embedding.

quantum circuit. Ideally, the loading routine should have constant or logarithmic complexity.

- The data must be represented appropriately for further calculations, e.g., arithmetic operations.

The following list contains the encoding techniques that are used in the literature. Note that the terms *encoding* and *embedding* can be used interchangeably as they refer to the same process. Figure 30 shows an overview of the data encoding techniques and their key characteristics.

(a) The **Basis Embedding** encodes the binary feature vector into a basis state. It is primarily used when real numbers are mathematically manipulated in quantum algorithms. Such an encoding represents real numbers as binary numbers and transforms them into quantum states. However, since the binary features are not differentiable, the basis encoding does not allow gradient computations [115].

(b) In the **Amplitude Embedding** technique, classical data is encoded into the amplitudes of a quantum state. Besides defining the measurement probabilities, each quantum system's wavefunction can be used to represent its amplitude as a data value. The main advantage of the amplitude embedding is that it can encode n real values (or n fixed-point precision approximation of real numbers) using $\mathcal{O}(\log n)$ qubits, i.e., it has a logarithmic runtime dependency on the dataset size. Due to its efficiency, it is employed in many QML algorithms [115].

(c) The **Angle Embedding** is performed by applying rotations on the x-axis or y-axis using specialized quantum gates, called Pauli rotation gates, along with the values that must be encoded. If we apply angle embedding on a dataset, the number of rotations is the same as the number of features in the dataset [115].

(d) The **Quantum Random Access Memory (QRAM) Encoding** [116] is a mechanism that allows accessing classically stored information in superposition by querying an index register. In other words, it enables direct access to the quantum data given the corresponding classical data value. By introducing a conditional rotation and branch selection procedure, we can feed the QRAM with the classical address of the required quantum data and get the qubit state at the output.

(e) The **Qsample Encoding** associates a real amplitude vector with a classical discrete probability distribution. It can be considered a hybrid case of Basis and Amplitude Encoding because amplitudes represent the encoded information, but the features are encoded in the qubits [115].

(f) The **Instantaneous Quantum Polynomial-time (IQP) Embedding** [117] employs the so-called IQP circuit, which is a quantum circuit of a block of Hadamard gates, followed by a set of gates that are diagonal in the computational basis. Each diagonal gate is composed of a single-qubit RZ rotation gate, encoding the n features, followed by a two-qubit ZZ entangler. The entangler, whose pattern can be customized, encodes the product between features.

(g) The **Quantum Approximate Optimization Algorithm**

TABLE II: Quantum datasets with key features.

Dataset	Ref.	Type	Volume/Count	Release Date
QDataSet	[118]	One-qubit and two-qubit systems evolving	14 TB (compressed), $10,000$ samples each	Aug.2021
QM7, QM8, QM9	[119], [120] [121], [122]	Organic molecules	Between $7,165$ and $133,885$ molecules, with 7-9 heavy atoms	2012-2015
PennyLane Datasets	[48]	Organic molecules and spin	42 geometries of 34 molecules, 4 spin models with 100 configurations	Apr.2022
TensorFlow Quantum Data	[123]	Image classification	One-qubit per pixel	Mar.2020
Mendley Datasets for Quantum Circuit Mapping	[124]	Circuit mapping problem as classification task	$188,434$ random quantum circuits	Sep.2021
QFlow Lite	[125]	Charge state recognition	$1,001$ - $1,599$ measurements, 100×100 - 250×250 pixel maps	Sep.2018
NTangled Datasets	[126]	Quantum states with multipartite entanglement	$12,040$ training samples, $4,060$ test samples	Nov.2021

(QAOA) Embedding [80] encodes n features into m qubits, where $m > n$. It uses a layered trainable quantum circuit inspired by the QAOA Ansatz. The feature-encoding circuit associates features with the angles of RX rotations. The main advantage is that it supports gradient computations with respect to both the features and the weights.

(h) The **Hamiltonian Embedding** scheme encodes features into the values of the Hamiltonian operator matrix [115]. The Hamiltonian operator of a system is a quantum mechanical operator that corresponds to the total energy of the system. Unlike previous methods that encode the data in the qubits' states, the Hamiltonian embedding method encodes the data into the operator. Since it is highly correlated with physics, various physical QML algorithms, including VQE and QAOA, are developed using the Hamiltonian encoding method.

(i) The **Displacement Embedding** encodes n features into the displacement of amplitudes or phases of m modes, where $n \leq m$. It is used in continuous-variable quantum computing models, where classical information is encoded in the displacement operator parameters [25].

E. Quantum Datasets Collection

Table II shows the collection of quantum datasets with key features. The following list provides a brief description of each dataset.

- The **QDataSet** [118], released in 2022, is composed of 52 datasets based on simulations of one-qubit and two-qubit systems evolving in the presence or absence of noise. It provides a large-scale set of datasets for QML practitioners to train, benchmark, and develop classical and quantum algorithms for common tasks in quantum sciences, such as quantum control, quantum tomography, and noise spectroscopy. It has been generated using customized code running on base-level Python packages to facilitate interoperability and portability across common QML platforms. Each dataset consists of $10,000$ samples and includes a range of information (stored in list, matrix, or tensor format) regarding quantum systems and their evolution, such as quantum state vectors, drift and control Hamiltonians, unitaries, Pauli measurement distributions, time series data, pulse sequence data for square and Gaussian pulses, noise and distortion data. The total compressed size of the QDataSet (compressed with Pickle and zip) is around 14 TB, while its uncompressed size is around 100 TB.
- The **QM7, QM8, and QM9** datasets are subsets of the GDB-13 [127] and GDB-17 [128] databases, which contain billions of organic molecules. The QM7 [119] is composed of $7,165$ molecules of up to 23 atoms (including 7 heavy atoms such as C, N, O, and S), represented using the Coulomb matrix. An extension of the QM7 dataset for multitask learning, called QM7B [120], has 13 additional properties (e.g., polarizability, HOMO and LUMO eigenvalues, excitation energies) and contains a total of $7,211$ molecules. The QMB8 [121] has a training set of $10,000$ molecules with up to 8 heavy atoms. The QM9 [122] contains $133,885$ molecules with up to 9 heavy atoms. These databases serve as benchmarks for hybrid QML models and systematic identification of structure-property relationships.
- The **PennyLane Datasets** [48] are composed of the *Quantum Chemistry (QChem) Datasets* for common molecular systems and the *Quantum Many-Body Physics (QMBP) Datasets*. The QChem datasets contain the electronic structure data for 42 different geometries of molecules such as linear hydrogen chains, metallic and non-metallic hydrides, and charged species. For each geometry of the molecule, it is possible to extract molecular data (i.e., information regarding the molecule), Hamiltonian data (i.e., Hamiltonian for the molecular system), tapering data (i.e., features based on Z2 symmetries of the molecular Hamiltonian for performing

tapering), tapered observables data, and variational data. The QMBP datasets contain spin models displaying quantum correlations. For each spin system, datasets are generated for 1-D lattices (linear chain) and 2-D lattices (rectangular grid). Each dataset contains values for 100 different configurations of tunable parameters such as the external magnetic field and coupling constants.

- The **TensorFlow Quantum Data** builds upon classical datasets such as MNIST [129] and Fashion-MNIST [130]. It implements the method proposed by Farhi et al. [123] to use binary encoding for representing each pixel with a qubit, with the state depending on the pixel's value. According to the experimental results of Huang et al. [131], this method achieves quantum speedup in the fault-tolerant regime compared to the classical system.
- The **Mendley Datasets for Quantum Circuits Mapping** [124] helps to solve the circuit mapping problem as a classification task. Each dataset contains features related to the calibration data of the physical device and others related to the generated quantum circuit for specific IBM quantum machines, such as IBMQ Santiago, IBMQ Athens, and IBMQ 16 Melbourne.
- The **QFlow Lite** dataset [125] contains a Python-based software suite to train neural networks to recognize the state of a device and differentiate between states in experimental data. It consists of $1,001$ idealized simulated measurements with gate configurations sampling (stored as 100×100 pixel maps) over different realizations of the same type of device. The QFlow Lite represents a reference dataset for researchers to use in their experiments for developing QML approaches and concepts. The expanded dataset, denominated QFlow 2.0, consists of $1,599$ idealized simulated measurements stored as 250×250 pixel maps. In addition, the QFlow 2.0 dataset includes two sets of noisy simulated measurements, one with a noise level varied around 1.5 times the optimized noise level, and the other with a noise level ranging from 0 to 7 times the optimized noise level.
- The **NTangled Datasets** [126] consist of trained weights for three hardware-efficient Ansatzes), strongly entangling, and convolutional, varying the number of qubits, depths, and types of multipartite entanglement. The states are generated by sending product state inputs into PQCs. The NTangled Datasets are used to benchmark QML models for supervised learning classification tasks.

VI. Tools and Technologies

This section introduces the tools, simulators, and hardware models that are available for QML practitioners. Since the infrastructure provided by QML industries is rapidly evolving, it is important to keep updated about the new technologies and software.

A. *Quantum Computing Models*

The first concept of a quantum computer was introduced in 1982 by Richard Feynman [31]. Since then, a significant amount of research efforts have been conducted that resulted in the development of real quantum computers at our disposal for experiments [132]. Due to their different use cases and architectures, quantum computers can be classified into quantum gate models (or quantum circuits) and quantum annealers.

1) Quantum Gate Model: The gate model involves operations in quantum computation using matrix operators, which are applied to the input qubits. These operators process the information and generate the final output distributions. They are designed with unitary constraints to qualify as valid quantum operators, making them reversible. The reversibility property is necessary for introducing some form of determinism in the quantum computation. A single operator is referred to as a gate in the quantum circuit and can be applied independently or grouped together with other similar quantum operators to perform computation on qubits. A gate can be combined with other gates in varying orientations and series such that they solve certain problems. The collection of gates arranged in configurations respective to the problem being solved along with qubits and measurements is referred to as a circuit (see Section II-D). Complex problems in the quantum domain can be solved by configuring gates in meaningful ways. The computation done with this quantum model is referred to as the gate model. Since it is easily interpretative and intuitive, it is nowadays the most used quantum computation model.

2) Quantum Annealer: Quantum Annealing (QA) is an adiabatic quantum computation method that finds optimal solutions to problems. It uses quantum tunneling, entanglement, and superposition to find the solution in terms of the system's energy. It usually corresponds to finding the global energy minimum in the quantum energy state plane. Instead of using temperature to explore the problem space, QA uses the laws of quantum mechanics to measure the energy state. A quantum annealer is relatively more noise-tolerant compared to the gate model. Quantum annealers shine when they are used to search for the best from many possible combinations. Such challenges are called combinatorial optimization problems (see Section IV-C4).

B. *Quantum Hardware*

Single qubits can be implemented using different materials and principles. As shown in Figure 31, companies do not use the same technology for implementing their quantum computers. For example, IBM Quantum uses niobium and aluminum on a silicon base to achieve superconductivity under near absolute zero temperature [69]. IonQ uses trapped ion quantum made of Ytterbium, a rare metal [133]. Another example is Xanadu, which applies photonics (light particles) in the quantum processing unit [134]. Using ion and light has advantages over the superconductivity materials as it works under room temperature. The superconducting qubits are more scalable and are implemented by other companies like Google [63] and D-Wave [135]. It is exciting to see future innovations in this area. In this section, after discussing the set of conditions necessary for designing qubits (denoted as DiVincenzo's criteria), we review the qubit technologies to understand their advantages and disadvantages and the reasons for their adoption in the industry. A detailed view of the quantum hardware technologies is also illustrated in Table III.

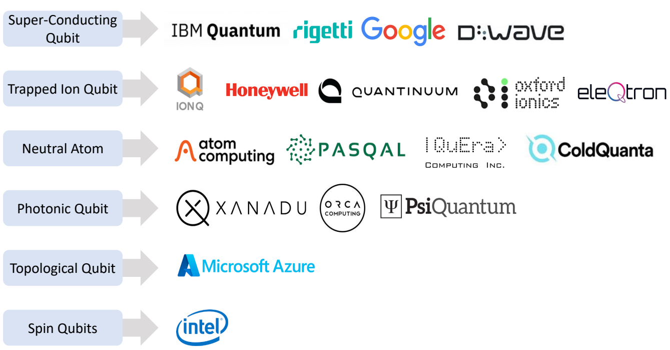

Fig. 31: Qubit technologies used by companies for their quantum computers.

1) DiVincenzo's Criteria: Introduced by theoretical physicist David P. DiVincenzo in the year 2000, DiVincenzo's criteria [147] defines a set of conditions necessary for constructing a quantum computer. The community widely accepts it, and, in the current state of the field, it represents a useful tool for designing qubits. The following list contains the fundamental characteristics for designing new qubit modalities in suitable experimental setup.

- Scalability with well-characterized qubits
- Ability to initialize qubits to a simple fiducial state
- Long relevant decoherence times
- A "universal" set of quantum gates
- A qubit-specific measurement capability

While the above criteria are necessary for designing qubits, the following two criteria are the minimum fundamental characteristics required for quantum communication.

- Ability to interconvert stationary and flying qubits
- Ability to faithfully transmit flying qubits between specified locations

2) Superconducting Qubits: Superconducting circuits are macroscopic in size but detain generic quantum properties such as entanglement, quantized energy levels, and superposition of states, all of which are more commonly associated with atoms. Their quantum state is manipulated using electromagnetic pulses to control the electric charge, the magnetic flux, or the phase difference across a Josephson junction [148] (a device with nonlinear inductance and no energy dissipation). As such, superconducting qubits represent the primitive building blocks of quantum computers [149]. There exist multiple types of superconducting qubits, such as flux qubits, phase qubits, charge qubits, and Transmon qubits. The most popular type is the Transmon qubit [150] given its design capability to limit noise effects, supporting error mitigation and correction.

3) Semiconducting Quantum Dots & Spin Qubits: Semiconductor charge and spin qubits based on gate-controlled semiconductor quantum dots, shallow dopants (i.e., traces of an impurity element introduced into a chemical material to alter its original electrical or optical properties), and color centers in wide-bandgap materials. Semiconductor Quantum Dots (QDs) [151] are nanoscale material clusters composed of 102–105 atoms. The size of the QDs is orders of magnitude larger than a typical atomic radius, yet small enough to provide quantum confinement of electrons and holes in all three spatial dimensions. Depending on their configuration, they can be controlled by both magnetic and electric fields. Therefore, they can be dephased by electric and magnetic field noise, with different types of spin qubits having various control mechanisms and noise susceptibilities.

In semiconductor quantum dots, spin qubits represent a prominent class of solid-state qubits in the effort to build a quantum computer. The simplest spin qubit is a single electron spin located in a quantum dot. However, many additional varieties have been designed, some containing multiple spins in multiple quantum dots, each with different benefits and drawbacks. Spin qubits based on solid-state defects have emerged as promising candidates because these qubits can be initialized, selectively controlled, and read with high fidelity at ambient temperatures.

4) Trapped Ions: As the name suggests, the qubits are ions trapped by electric fields and manipulated with lasers [54]. Trapped ions have relatively long coherence times, which implies that the qubits are long-lived. Moreover, they can easily interact with their neighbors.

5) Neutral Atoms: In the neutral atom quantum computing model, light beams manipulate arrays of single neutral atoms to encode and measure quantum states. In these quantum processors, a qubit is defined by one of two electronic states of an atom, and these single neutral atoms are arranged in configurable arrays. Neutral atom hyperfine qubits provide inherent scalability owing to their identical characteristics, long coherence times, and the ability to be trapped in dense, multidimensional arrays [152]. Combined with the strong entangling interactions provided by Rydberg states [153], all the necessary characteristics for quantum computation are met.

6) Photonic Atoms: One way to get scalable structures is to use photons (i.e., light particles) to represent qubits [154]. It consists of a ring that is used for photon storage, along with the scattering unit. Unlike other physical systems, photonics allows us access to an infinite number of states.

7) Transition Metal Dichalcogenides: Among the possibilities in the solid state, a defect in Transition Metal Dichalcogenides (TMDs), known as the nitrogen-vacancy (NV-1) center, stands out for its robustness. Its quantum state may be initialized, manipulated, and measured with high fidelity at room temperature. These controls can be used in conjunction with electronic structure theory to smartly sort through candidate defect systems for probable qubit

TABLE III: Quantum hardware technologies with key features.

Brand & Technology	Processor	Ref.	Connectivity	# Qubits	Single-Qubit Native Gates	Multi-Qubit Native Gates	Release Date
IBM Superconducting Qubit	Osprey	[69]	All-to-All	433	ID, RZ , SX, X	Echoed cross resonance gate (ECR)	Nov.2022
	Eagle	[136]		127			Nov.2021
	Egret			33		CZ	Apr.2022
	Hummingbird			65		CX	Nov.2020
	Falcon			27			Nov.2019
	Canary			5-16			Jan.2017
IonQ Trapped Ions	Forte	[137]	All-to-All	32	Gpi, GPi2, Virtual Z	Fully Entangling MS, Partially Entangling MS	2022
	Aria	[133]		21			2022
	Harmony	[133]		9			2020
Xanadu Photonic Qubits	Borealis	[134]	3D connectivity	216	NA	NA	Jun.2022
	X-Series	[138]	NA	NA			Mar.2021
Google Superconducting Qubits	Sycamore (Weber QC)	[63]	Square-Lattice Grid	54	Phased XZ, Virtual Z, Physical Z, Pauli Gates XYZ, PhasedXPowGate	Sycamore Gate, Square Root of iSWAP, CZ, FSim Gateset, Wait Gate, Parameterized Gates	Oct.2019
	Bristlecone	[139]	9-qubit linear array	72	NA	NA	Mar.2018
	Foxtail	[140]		NA			2016
Rigetti Superconducting Qubits	Aspen-M-3	[141]	Octagonal with 3-fold connectivity (2-fold for edges)	80	RX (angle, qubit) RZ (angle, qubit)	CPHASE, MEASURE (qubit, classical reg), CZ	Dec.2022
	Aspen-M-2			80			Aug.2022
	Aspen-M-1			80			Feb.2022
	Aspen-11			40			Dec.2021
	Aspen-10			32			Nov.2021
	Aspen-9			32			Feb.2021
	Aspen-8			31			May.2020
	Aspen-7			28			Nov.2019
	Aspen-4			13			Mar.2019
	Aspen-1			16			Nov.2018
	Acorn			19			Dec.2017
	Agave			8			Jun.2017
Intel Spin Qubits	Intel Quantum Hardware	[142]	NA	12	NA	NA	2023
Quantinuum (Honeywell) Trapped Ions	H2-1	[143]	fully connected qubits	32	Rotation Gates	ZZ	Aug.2023
	H1-3			20			Jun.2022
	H1-2			12			Jul.2021
	H-1			10			Sep.2020
Atom Computing Neutral Atoms	Phoenix	[144]	8:1 connectivity	100	NA	NA	Jul.2021
QuEra Neutral Atoms	Aquila	[145]	NA	256	Rabi Oscillations Time-dependent protocols Dynamic Decoupling Protocols	Adiabatic State Preparation and Rydberg Blockade Rabi Frequency Enhancement Levin-Pichler gate analogues Non-Equlibrium dynamics of 2 qubits	Jul.2021
D-Wave Systems Superconducting Qubits	Advantage	[135]	Degree-15 Lattice ($15 \times 15 \times 12$)	$5,000+$	superconducting quantum annealing	superconducting quantum annealing	Sep.2022
	2000Q	[146]	Degree-6 Lattice ($8 \times 8 \times 8$)	$2,000+$	Quantum Annealing	Quantum Annealing	Mar.2017

Legend **ID:** Identity. **RZ:** Rotation around Z-axis. **X:** NOT gate. **SX:** Square root of X. **CZ:** Controlled Z-gate. **CX:** Controlled X-gate. **Gpi:** π or bit-flip rotation with an embedded phase. **GPi2:** $RX\left(\frac{\pi}{2}\right)$ or $RY\left(\frac{\pi}{2}\right)$ with an embedded phase. **Fully Entangling MS:** XX gate, a simultaneous, entangling $\frac{\pi}{2}$ rotation on both qubits. **Partially Entangling MS:** a third (optional) arbitrary angle θ is added. **FSim Gateset:** provides all three Sycamore, iSWAP, and CZ gates in one set. In addition, by using this combined gate set, it can be parametrized, which allows for efficient sweeps across varying two-qubit gates. **CPHASE:** operation applied with a rotation parameter θ. **CZ (Rigetti):** offers single-pulse construction, allowing for lower error rates when full phase control is not needed.

representation [155]. However, this technology has not really been popular in the industry until now.

8) Topological Nanowire Qubits (Majorana Qubits): Majorana particles are a unique class of topological nanowire particles that are their own antiparticles. The quasi-particles emerge from the topological nature of systems and are bound at zero energy. They are predicted to obey non-abelian braiding statistics such that they can “remember” their histories. A *pair* of Majorana fermions represents a single qubit, and its realization could lead the way to fault-tolerant quantum computing [156]. Due to their stability at close to normal temperatures, Majorana Qubits work fine and are the technology chosen by Microsoft [157].

C. Quantum Circuit Mapping

The process of circuit mapping consists of compiling a quantum circuit to the given device. The circuit is transformed

TABLE IV: Quantum simulators with key features.

Brand	Simulator Name	Type	# Qubits	Noise Model
IBM	simulator_statevector	Schrödinger wavefunction	32	Yes
	simulator_stabilizer	Clifford	5000	Yes (Clifford only)
	simulator_extended_stabilizer	Extended Clifford	63	No
	simulator_mps	Matrix Product State	100	No
	ibmq_qasm_simulator	General, Context-Aware	32	Yes
	sampler	Qiskit Runtime Primitive (quasi-probability dist. generator)	NA (inherited from backend)	Yes (via *options* parameter)
	estimator	Qiskit Runtime Primitive (expectation values calculator)	NA	Yes (via *options* parameter)
Xanadu	default.qubit	Qubit Simulator	up to 26	Yes, non-native
	default.mixed	Qubit Simulator	up to 16	Yes (Native support)
	default.gaussian	Qubit Simulator	NA	Yes, non-native
	lightning.qubit	Quantum Photonic Simulator	up to 26	Yes, non-native
	lightning.gpu	GPU-Accelerated Simulator	up to 29 (+3 with MPI optimization)	Yes, non-native
	lightning.kokkos	Plugin for Kokkos Library	up to 29 (+3 with MPI optimization)	Yes, non-native
	Jet	tensor network contractions (acceleration)	NA	Yes, non-native
Rigetti	Quantum Virtual Machine (QVM)	Emulated Execution	NA	Yes (built-in and custom)
Intel	Intel Quantum Simulator (IQS)	Generic Qubit Simulator	32, 40	NA
Quantinuum (Honeywell)	H2 Emulator	Functional Simulator	32	Yes (via Quantinuum API)
	H1 Emulator	Functional Simulator	20	Yes (via Quantinuum API)
Google	Qsim	Full Wave Function Simulator	40	Yes (via *channel* in Criq)

to comply with the architecture's limited qubit connectivity.

Combining one-qubit rotation gates and two-qubit Controlled-NOT (CNOT) gates can realize any quantum circuit. However, CNOT gates cannot be applied to all pairs of qubits in the currently available NISQ quantum devices. Therefore, the circuit must be transformed into an equivalent circuit that does not violate the limitation of the device. The transformation often requires additional gates that limit the usability of the device [158]. The quality of the circuit mapping can be assessed via different metrics, such as the resulting circuit's (two-qubit) gate count, depth, or the expected fidelity [159].

Circuit mapping is a fundamental step in the context of QML. The factors affecting the development of QML algorithms directly depend on the design and optimization of circuit mapping. For instance, considering a circuit mapping capability of coupling up to 4 qubits significantly limits the depth and size of QNN layers.

D. Quantum Simulators and Primitives

Quantum Simulators are cloud-based classical systems emulating quantum systems. These are useful in the NISQ era as designers fully control the hardware characteristics that enable experiments of fault-tolerant quantum systems. Ideal quantum systems can be simulated for long-term design and development with no noise in the system. Moreover, noise models can be introduced and analyzed to comprehend the intuitive and counterintuitive behaviors of the quantum systems under noise.

Quantum Primitives are foundational building blocks for designing and optimizing quantum workloads. They provide options to customize the iteration and execution of programs to maximize the solution quality. A primitive is a set of key language elements that serve as the foundation for a programming language. Every language supports a core set of primitives that grant the basic building blocks for instructing a processor on executing specific operations.

To be broadly applicable, quantum computers must accurately control and process the information stored in thousands of quantum bits. However, as the number of qubits increases, noise and cross-talk tend to increase the error rate rapidly. In this thrust, it is important to study how the error rates scale with the increased qubit number and system connectivity. To mitigate this, quantum computers can be built in a modular fashion, with adequate infrastructure needed to control them efficiently, and the software compilation of quantum algorithms should be tailored to specific system architectures and error characteristics.

Several software simulators have been released by major companies involved in QML research. While Table IV summarizes the key features of the simulators, the following list provides key details of these tools.

- **IBM Qiskit** [47] is an open-source software development kit for quantum systems at the level of circuits, pulses, and algorithms. With the help of domain-specific APIs, Qiskit builds a software stack that makes it easy for users to use quantum computers, as it allows practitioners to easily design experiments and applications and run them on classical simulators or real quantum computers. An overview of the main characteristics of Qiskit is illustrated in Figure 32. It comprises the following components:
  - *Qiskit Runtime*: It contains computational primitives to perform foundational quantum computing tasks and supports error correction and mitigation techniques.
  - *Qiskit Composer*: A graphical quantum programming tool that lets users describe the operations to build quantum circuits and run them on real quantum hardware or simulators.
  - *Qiskit Nature*: It supports different applications and provides the necessary components to convert classical codes into representations required by quantum computers.
  - *Qiskit Finance*: It provides a set of illustrative applications and tools, including data providers for real or random data, Ising translators for portfolio optimization, and implementations for pricing different financial options or for credit risk analysis.
  - *Qiskit Machine Learning*: It provides fundamental QK and QNN building blocks for QML algorithms that apply them to solve different tasks such as regression and classification. Moreover, it connects to PyTorch to enhance classical ML workflows with quantum components.
  - *Qiskit Optimization*: It provides a range of optimizations, including high-level modeling of optimization problems, automatic conversion of problems to different required representations, and a suite of easy-to-use quantum optimization algorithms.
  - *Qiskit Experiments*: It runs characterization, calibration, and verification steps for the experiments.
  - *OpenQASM*: It is a simple text-format quantum assembly language for describing acyclic quantum circuits composed of single-qubit, controlled single-qubit, multiple-qubit, and controlled multiple-qubit gates. It is used to implement experiments with low-depth quantum circuits. OpenQASM represents universal physical circuits with straight-line code that includes measurement, reset, fast feedback, and gate subroutines. The simple text language can be written by hand or by higher-level tools and may be executed on the IBM Q Experience.
- **Xanadu PennyLane** [48] is the leading tool for programming quantum computers. It is a cross-platform Python library that enables quantum differentiable programming and integration with ML tools to train a quantum computer like you would train a DNN. PennyLane also supports a comprehensive set of features, hardware, simulators, and community-led resources that enable users of all levels to build, optimize, and deploy quantum-classical applications easily. An overview of its main features is depicted in Figure 33.
- **Xanadu Strawberry Fields** [160] is a full-stack Python library for designing quantum algorithms and executing them directly on photonic quantum computers such as Xanadu's next-generation quantum hardware.
- **Rigetti PyQuil** [161], [162] is a Python library for writing and running quantum programs using Quantum instruction language (Quil) programs. PyQuil allows developers to easily generate Quil programs from quantum gates and classical operations, compile and simulate these programs using the Quil compiler (Quilc) and the Quantum Virtual Machine (QVM), or execute them on a Rigetti quantum processor. Quilc is an optimizing compiler for gate-based quantum programs, capable of performing circuit transformations to produce optimal circuit implementations for a specific Rigetti processor. Such optimizations allow developers to write programs faster while preserving (or improving) their execution fidelity on a given hardware system. Quil is built and executed using the Forest SDK, a software tool that allows users to write quantum programs in Quil.
- **Quantinuum (Honeywell) TKET** [163] is an advanced software development kit, accessible through the PyTKET Python package, for the creation and execution of programs for gate-based quantum computers. It is platform-inclusive, and its state-of-the-art circuit optimization routines allow users to extract as much power as possible from any of today's NISQ devices.
- **Google Cirq** [164] is an open-source framework for programming quantum computers, whose software library can be used for writing, manipulating, optimizing, and running quantum circuits on quantum computers and quantum simulators. Cirq provides valuable abstractions for dealing with today's NISQ computers, where hardware details are vital to achieving state-of-the-art results.
- **Google OpenFermion** [165] is an open-source library for analyzing and compiling quantum algorithms to simulate fermionic systems, including quantum chemistry. The package provides efficient data structures for representing fermionic operators and fermionic circuit primitives for their execution on quantum devices. Plugins to OpenFermion provide users with efficient and low-overhead means of translating electronic structure calculations into quantum circuit calculations.
- **Google TensorFlow Quantum (TFQ)** [166] is a QML library for rapid prototyping hybrid quantum-classical ML models. Research in quantum algorithms and applications can leverage Google's TFQ, all from within TensorFlow. It integrates quantum computing algorithms and logic designed in Cirq and provides

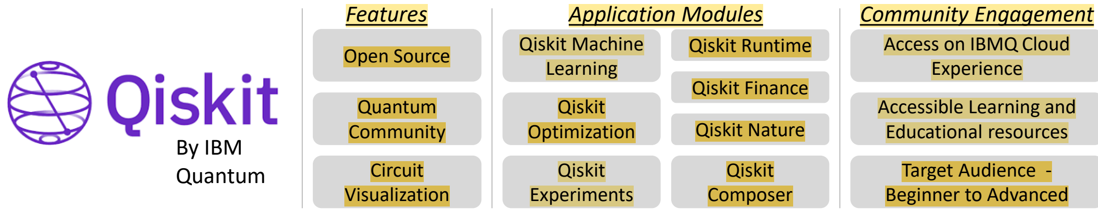

Fig. 32: Qiskit features and modules.

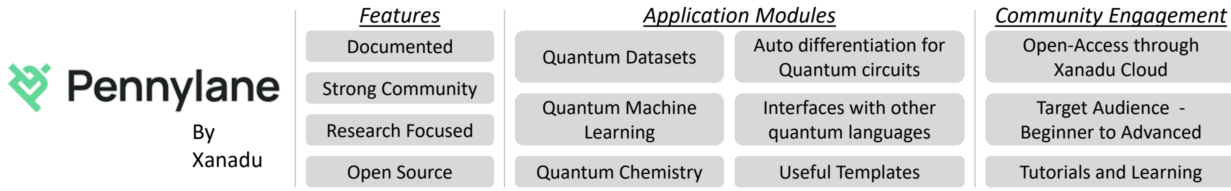

Fig. 33: PennyLane features and modules.

quantum computing primitives compatible with existing TensorFlow APIs, along with high-performance quantum circuit simulators.

- **D-Wave Leap** [167] is a programming model and execution framework to execute quantum and hybrid workloads effectively.

E. Supporting Infrastructure Tools and Hardware

To help users access quantum resources, several cloud-based platforms and development support infrastructures have been built. The following list describes the most popular resources used in the community.

- **qBraid** [168] is a cloud-based quantum computing platform designed to accelerate the pace of progress in the field. It offers access to the ecosystem of quantum software, hardware, and an entire suite of application-specific quantum algorithms, helping software engineers run programs on quantum computers available from AWS, IBM, Rigetti, IonQ, and QuEra within minutes. qBraid Lab is a web-based IDE interface (through JupiterLab) that provides software tools for developers and access to quantum hardware. It also includes CLI, a Command Line Interface for interacting with all parts of the qBraid platform.
- **IonQ Quantum Cloud** [169] provides access compatible with most Quantum SDKs with their continuous effort to extend the support to newer tools. The cloud provides access to all IonQ's quantum processors and supports noisy and ideal simulators.
- **Amazon Braket by AWS** [170] is a fully managed quantum computing service designed to speed up scientific research and software development for quantum computing. Its support is useful for experimenting with quantum computing algorithms, testing different quantum hardware, and building and developing quantum software. It provides access to different quantum computers and simulators using a consistent collection of development tools. Users can build quantum projects on a trusted cloud with affordable pricing and management controls for both quantum and classical workloads. It allows faster execution of hybrid quantum-classical algorithms with priority access to quantum computers and no classical infrastructure to manage.
- **Microsoft Azure Quantum** [171] provides quantum cloud access equipped with open-source development toolkits and environments for most leading quantum technologies with runtime metrics. Microsoft is adopting topological qubits as their qubit modality for developing their quantum computers. Their aim behind using these qubits directly corresponds with the long-term scalability of quantum computers. Since the topological qubits can survive at regular room temperatures, their usage improves the systems' stability compared to other qubit technologies. The resource estimator tool provided by Azure monitors the quantum system's runtime statistics. It estimates the resources allocated by the quantum code and refines the solution to a reasonable extent. This tool calculates the number of logical and physical qubits and runtime requirements to execute quantum applications on future-scaled quantum computers. Additional features, like determining the number of physical qubits required by circuits and evaluating the differences across qubit technologies, facilitate the design of quantum solutions with varying technologies and refining them to scaled versions for FTQC.
- **Zapata Orquestra** [172] is a platform to develop and deploy generative AI applications. It allows rapid prototyping, benchmarking, and executing workflows across quantum and classical computing resources. Their objectives focus on prioritizing use cases that can deliver a near-term financial impact and build customer first-mover capabilities, ultimately leading to solving computationally complex problems.
- **Xanadu Cloud** [173] is an interface that allows the

simulation and execution of programs on quantum photonic hardware. Users can develop quantum applications and program quantum computers with PennyLane's library of tools, demonstrations, tutorials, and community support forums.

VII. Applications

QC has been pacing itself from theory to practicality over the past decades, with immense efforts and investment attracting its attention, now more than ever. Although, there are significant hurdles in achieving high compute efficiency, a path is still paved in the direction of robust, reliable, and useful quantum computers. There is great potential in solving QML problems in many application areas that are discussed in this section.

In the following domains, the introduction of quantum technology introduces improvements in speed, compatibility, accuracy, or memory that make them suitable to be solved using quantum technologies.

A. Molecular Simulation: Drug Design

Identifying and developing small molecules and macromolecules that may help cure illnesses and diseases is the primary activity of pharmaceutical companies. Given its focus on molecular formations, the pharma industry is a natural candidate for deploying quantum computing. The molecules (also including those that might be used for drugs) are actually quantum systems, i.e., systems based on quantum physics. Quantum systems can predict and simulate these molecules' structure, properties, and behavior (or reactivity) more effectively than conventional computing, making QML an extensive branch of research in the respective industry. Exact methods are computationally intractable by standard computers, and often approximate methods are not sufficiently accurate for simulating such critical interactions on the sub-atomic level, as is the case for many compounds. As quantum computers become more powerful, significant discoveries will be made.

QC's primary value for pharmaceuticals lies in the R&D phase of drug design [174]. QML can help drug developers build predictive models to identify which molecules will likely be effective drug candidates. These models can filter out less promising candidates early in the drug development process, saving resources and time [175]. A practical use-case of this application is represented by material modeling [176], which can be efficiently implemented with QML.

B. Quantum Sensing

Quantum Sensing is an advanced sensor technology that detects changes in motion and electromagnetic fields by collecting data at the atomic level [177].

GPS, radar, lidar, and other electromagnetic technologies use quantum physics to provide increasingly common tools on city streets. In aircraft and even on basic cell phones, quantum sensing is in the process of shifting from being a highly-prized capability few can afford to being in daily use everywhere. Once it achieves widespread adoption, quantum sensing is expected to improve capabilities for the following domains dramatically:

- Aircraft, Automobile and Electronics Manufacturers
- Geology and Civil Engineering
- Environmental Management
- Weather Forecasting
- Cosmic wave detection and simulations (gravitational wave sensing)

As witnessed by the increasing number of startups in this field, quantum sensors are finding their way from laboratories to the real world. The atomic length scale of quantum sensors and their coherence properties enable unprecedented spatial resolution and sensitivity. Biomedical applications ranging from brain imaging to single-cell spectroscopy [178] could benefit from these quantum technologies, but evaluating the potential impact of the techniques is not trivial.

Several projects in the field of civil engineering have launched the adoption of quantum sensing technologies. Since quantum sensors allow more precise and accurate measurements, they relieve and facilitate in making the right decisions for critical tasks (e.g., underground detection of sinkholes, which currently are manually done by experts using their personal knowledge with minimum technological support. This causes a massive amount of resources and time wasted due to the lack of accuracy in correctly identifying the areas where to dig holes.

Metrology is another applicable domain for quantum sensing [179]. The current interests in this direction demand compact and susceptible measurement systems. Integrating electronics with quantum sensing technology should be one of the viable areas of interest for NISQ devices.

C. Quantum Cryptography

This application domain entirely focuses on creating new and improved cryptography schemes that ensure security against attacks and threats by quantum breach technology. Proactively taking measures against the utilization of quantum technology for breaking such schemes is referred to as Quantum Safe cryptography. Keeping that in mind, currently used encryption methods for many critical communication and transaction processes use the Rivest–Shamir–Adleman (RSA) scheme [180]. It is fundamentally based on the keys generated from prime factors of very large numbers. Although any classical computer can find prime numbers of a given small number given the understanding of complexity theory, it has been proven that for large numbers, the complexity of finding its prime numbers becomes difficult to intractable, i.e., solvable only in indefinite time. This particular property of the problem with classical computers made it an obvious choice to use the RSA as an encryption method. Something that would require thousands of years to solve with the available technology would raise no harm. However, the works of Peter Shor [181], [182] showed the capability of quantum computers to obtain solutions to problems that were presumably be indefinitely solved in the classical domain. This single instance of challenges to established security claims opens new avenues, driven by the need to develop improved and robust cryptography methods that make the communication *Quantum Safe* [183], [184]. Instead of completely changing a classical algorithm into quantum, the focus is on adding quantum modules to existing classical cryptography techniques. The advantage of this strategy is that while the existing classical techniques remain intact, applying the quantum methods improves their efficiency and speed [185]. This proved to be a very useful domain of potential applications for QC that builds over the existing schemes and makes them robust and efficient rather than re-inventing the whole process.

D. Financial Modeling

By enabling more accurate and complex financial data modeling, improving risk management, and optimizing investment portfolios, QML could lead to better and informed decision-making with higher returns for investors. It can also enable financial services organizations to re-engineer operational processes [186]. QML's specific use cases for financial services can be classified into three main categories: targeted predictions, trading optimizations, and risk profiling. Quantum technology computes more efficiently than classical computing when generating probability distributions, mapping data, testing samples, and iterating. These properties align well with the financial modeling since they can help enhance investment gains, reduce capital requirements, open new investment opportunities, and improve the identification and management of risk and compliance. The simplest instance of practical implementation of QML for finance can include a task of portfolio optimization using numerous quantum optimization algorithms, fundamentally derived from the parameterized quantum circuits. If not at the current stage, ultimately such tasks will benefit in terms of compute speed and processing information capacity for solutions using QML techniques.

E. Logistics Optimization

Supply chain and logistics professionals have been overloaded over the past several years. An increasing amount of uncertainties, extreme weather conditions, and pandemic-fueled supply and demand changes have exponentially increased logistics complexity. While businesses need supply chain optimizations to account for the complexity of the entire ecosystem, existing optimizations no longer provide the cure to supply chain and logistics challenges. In this regard, QML could provide solutions for persistent supply-chain and logistics problems, such as air traffic routing, trade shipment routing, telecom networks, quantum internet, and healthcare. Organizations in the transportation and logistics area could be among those to gain the most advantage from quantum computing's groundbreaking capabilities. An organization's path from the initial investigation of QML's impact to actual readiness for implementation can span a few years.

F. Environmental, Social, and Governance (ESG)

With the current environmental situation, we see the global need to establish and ensure sustainability. Despite that, technological developments are no further uncorrelated to the environment. Hence, while the quantum technology itself should be designed to be environmentally sustainable, it can be a catalyst in enabling sustainable energy solutions at a large infrastructure scale. Quantum technology, apart from reducing the efficiency and execution time, can also introduce and develop methods that complement Environmental, Social, and Governance (ESG) goals. It can drastically reduce energy wastage and fuel costs with its potential in route optimizations and sensing technologies. Moreover, it provides solutions that empower people to leave aside tedious and repetitive tasks and has a huge social impact by fulfilling problems that drastically improve the quality of life. Therefore, there is great potential to evolve the quantum field from the ESG sustainability perspective.

VIII. Conclusion and Road Ahead

The development of the QC field, particularly in QML, opens new avenues for demonstrating the true potential of technological advancements in various subfields. Despite the rapid and substantial progress, QML faces significant challenges related to the size and noise constraints, which affect the *scalability* and *reliability* of these systems. The path towards FTQC must be pursued diligently, following the steps outlined the roadmap for QC development. To achieve groundbreaking advancements in this relatively new and exciting field, it is essential to adopt innovative approaches rather than strictly adhering to traditional research and development steps. For QML implementations, finding analogies between classical ML and QML is not always necessary. Instead, exploring alternative perspectives to redefine metrics and standards from a quantum viewpoint is crucial. The open research directions can be summarized as follows.

- **Benchmarks:** Establishing benchmark suites for QML is essential for creating a standardized framework for evaluating and comparing different QML algorithms and infrastructures. The current lack of comprehensive comparative frameworks and standardized hybrid QML implementations hampers progress. In classical ML, benchmarks and state-of-the-art algorithms are well-established, providing clear metrics for performance evaluation. Although QML algorithms are still in their early stages and may not yet match the accuracy of mature classical ML algorithms, quantum computing offers unique opportunities to redefine comparison metrics and develop new benchmarks that capture the strengths of QML.
- **Scalability:** Achieving scalability in quantum computing involves moving beyond single quantum chips to develop multi-chip devices with efficient interconnection and communication systems. Companies like IBM and

Honeywell are making strides toward large-scale quantum devices, as shown in Figure 14, but significant challenges remain. The primary focus should be on minimizing error rates and enhancing system reliability. As suggested in [32], reducing error rates is a top priority for making quantum computation technology both impactful and widely adopted. Innovations in error correction and fault-tolerant quantum computing will be crucial for scaling up quantum systems.
- **Stable Technology Standardization:** To suppress noise and enable computations in noisy environments, it is vital to establish protocols that effectively characterize noise correlations between qubits in a scalable and cost-effective manner. The advanced technology required for quantum computing can be expensive, necessitating a focus on developing fault-tolerant systems and cost-efficient error-reduction methods. Balancing cost and reliability is key to selecting suitable technologies. Efforts should be directed toward designing robust error correction techniques that maintain performance while keeping costs manageable.
- **NISQ Limitations:** In the NISQ era, where quantum devices have limited qubits and high noise levels, it is crucial to design optimizations tailored to these constraints. Realistic expectations for QML technology deliverables are necessary, recognizing that while quantum computers have the potential to outperform classical systems, current hardware limitations must be accounted for. Developing QML algorithms and optimization strategies that operate efficiently within the context of these limitations is essential. This includes leveraging hybrid quantum-classical approaches, where classical preprocessing and postprocessing can complement quantum computations.

This survey provides a comprehensive overview of the current state-of-the-art algorithms, datasets, tools, and applications of QML. It serves as a reference resource to instruct and motivate passionate readers to learn and contribute to the research and design of innovative methodologies, tools, algorithms, and technologies that facilitate advancements in this emerging field. By exploring these directions, we can push the boundaries of what is possible with QML and pave the way for its broader adoption and impact.

ACKNOWLEDGMENTS

This work was supported in part by the NYUAD Center for Quantum and Topological Systems (CQTS), funded by Tamkeen under the NYUAD Research Institute grant CG008.

REFERENCES

[1] A. Marchisio, M. A. Hanif, M. Martina, and M. Shafique, "Prunet: Class-blind pruning method for deep neural networks," in *2018 International Joint Conference on Neural Networks, IJCNN 2018, Rio de Janeiro, Brazil, July 8-13, 2018*. IEEE, 2018, pp. 1–8. [Online]. Available: https://doi.org/10.1109/IJCNN.2018.8489764

[2] M. A. Hanif, A. Marchisio, T. Arif, R. Hafiz, S. Rehman, and M. Shafique, "X-dnns: Systematic cross-layer approximations for energy-efficient deep neural networks," *J. Low Power Electron.*, vol. 14, no. 4, pp. 520–534, 2018. [Online]. Available: https://doi.org/10.1166/jolpe.2018.1575

[3] A. Marchisio, M. A. Hanif, F. Khalid, G. Plastiras, C. Kyrkou, T. Theocharides, and M. Shafique, "Deep learning for edge computing: Current trends, cross-layer optimizations, and open research challenges," in *2019 IEEE Computer Society Annual Symposium on VLSI, ISVLSI 2019, Miami, FL, USA, July 15-17, 2019*. IEEE, 2019, pp. 553–559. [Online]. Available: https://doi.org/10.1109/ISVLSI.2019.00105

[4] M. A. Hanif, G. M. Sarda, A. Marchisio, G. Masera, M. Martina, and M. Shafique, "Conlocnn: Exploiting correlation and non-uniform quantization for energy-efficient low-precision deep convolutional neural networks," in *International Joint Conference on Neural Networks, IJCNN 2022, Padua, Italy, July 18-23, 2022*. IEEE, 2022, pp. 1–8. [Online]. Available: https://doi.org/10.1109/IJCNN55064.2022.9892902

[5] V. Leon, M. A. Hanif, G. Armeniakos, X. Jiao, M. Shafique, K. Z. Pekmestzi, and D. Soudris, "Approximate computing survey, part I: terminology and software & hardware approximation techniques," *CoRR*, vol. abs/2307.11124, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2307.11124

[6] J. L. Gustafson, "Moore's law," in *Encyclopedia of Parallel Computing*, D. A. Padua, Ed. Springer, 2011, pp. 1177–1184. [Online]. Available: https://doi.org/10.1007/978-0-387-09766-4_81

[7] OpenAI, "GPT-4 technical report," *CoRR*, vol. abs/2303.08774, 2023. [Online]. Available: https://doi.org/10.48550/arXiv.2303.08774

[8] E. Aïmeur, G. Brassard, and S. Gambs, "Machine learning in a quantum world," in *Advances in Artificial Intelligence, 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006, Québec City, Québec, Canada, June 7-9, 2006, Proceedings*, ser. Lecture Notes in Computer Science, L. Lamontagne and M. Marchand, Eds., vol. 4013. Springer, 2006, pp. 431–442. [Online]. Available: https://doi.org/10.1007/11766247_37

[9] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," *Contemporary Physics*, vol. 56, no. 2, pp. 172–185, oct 2014. [Online]. Available: https://doi.org/10.1080%2F00107514.2014.964942

[10] J. Adcock, E. Allen, M. Day, S. Frick, J. Hinchliff, M. Johnson, S. Morley-Short, S. Pallister, A. Price, and S. Stanisic, "Advances in quantum machine learning," *CoRR*, vol. abs/1512.02900, 2015.

[11] A. Montanaro, "Quantum algorithms: an overview," *npj Quantum Information*, vol. 2, no. 1, jan 2016. [Online]. Available: https://doi.org/10.1038%2Fnpjqi.2015.23

[12] V. Dunjko and H. J. Briegel, "Machine learning \& artificial intelligence in the quantum domain," *CoRR*, vol. abs/1709.02779, 2017. [Online]. Available: http://arxiv.org/abs/1709.02779

[13] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, "Quantum machine learning: a classical perspective," *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 474, no. 2209, p. 20170551, jan 2018. [Online]. Available: https://doi.org/10.1098%2Frspa.2017.0551

[14] S. Jeswal and S. Chakraverty, "Recent developments and applications in quantum neural network: A review," *Archives of Computational Methods in Engineering*, vol. 26, 05 2018.

[15] S. Resch and U. R. Karpuzcu, "Quantum computing: An overview across the system stack," *CoRR*, vol. abs/1905.07240, 2019.

[16] S. B. Ramezani, A. Sommers, H. K. Manchukonda, S. Rahimi, and A. Amirlatifi, "Machine learning algorithms in quantum computing: A survey," in *2020 International Joint Conference on Neural Networks, IJCNN 2020, Glasgow, United Kingdom, July 19-24, 2020*. IEEE, 2020, pp. 1–8. [Online]. Available: https://doi.org/10.1109/IJCNN48605.2020.9207714

[17] S. Chakraborty, T. Das, S. Sutradhar, M. Das, and S. Deb, "An analytical review of quantum neural network models and relevant research," in *2020 5th International Conference on Communication and Electronics Systems (ICCES)*, 2020, pp. 1395–1400.

[18] K. Bharti, T. Haug, V. Vedral, and L.-C. Kwek, "Machine learning meets quantum foundations: A brief survey," *AVS Quantum Science*, vol. 2, no. 3, jul 2020. [Online]. Available: https://doi.org/10.1116%2F5.0007529

[19] S. Mangini, F. Tacchino, D. Gerace, D. Bajoni, and C. Macchiavello, "Quantum computing models for artificial neural networks," *Europhysics Letters*, vol. 134, no. 1, p. 10002, apr 2021. [Online]. Available: https://doi.org/10.1209%2F0295-5075%2F134%2F10002

[20] Y. Kwak, W. J. Yun, S. Jung, and J. Kim, "Quantum neural networks: Concepts, applications, and challenges," in *Twelfth International Conference on Ubiquitous and Future Networks, ICUFN 2021, Jeju Island, South Korea, August 17-20, 2021*. IEEE, 2021, pp. 413–416. [Online]. Available: https://doi.org/10.1109/ICUFN49451.2021.9528698

[21] R. Zhao and S. Wang, "A review of quantum neural networks: Methods, models, dilemma," *CoRR*, vol. abs/2109.01840, 2021. [Online]. Available: https://arxiv.org/abs/2109.01840

[22] M. T. N, A. Hiremath, N. M, S.-L. Peng, S. M. R, and P. S. K, "A survey on machine learning techniques using quantum computing," in *2022 Fourth International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT)*, 2022, pp. 1–6.

[23] M. J. Umer and M. I. Sharif, "A comprehensive survey on quantum machine learning and possible applications," *Int. J. E Health Medical Commun.*, vol. 13, no. 5, pp. 1–17, 2022. [Online]. Available: https://doi.org/10.4018/ijehmc.315730

[24] N. Schetakis, D. Aghamalyan, P. Griffin, and M. Boguslavsky, "Review of some existing qml frameworks and novel hybrid classical–quantum neural networks realising binary classification for the noisy datasets," *Scientific Reports*, vol. 12, p. 11927, 07 2022.

[25] S. Markidis, "Programming quantum neural networks on NISQ systems: An overview of technologies and methodologies," *Entropy*, vol. 25, no. 4, p. 694, 2023. [Online]. Available: https://doi.org/10.3390/e25040694

[26] F. V. Massoli, L. Vadicamo, G. Amato, and F. Falchi, "A leap among quantum computing and quantum neural networks: A survey," *ACM Comput. Surv.*, vol. 55, no. 5, pp. 98:1–98:37, 2023. [Online]. Available: https://doi.org/10.1145/3529756

[27] A. M. Dalzell, S. McArdle, M. Berta, P. Bienias, C.-F. Chen, A. Gilyén, C. T. Hann, M. J. Kastoryano, E. T. Khabiboulline, A. Kubica, G. Salton, S. Wang, and F. G. S. L. Brandão, "Quantum algorithms: A survey of applications and end-to-end complexities," *CoRR*, vol. abs/2310.03011, 2023.

[28] L. Huynh, J. Hong, A. Mian, H. Suzuki, Y. Wu, and S. Camtepe, "Quantum-inspired machine learning: a survey," *CoRR*, vol. abs/2308.11269, 2023.

[29] K. A. Tychola, T. Kalampokas, and G. A. Papakostas, "Quantum machine learning—an overview," *Electronics*, vol. 12, no. 11, 2023. [Online]. Available: https://www.mdpi.com/2079-9292/12/11/2379

[30] E. N. Evans, D. Byrne, and M. G. Cook, "A quick introduction to quantum machine learning for non-practitioners," *CoRR*, vol. abs/2402.14694, 2024. [Online]. Available: https://doi.org/10.48550/arXiv.2402.14694

[31] R. Feynmann, "Simulating physics with computers," *Int J Theor Phys*, vol. 21, p. 467–488, 1982.

[32] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, aug 2018. [Online]. Available: https://doi.org/10.22331%2Fq-2018-08-06-79

[33] J. Preskill, "Quantum computing 40 years later," *CoRR*, vol. abs/2106.10522, 2023.

[34] M. Schuld and N. Killoran, "Is quantum advantage the right goal for quantum machine learning?" *PRX Quantum*, vol. 3, p. 030101, Jul 2022. [Online]. Available: https://link.aps.org/doi/10.1103/PRXQuantum.3.030101

[35] D. C. Marinescu and G. M. Marinescu, "Chapter 2 - measurements and quantum information," in *Classical and Quantum Information*, D. C. Marinescu and G. M. Marinescu, Eds. Boston: Academic Press, 2012, pp. 133–220. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123838742000023

[36] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[37] B. R. Wheaton, "Wave-particle duality: A modern view," in *Compendium of Quantum Physics*, D. M. Greenberger, K. Hentschel, and F. Weinert, Eds. Springer, 2009, pp. 835–840. [Online]. Available: https://doi.org/10.1007/978-3-540-70626-7_234

[38] O. C. Stoica, "Born rule: quantum probability as classical probability," *CoRR*, vol. abs/2209.08621, 2023.

[39] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, "Elementary gates for quantum computation," *Physical Review A*, vol. 52, no. 5, pp. 3457–3467, nov 1995. [Online]. Available: https://doi.org/10.1103%2Fphysreva.52.3457

[40] D. C. Marinescu and G. M. Marinescu, "Chapter 1 - preliminaries," in *Classical and Quantum Information*, D. C. Marinescu and G. M. Marinescu, Eds. Boston: Academic Press, 2012, pp. 1–131. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780123838742000011

[41] E. Muñoz Coreas and H. Thapliyal, "Everything You Always Wanted to Know About Quantum Circuits," *Wiley Encyclopedia of Electrical and Electronics Engineering*, 8 2022. [Online]. Available: 10.1002/047134608X.W8440

[42] G. Adesso, T. R. Bromley, and M. Cianciaruso, "Measures and applications of quantum correlations," *Journal of Physics A: Mathematical and Theoretical*, vol. 49, no. 47, p. 473001, nov 2016. [Online]. Available: https://doi.org/10.1088%2F1751-8113%2F49%2F47%2F473001

[43] D. G. Bussandri and P. W. Lamberti, "Chapter 7 - quantum metrology and quantum correlations," in *Information Geometry*, ser. Handbook of Statistics, A. Plastino, A. S. Srinivasa Rao, and C. Rao, Eds. Elsevier, 2021, vol. 45, pp. 149–160. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0169716121000250

[44] A. S. Cacciapuoti, M. Caleffi, R. V. Meter, and L. Hanzo, "When entanglement meets classical communications: Quantum teleportation for the quantum internet," *IEEE Trans. Commun.*, vol. 68, no. 6, pp. 3808–3833, 2020. [Online]. Available: https://doi.org/10.1109/TCOMM.2020.2978071

[45] J. Illiano, M. Caleffi, A. Manzalini, and A. S. Cacciapuoti, "Quantum internet protocol stack: A comprehensive survey," *Computer Networks*, vol. 213, p. 109092, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1389128622002250

[46] A. Einstein, M. Born, and H. Born, *The Born-Einstein Letters: Correspondence Between Albert Einstein and Max and Hedwig Born from 1916-1955, with Commentaries by Max Born*. Macmillan, 1971. [Online]. Available: https://books.google.ae/books?id=HvZAAQAAIAAJ

[47] Qiskit contributors, "Qiskit: An Open-source Framework for Quantum Computing," 2023, (Accessed on 10/10/2023). [Online]. Available: https://doi.org/10.5281/zenodo.2562111

[48] V. Bergholm, J. A. Izaac, M. Schuld, C. Gogolin, and N. Killoran, "Pennylane: Automatic differentiation of hybrid quantum-classical computations," *CoRR*, vol. abs/1811.04968, 2018. [Online]. Available: http://arxiv.org/abs/1811.04968

[49] M. Schlosshauer, "Quantum decoherence," *Physics Reports*, vol. 831, pp. 1–57, oct 2019. [Online]. Available: https://doi.org/10.1016%2Fj.physrep.2019.10.001

[50] A. Shaib, M. H. Naim, M. E. Fouda, R. Kanj, and F. J. Kurdahi, "Efficient noise mitigation technique for quantum computing," *CoRR*, vol. abs/2109.05136, 2021. [Online]. Available: https://arxiv.org/abs/2109.05136

[51] W. Heisenberg, "Über den anschaulichen Inhalt der quantentheoretischen Kinematik und Mechanik," *Zeitschrift für Physik*, vol. 33, pp. 879–893, 1925.

[52] W. K. Wootters and W. H. Zurek, "A single quantum cannot be cloned," *Nature*, vol. 299, no. 5886, pp. 802–803, Oct. 1982. [Online]. Available: http://dx.doi.org/10.1038/299802a0

[53] A. Peres, "Reversible logic and quantum computers," *Phys. Rev. A*, vol. 32, pp. 3266–3276, Dec 1985. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.32.3266

[54] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[55] P. W. Shor, "Scheme for reducing decoherence in quantum computer memory," *Phys. Rev. A*, vol. 52, pp. R2493–R2496, Oct 1995. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.52.R2493

[56] R. LaRose, A. Mari, S. Kaiser, P. J. Karalekas, A. A. Alves, P. Czarnik, M. E. Mandouh, M. H. Gordon, Y. Hindy, A. Robertson, P. Thakre, M. Wahl, D. Samuel, R. Mistri, M. Tremblay, N. Gardner, N. T. Stemen, N. Shammah, and W. J. Zeng, "Mitiq: A software package for error mitigation on noisy quantum computers," *Quantum*, vol. 6, p. 774, aug 2022. [Online]. Available: https://doi.org/10.22331%2Fq-2022-08-11-774

[57] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari, and W. J. Zeng, "Digital zero noise extrapolation for quantum error mitigation," in *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*. IEEE, oct 2020. [Online]. Available: https://doi.org/10.1109%2Fqce49297.2020.00045

[58] E. van den Berg, Z. K. Minev, A. Kandala, and K. Temme, "Probabilistic error cancellation with sparse pauli-lindblad models on noisy quantum processors," *Nature Physics*, 2023. [Online]. Available: https://doi.org/10.1038/s41567-023-02042-2

[59] K. Bertels, A. Sarkar, A. M. Krol, R. Budhrani, J. Samadi, E. Geoffroy, J. Matos, R. Abreu, G. G. E. Gielen, and I. Ashraf, "Quantum accelerator stack: A research roadmap," *CoRR*, vol. abs/2102.02035, 2021. [Online]. Available: https://arxiv.org/abs/2102.02035

[60] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, no. 9, pp. 631–633, jul 2014. [Online]. Available: https://doi.org/10.1038%2Fnphys3029

[61] K. Yu, S. Lin, and G.-D. Guo, "Quantum dimensionality reduction by linear discriminant analysis," *Physica A: Statistical Mechanics and its Applications*, vol. 614, p. 128554, mar 2023. [Online]. Available: https://doi.org/10.1016%2Fj.physa.2023.128554

[62] C. H. Papadimitriou and M. Yannakakis, "Optimization, approximation, and complexity classes," *J. Comput. Syst. Sci.*, vol. 43, no. 3, pp. 425–440, 1991. [Online]. Available: https://doi.org/10.1016/0022-0000(91)90023-X

[63] R. B. e. a. Frank Arute, Kunal Arya, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, p. 505–510, 2019. [Online]. Available: https://www.nature.com/articles/s41586-019-1666-5

[64] J. Watrous, "Quantum computational complexity," in *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, Ed. Springer, 2009, pp. 7174–7201. [Online]. Available: https://doi.org/10.1007/978-0-387-30440-3_428

[65] B. Pokharel and D. A. Lidar, "Demonstration of algorithmic quantum speedup," *Physical Review Letters*, vol. 130, no. 21, May 2023. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.130.210602

[66] Y. Liu, S. Arunachalam, and K. Temme, "A rigorous and robust quantum speed-up in supervised machine learning," *Nature Physics*, vol. 17, no. 9, p. 1013–1017, Jul. 2021. [Online]. Available: http://dx.doi.org/10.1038/s41567-021-01287-z

[67] I. Quantum, "The ibm quantum development roadmap," *Solving the scaling problem*, 2022. [Online]. Available: https://www.ibm.com/quantum/roadmap

[68] Honeywell, "Get to Know Honeywell's Latest Quantum Computer," *Honeywell News*, 2022, (Accessed on 10/10/2023). [Online]. Available: https://www.honeywell.com/us/en/news/2020/10/get-to-know-honeywell-s-latest-quantum-computer-system-model-h1

[69] IBM, "IBM Unveils 400 Qubit-Plus Quantum Processor and Next-Generation IBM Quantum System Two," https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two, 2022, (Accessed on 10/10/2023).

[70] Honeywell, "HONEYWELL SYSTEM MODEL H1," *Honeywell Quantum*, 2022, (Accessed on 10/10/2023). [Online]. Available: https://www.honeywell.com/us/en/company/quantum/quantum-computer

[71] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," *Quantum Science and Technology*, vol. 4, no. 4, p. 043001, nov 2019. [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Fab4eb5

[72] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, aug 2021. [Online]. Available: https://doi.org/10.1038%2Fs42254-021-00348-9

[73] V. R. Dasari, M. S. Im, and L. Beshaj, "Solving machine learning optimization problems using quantum computers," *CoRR*, vol. abs/1911.08587, 2019. [Online]. Available: http://arxiv.org/abs/1911.08587

[74] D. Amaro, C. Modica, M. Rosenkranz, M. Fiorentini, M. Benedetti, and M. Lubasch, "Filtering variational quantum algorithms for combinatorial optimization," *Quantum Science and Technology*, vol. 7, no. 1, p. 015021, Jan. 2022. [Online]. Available: http://dx.doi.org/10.1088/2058-9565/ac3e54

[75] L. K. Grover, "A fast quantum mechanical algorithm for database search," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, Philadelphia, Pennsylvania, USA, May 22-24, 1996*, G. L. Miller, Ed. ACM, 1996, pp. 212–219. [Online]. Available: https://doi.org/10.1145/237814.237866

[76] G. Brassard and P. Hoyer, "An exact quantum polynomial-time algorithm for simon's problem," in *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*, ser. ISTCS-97. IEEE Comput. Soc. [Online]. Available: http://dx.doi.org/10.1109/ISTCS.1997.595153

[77] G. Brassard, P. Høyer, M. Mosca, and A. Tapp, "Quantum amplitude amplification and estimation," *Quantum Computation and Information*, p. 53–74, 2002. [Online]. Available: http://dx.doi.org/10.1090/conm/305/05215

[78] L. K. Grover, "Quantum mechanics helps in searching for a needle in a haystack," *Physical Review Letters*, vol. 79, no. 2, p. 325–328, Jul. 1997. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.79.325

[79] D. Grinko, J. Gacon, C. Zoufal, and S. Woerner, "Iterative quantum amplitude estimation," *npj Quantum Information*, vol. 7, no. 1, Mar. 2021. [Online]. Available: http://dx.doi.org/10.1038/s41534-021-00379-1

[80] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *CoRR*, vol. abs/1411.4028, 2014.

[81] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G. H. Booth, and J. Tennyson, "The variational quantum eigensolver: A review of methods and best practices," *Physics Reports*, vol. 986, pp. 1–128, nov 2022. [Online]. Available: https://doi.org/10.1016%2Fj.physrep.2022.08.003

[82] Y. Zhang, "Quantum computing via the bethe ansatz," *Quantum Inf. Process.*, vol. 11, no. 2, pp. 585–590, 2012. [Online]. Available: https://doi.org/10.1007/s11128-011-0268-4

[83] J. J. Park and H. Nha, "Fluctuation theorem for information thermodynamics of quantum correlated systems," *Entropy*, vol. 25, no. 1, p. 165, 2023. [Online]. Available: https://doi.org/10.3390/e25010165

[84] A. Borle and S. J. Lomonaco, "Analyzing the quantum annealing approach for solving linear least squares problems," in *WALCOM: Algorithms and Computation - 13th International Conference, WALCOM 2019, Guwahati, India, February 27 - March 2, 2019, Proceedings*, ser. Lecture Notes in Computer Science, G. K. Das, P. S. Mandal, K. Mukhopadhyaya, and S. Nakano, Eds., vol. 11355. Springer, 2019, pp. 289–301. [Online]. Available: https://doi.org/10.1007/978-3-030-10564-8_23

[85] K. Jun, "Qubo formulations for numerical quantum computing," *CoRR*, vol. abs/2106.10819, 2022.

[86] H. B. Callen, *Thermodynamics and an introduction to thermostatistics; 2nd ed.* New York, NY: Wiley, 1985. [Online]. Available: https://cds.cern.ch/record/450289

[87] M. Panella and G. Martinelli, "Neural networks with quantum architecture and quantum learning," *Int. J. Circuit Theory Appl.*, vol. 39, no. 1, pp. 61–77, 2011. [Online]. Available: https://doi.org/10.1002/cta.619

[88] M. Capra, B. Bussolino, A. Marchisio, G. Masera, M. Martina, and M. Shafique, "Hardware and software optimizations for accelerating deep neural networks: Survey of current trends, challenges, and the road ahead," *IEEE Access*, vol. 8, pp. 225 134–225 180, 2020. [Online]. Available: https://doi.org/10.1109/ACCESS.2020.3039858

[89] I. Cong, S. Choi, and M. D. Lukin, "Quantum convolutional neural networks," *Nature Physics*, vol. 15, no. 12, pp. 1273–1278, aug 2019. [Online]. Available: https://doi.org/10.1038%2Fs41567-019-0648-8

[90] M. Henderson, S. Shakya, S. Pradhan, and T. Cook, "Quanvolutional neural networks: powering image recognition with quantum circuits," *Quantum Mach. Intell.*, vol. 2, no. 1, pp. 1–9, 2020. [Online]. Available: https://doi.org/10.1007/s42484-020-00012-y

[91] J. Bausch, "Recurrent quantum neural networks," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., 2020. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/0ec96be397dd6d3cf2fecb4a2d627c1c-Abstract.html

[92] Y. Li, Z. Wang, R. Han, S. Shi, J. Li, R. Shang, H. Zheng, G. Zhong, and Y. Gu, "Quantum recurrent neural networks for sequential learning," *Neural Networks*, vol. 166, pp. 148–161, 2023. [Online]. Available: https://doi.org/10.1016/j.neunet.2023.07.003

[93] H. Huang, Y. Du, M. Gong, Y. Zhao, Y. Wu, C. Wang, S. Li, F. Liang, J. Lin, Y. Xu, R. Yang, T. Liu, M. Hsieh, H. Deng, H. Rong, C. Peng, C. Lu, Y. Chen, D. Tao, X. Zhu, and J. Pan, "Experimental quantum generative adversarial networks for image generation," *CoRR*, vol. abs/2010.06201, 2020. [Online]. Available: https://arxiv.org/abs/2010.06201

[94] J. Romero, J. P. Olson, and A. Aspuru-Guzik, "Quantum autoencoders for efficient compression of quantum data," *Quantum Science and Technology*, vol. 2, no. 4, p. 045001, aug 2017. [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Faa8072

[95] Y. Liao, D. Ebler, F. Liu, and O. Dahlsten, "Quantum advantage in training binary neural networks," *CoRR*, vol. abs/1810.12948, 2019.

[96] S. L. Alarcon, C. Merkel, M. Hoffnagle, S. Ly, and A. Pozas-Kerstjens, "Accelerating the training of single layer binary neural networks using the hhl quantum algorithm," in *2022 IEEE 40th International Conference on Computer Design (ICCD)*, 2022, pp. 427–433.

[97] J. Carrasquilla, M. Hibat-Allah, E. Inack, A. Makhzani, K. Neklyudov, G. W. Taylor, and G. Torlai, "Quantum hypernetworks: Training binary neural networks in quantum superposition," *CoRR*, vol. abs/2301.08292, 2023.

[98] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, "Data re-uploading for a universal quantum classifier," *Quantum*, vol. 4, p. 226, Feb. 2020. [Online]. Available: http://dx.doi.org/10.22331/q-2020-02-06-226

[99] Z. Yu, H. Yao, M. Li, and X. Wang, "Power and limitations of single-qubit native quantum neural networks," *CoRR*, vol. abs/2205.07848, 2022.

[100] E. Peña Tapia, G. Scarpa, and A. Pozas-Kerstjens, "A didactic approach to quantum machine learning with a single qubit," *Physica Scripta*, vol. 98, no. 5, p. 054001, Apr. 2023. [Online]. Available: http://dx.doi.org/10.1088/1402-4896/acc5b8

[101] T. Goto, Q. H. Tran, and K. Nakajima, "Universal approximation property of quantum machine learning models in quantum-enhanced feature spaces," *Phys. Rev. Lett.*, vol. 127, p. 090506, Aug 2021. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.127.090506

[102] A. Pérez-Salinas, D. López-Núñez, A. García-Sáez, P. Forn-Díaz, and J. I. Latorre, "One qubit as a universal approximant," *Physical Review A*, vol. 104, no. 1, Jul. 2021. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.104.012405

[103] E. Torabian and R. V. Krems, "Compositional optimization of quantum circuits for quantum kernels of support vector machines," *CoRR*, vol. abs/2203.13848, 2023.

[104] Z. Krunic, F. Flother, G. Seegan, N. Earnest-Noble, and S. Omar, "Quantum kernels for real-world predictions based on electronic health records," *IEEE Transactions on Quantum Engineering*, vol. 3, pp. 1–11, 2022. [Online]. Available: https://doi.org/10.1109%2Ftqe.2022.3176806

[105] T. Sancho-Lorente, J. Romá n-Roche, and D. Zueco, "Quantum kernels to learn the phases of quantum matter," *Physical Review A*, vol. 105, no. 4, apr 2022. [Online]. Available: https://doi.org/10.1103%2Fphysreva.105.042432

[106] D. Horn and A. Gottlieb, "Algorithm for data clustering in pattern recognition problems based on quantum mechanics," *Physical review letters*, vol. 88, p. 018702, 02 2002.

[107] E. Aïmeur, G. Brassard, and S. Gambs, "Quantum clustering algorithms," in *Machine Learning, Proceedings of the Twenty-Fourth International Conference (ICML 2007), Corvallis, Oregon, USA, June 20-24, 2007*, ser. ACM International Conference Proceeding Series, Z. Ghahramani, Ed., vol. 227. ACM, 2007, pp. 1–8. [Online]. Available: https://doi.org/10.1145/1273496.1273497

[108] Z. Zhao, A. Pozas-Kerstjens, P. Rebentrost, and P. Wittek, "Bayesian deep learning on a quantum computer," *Quantum Machine Intelligence*, vol. 1, no. 1–2, p. 41–51, May 2019. [Online]. Available: http://dx.doi.org/10.1007/s42484-019-00004-7

[109] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *CoRR*, vol. abs/1803.11173, 2018. [Online]. Available: http://arxiv.org/abs/1803.11173

[110] J. Mancilla and C. Pere, "A preprocessing perspective for quantum machine learning classification advantage in finance using NISQ algorithms," *Entropy*, vol. 24, no. 11, p. 1656, 2022. [Online]. Available: https://doi.org/10.3390/e24111656

[111] D. Sierra-Sosa, J. D. Arcila-Moreno, B. Garcia-Zapirain, and A. Elmaghraby, "Diabetes type 2: Poincaré data preprocessing for quantum machine learning," *Computers, Materials & Continua*, vol. 67, no. 2, pp. 1849–1861, 2021. [Online]. Available: http://www.techscience.com/cmc/v67n2/41306

[112] R. LaRose and B. Coyle, "Robust data encodings for quantum classifiers," *CoRR*, vol. abs/2003.01695, 2020. [Online]. Available: https://arxiv.org/abs/2003.01695

[113] G. Li, R. Ye, X. Zhao, and X. Wang, "Concentration of data encoding in parameterized quantum circuits," in *NeurIPS*, 2022. [Online]. Available: http://papers.nips.cc/paper_files/paper/2022/hash/7b2d0730df1edd8c97df4bf83696025d-Abstract-Conference.html

[114] M. Weigold, J. Barzen, F. Leymann, and M. Salm, "Expanding data encoding patterns for quantum algorithms," in *18th IEEE International Conference on Software Architecture Companion, ICSA Companion 2021, Stuttgart, Germany, March 22-26, 2021*. IEEE, 2021, pp. 95–101. [Online]. Available: https://doi.org/10.1109/ICSA-C52384.2021.00025

[115] M. Schuld and F. Petruccione, *Supervised Learning with Quantum Computers*, 1st ed. Springer Publishing Company, Incorporated, 2018.

[116] D. K. Park, F. Petruccione, and J.-K. K. Rhee, "Circuit-based quantum random access memory for classical data," *Scientific Reports*, vol. 9, no. 1, mar 2019. [Online]. Available: https://doi.org/10.1038%2Fs41598-019-40439-3

[117] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, mar 2019. [Online]. Available: https://doi.org/10.1038%2Fs41586-019-0980-2

[118] E. Perrier, A. Youssry, and C. Ferrie, "Qdataset, quantum datasets for machine learning," *Scientific Data*, vol. 9, 09 2022.

[119] M. Rupp, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Fast and accurate modeling of molecular atomization energies with machine learning," *Physical Review Letters*, vol. 108, p. 058301, 2012.

[120] G. Montavon, M. Rupp, V. Gobre, A. Vazquez-Mayagoitia, K. Hansen, A. Tkatchenko, K.-R. Müller, and O. A. von Lilienfeld, "Machine learning of molecular electronic properties in chemical compound space," *New Journal of Physics*, vol. 15, no. 9, p. 095003, 2013. [Online]. Available: http://stacks.iop.org/1367-2630/15/i=9/a=095003

[121] R. Ramakrishnan, M. Hartmann, E. Tapavicza, and O. A. von Lilienfeld, "Electronic spectra from TDDFT and machine learning in chemical space," *The Journal of Chemical Physics*, vol. 143, no. 8, p. 084111, 08 2015. [Online]. Available: https://doi.org/10.1063/1.4928757

[122] R. Ramakrishnan, P. O. Dral, M. Rupp, and O. A. von Lilienfeld, "Quantum chemistry structures and properties of 134 kilo molecules," *Scientific Data*, vol. 1, 2014.

[123] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," *CoRR*, vol. abs/1802.06002, 2018.

[124] G. Acampora, R. Schiattarella, and A. Troiano, "A dataset for quantum circuit mapping," *Data in Brief*, vol. 39, p. 107526, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352340921008027

[125] J. P. Zwolak, S. S. Kalantre, X. Wu, S. Ragole, and J. M. Taylor, "Qflow lite dataset: A machine-learning approach to the charge states in quantum dot experiments," *PLOS ONE*, vol. 13, no. 10, pp. 1–17, 10 2018. [Online]. Available: https://doi.org/10.1371/journal.pone.0205844

[126] L. Schatzki, A. Arrasmith, P. J. Coles, and M. Cerezo, "Entangled datasets for quantum machine learning," *CoRR*, vol. abs/2109.03400, 2021. [Online]. Available: https://arxiv.org/abs/2109.03400

[127] L. C. Blum and J.-L. Reymond, "970 million druglike small molecules for virtual screening in the chemical universe database GDB-13," *J. Am. Chem. Soc.*, vol. 131, p. 8732, 2009.

[128] L. Ruddigkeit, R. van Deursen, L. C. Blum, and J. Reymond, "Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17," *J. Chem. Inf. Model.*, vol. 52, no. 11, pp. 2864–2875, 2012. [Online]. Available: https://doi.org/10.1021/ci300415d

[129] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. [Online]. Available: https://doi.org/10.1109/5.726791

[130] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms," *CoRR*, vol. abs/1708.07747, 2017. [Online]. Available: http://arxiv.org/abs/1708.07747

[131] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, "Power of data in quantum machine learning," *Nature Communications*, vol. 12, no. 1, may 2021. [Online]. Available: https://doi.org/10.1038%2Fs41467-021-22539-9

[132] IBM, "Quantum Lab — IBM Quantum Learning," https://quantum-computing.ibm.com/lab, (Accessed on 10/10/2023).

[133] IonQ, "Compare Quantum Systems," https://ionq.com/quantum-systems/compare, (Accessed on 10/10/2023).

[134] L. Madsen, F. Laudenbach, M. Askarani, F. Rortais, T. Vincent, J. Bulmer, F. Miatto, L. Neuhaus, L. Helt, M. Collins, A. Lita, T. Gerrits, S. Nam, V. Vaidya, M. Menotti, I. Dhand, Z. Vernon, N. Quesada, and J. Lavoie, "Quantum computational advantage with a programmable photonic processor," *Nature*, vol. 606, pp. 75–81, 06 2022.

[135] C. McGeoch and P. Farre, "The D-Wave Advantage System: An Overview," Tech. Rep., 09 2020.

[136] IBM, "Hardware — IBM Quantum Documentation," https://docs.quantum-computing.ibm.com, (Accessed on 10/10/2023).

[137] J.-S. Chen, E. Nielsen, M. Ebert, V. Inlek, K. Wright, V. Chaplin, A. Maksymov, E. Páez, A. Poudel, P. Maunz, and J. Gamble, "Benchmarking a trapped-ion quantum computer with 29 algorithmic qubits," *CoRR*, vol. abs/2308.05071, 2023.

[138] J. M. Arrazola, V. Bergholm, K. Brádler, T. R. Bromley, M. J. Collins, I. Dhand, A. Fumagalli, T. Gerrits, A. Goussev, L. G. Helt, J. Hundal, T. Isacsson, R. B. Israel, J. Izaac, S. Jahangiri, R. Janik, N. Killoran, S. P. Kumar, J. Lavoie, A. E. Lita, D. H. Mahler, M. Menotti, B. Morrison, S. W. Nam, L. Neuhaus, H. Y. Qi, N. Quesada, A. Repingon, K. K. Sabapathy, M. Schuld, D. Su, J. Swinarton, A. Száva, K. Tan, P. Tan, V. D. Vaidya, Z. Vernon, Z. Zabaneh, and Y. Zhang, "Quantum circuits with many photons on a programmable nanophotonic chip," *Nature*, vol. 591, no. 7848, pp. 54–60, mar 2021. [Online]. Available: https://doi.org/10.1038%2Fs41586-021-03202-1

[139] Google, "A Preview of Bristlecone, Google's New Quantum Processor," https://blog.research.google/2018/03/a-preview-of-bristlecone-googles-new.html, march 2018, (Accessed on 10/10/2023).

[140] "Google Quantum AI," https://quantumai.google/, 2016, (Accessed on 10/10/2023).

[141] R. Computing, "Building scalable, innovative quantum systems," https://www.rigetti.com/what-we-build, (Accessed on 10/10/2023).

[142] Intel, "Intel Debuts 2nd-Gen Horse Ridge Cryogenic Quantum Control Chip — intel.com," https://www.intel.com/content/www/us/en/newsroom/news/2nd-gen-horse-ridge-cryogenic-quantum-control-chip.html#gs.4h1l59, (Accessed on 10/10/2023).

[143] Quantinuum, "Hardware — The Future of Quantum Hardware," https://www.quantinuum.com/hardware, (Accessed on 10/10/2023).

[144] K. Barnes, P. Battaglino, B. J. Bloom, K. Cassella, R. Coxe, N. Crisosto, J. P. King, S. S. Kondov, K. Kotru, S. C. Larsen, J. Lauigan, B. J. Lester, M. McDonald, E. Megidish, S. Narayanaswami, C. Nishiguchi, R. Notermans, L. S. Peng, A. Ryou, T.-Y. Wu, and M. Yarwood, "Assembly and coherent control of a register of nuclear spin qubits," *Nature Communications*, vol. 13, no. 1, may 2022. [Online]. Available: https://doi.org/10.1038%2Fs41467-022-29977-z

[145] J. Wurtz, A. Bylinskii, B. Braverman, J. Amato-Grill, S. H. Cantu, F. Huber, A. Lukin, F. Liu, P. Weinberg, J. Long, S.-T. Wang, N. Gemelke, and A. Keesling, "Aquila: Quera's 256-qubit neutral-atom quantum computer," *CoRR*, vol. abs/2306.11727, 2023.

[146] J. King, S. Yarkoni, J. Raymond, I. Ozfidan, A. D. King, M. M. Nevisi, J. P. Hilton, and C. C. McGeoch, "Quantum Annealing amid Local Ruggedness and Global Frustration," Tech. Rep., 03 2017.

[147] D. P. DiVincenzo, "The physical implementation of quantum computation," *Fortschritte der Physik*, vol. 48, no. 9-11, pp. 771–783, sep 2000. [Online]. Available: https://doi.org/10.1002%2F1521-3978%28200009%2948%3A9%2F11%3C771%3A%3Aaid-prop771%3E3.0.co%3B2-e

[148] A. F. Kockum and F. Nori, "Quantum bits with josephson junctions," in *Fundamentals and Frontiers of the Josephson Effect*. Springer International Publishing, 2019, pp. 703–741. [Online]. Available: https://doi.org/10.1007%2F978-3-030-20726-7_17

[149] S. Bravyi, O. Dial, J. M. Gambetta, D. Gil, and Z. Nazario, "The future of quantum computing with superconducting qubits," *Journal of Applied Physics*, vol. 132, no. 16, p. 160902, oct 2022. [Online]. Available: https://doi.org/10.1063%2F5.0082975

[150] A. Salmanogli, "Entanglement engineering by transmon qubit in a circuit qed," *CoRR*, vol. abs/2109.00316, 2021.

[151] S. P. Harvey, "Quantum dots/spin qubits," feb 2022. [Online]. Available: https://doi.org/10.1093%2Facrefore%2F9780190871994.013.83

[152] L. Henriet, L. Beguin, A. Signoles, T. Lahaye, A. Browaeys, G.-O. Reymond, and C. Jurczak, "Quantum computing with neutral atoms," *Quantum*, vol. 4, p. 327, sep 2020. [Online]. Available: https://doi.org/10.22331%2Fq-2020-09-21-327

[153] X.-F. Shi, "Quantum logic and entanglement by neutral rydberg atoms: methods and fidelity," *Quantum Science and Technology*, vol. 7, no. 2, p. 023002, apr 2022. [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Fac18b8

[154] S. Slussarenko and G. J. Pryde, "Photonic quantum information processing: A concise review," *Applied Physics Reviews*, vol. 6, no. 4, p. 041303, dec 2019. [Online]. Available: https://doi.org/10.1063%2F1.5115814

[155] J.-Y. Tsai, J. Pan, H. Lin, A. Bansil, and Q. Yan, "Antisite defect qubits in monolayer transition metal dichalcogenides," *Nature Communications*, vol. 13, no. 1, jan 2022. [Online]. Available: https://doi.org/10.1038%2Fs41467-022-28133-x

[156] C. Tutschku, R. W. Reinthaler, C. Lei, A. H. MacDonald, and E. M. Hankiewicz, "Majorana-based quantum computing in nanowire devices," *Physical Review B*, vol. 102, no. 12, sep 2020. [Online]. Available: https://doi.org/10.1103%2Fphysrevb.102.125407

[157] M. Aghaee, A. Akkala, Z. Alam, R. Ali, A. A. Ramirez, M. Andrzejczuk, A. E. Antipov, P. Aseev, M. Astafev, B. Bauer, J. Becker, S. Boddapati, F. Boekhout, J. Bommer, T. Bosma, L. Bourdet, S. Boutin, P. Caroff, L. Casparis, M. Cassidy, S. Chatoor, A. W. Christensen, N. Clay, W. S. Cole, F. Corsetti, A. Cui, P. Dalampiras, A. Dokania, G. de Lange, M. de Moor, J. C. E. Saldaña, S. Fallahi, Z. H. Fathabad, J. Gamble, G. Gardner, D. Govender, F. Griggio, R. Grigoryan, S. Gronin, J. Gukelberger, E. B. Hansen, S. Heedt, J. H. Zamorano, S. Ho, U. L. Holgaard, H. Ingerslev, L. Johansson, J. Jones, R. Kallaher, F. Karimi, T. Karzig, C. King, M. E. Kloster, C. Knapp, D. Kocon, J. Koski, P. Kostamo, P. Krogstrup, M. Kumar, T. Laeven, T. Larsen, K. Li, T. Lindemann, J. Love, R. Lutchyn, M. H. Madsen, M. Manfra, S. Markussen, E. Martinez, R. McNeil, E. Memisevic, T. Morgan, A. Mullally, C. Nayak, J. Nielsen, W. H. P. Nielsen, B. Nijholt, A. Nurmohamed, E. O'Farrell, K. Otani, S. Pauka, K. Petersson, L. Petit, D. I. Pikulin, F. Preiss, M. Quintero-Perez, M. Rajpalke, K. Rasmussen, D. Razmadze, O. Reentila, D. Reilly, R. Rouse, I. Sadovskyy, L. Sainiemi, S. Schreppler, V. Sidorkin, A. Singh, S. Singh, S. Sinha, P. Sohr, T. Stankevič, L. Stek, H. Suominen, J. Suter, V. Svidenko, S. Teicher, M. Temuerhan, N. Thiyagarajah, R. Tholapi, M. Thomas, E. Toomey, S. Upadhyay, I. Urban, S. Vaitiekėnas, K. V. Hoogdalem, D. V. Woerkom, D. V. Viazmitinov, D. Vogel, S. Waddy, J. Watson, J. Weston, G. W. Winkler, C. K. Yang, S. Yau, D. Yi, E. Yucelen, A. Webster, R. Zeisel, and R. Z. and, "InAs-al hybrid devices passing the topological gap protocol," *Physical Review B*, vol. 107, no. 24, jun 2023. [Online]. Available: https://doi.org/10.1103%2Fphysrevb.107.245423

[158] T. Itoko, R. Raymond, T. Imamichi, and A. Matsuo, "Optimization of quantum circuit mapping using gate transformation and commutation," *Integration*, vol. 70, pp. 43–50, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167926019302755

[159] R. Wille and L. Burgholzer, "Mqt qmap: Efficient quantum circuit mapping," in *Proceedings of the 2023 International Symposium on Physical Design*, ser. ISPD '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 198–204. [Online]. Available: https://doi.org/10.1145/3569052.3578928

[160] N. Killoran, J. Izaac, N. Quesada, V. Bergholm, M. Amy, and C. Weedbrook, "Strawberry Fields: A software platform for photonic quantum computing," *Quantum*, vol. 3, p. 129, 2019.

[161] R. S. Smith, M. J. Curtis, and W. J. Zeng, "A practical quantum instruction set architecture," *CoRR*, vol. abs/1608.03355, 2016. [Online]. Available: http://arxiv.org/abs/1608.03355

[162] P. J. Karalekas, N. A. Tezak, E. C. Peterson, C. A. Ryan, M. P. da Silva, and R. S. Smith, "A quantum-classical cloud platform

optimized for variational hybrid algorithms," *Quantum Science and Technology*, vol. 5, no. 2, p. 024003, apr 2020. [Online]. Available: https://doi.org/10.1088%2F2058-9565%2Fab7559

[163] "Quantinuum TKET," https://www.quantinuum.com/developers/tket, (Accessed on 10/10/2023).

[164] C. Developers, "Cirq," Jul. 2023, See full list of authors on Github: https://github .com/quantumlib/Cirq/graphs/contributors. [Online]. Available: https://doi.org/10.5281/zenodo.8161252

[165] J. R. McClean, N. C. Rubin, K. J. Sung, I. D. Kivlichan, X. Bonet-Monroig, Y. Cao, C. Dai, E. S. Fried, C. Gidney, B. Gimby, P. Gokhale, T. Häner, T. Hardikar, V. Havlíček, O. Higgott, C. Huang, J. Izaac, Z. Jiang, X. Liu, S. McArdle, M. Neeley, T. O'Brien, B. O'Gorman, I. Ozfidan, M. D. Radin, J. Romero, N. P. D. Sawaya, B. Senjean, K. Setia, S. Sim, D. S. Steiger, M. Steudtner, Q. Sun, W. Sun, D. Wang, F. Zhang, and R. Babbush, "Openfermion: the electronic structure package for quantum computers," *Quantum Science and Technology*, vol. 5, no. 3, p. 034014, jun 2020. [Online]. Available: https://dx.doi.org/10.1088/2058-9565/ab8ebc

[166] M. Broughton, G. Verdon, T. McCourt, A. J. Martinez, J. H. Yoo, S. V. Isakov, P. Massey, M. Y. Niu, R. Halavati, E. Peters, M. Leib, A. Skolik, M. Streif, D. V. Dollen, J. R. McClean, S. Boixo, D. Bacon, A. K. Ho, H. Neven, and M. Mohseni, "Tensorflow quantum: A software framework for quantum machine learning," *CoRR*, vol. abs/2003.02989, 2020. [Online]. Available: https://arxiv.org/abs/2003.02989

[167] "The Leap™ Quantum Cloud Service — D-Wave," https://www.dwavesys.com/solutions-and-products/cloud-platform/, (Accessed on 10/10/2023).

[168] "qBraid," https://www.qbraid.com/, (Accessed on 10/10/2023).

[169] "IonQ Quantum Cloud," https://ionq.com/quantum-cloud, (Accessed on 10/10/2023).

[170] Amazon Web Services, "Amazon Braket," 2020. [Online]. Available: https://aws.amazon.com/braket/

[171] "Azure Quantum - Quantum Cloud Computing Service," https://azure.microsoft.com/en-us/products/quantum, (Accessed on 10/10/2023).

[172] "Build and deploy Industrial Generative AI applications on Orquestra," https://zapata.ai/orquestra-platform/, (Accessed on 10/10/2023).

[173] "Xanadu — Welcome to Xanadu," https://www.xanadu.ai/, (Accessed on 10/10/2023).

[174] E. Epifanovsky *et al.*, "Software for the frontiers of quantum chemistry: An overview of developments in the Q-Chem 5 package," *The Journal of Chemical Physics*, vol. 155, no. 8, p. 084801, 08 2021. [Online]. Available: https://doi.org/10.1063/5.0055522

[175] R. Santagati, A. Aspuru-Guzik, R. Babbush, M. Degroote, L. Gonzalez, E. Kyoseva, N. Moll, M. Oppel, R. M. Parrish, N. C. Rubin, M. Streif, C. S. Tautermann, H. Weiss, N. Wiebe, and C. Utschig-Utschig, "Drug design on quantum computers," *CoRR*, vol. abs/2301.04114, 2023.

[176] M. Lourenço, L. Barrios Herrera, J. Hostaš, P. Calaminici, A. Köster, A. Tchagang, and D. Salahub, "Qmlmaterial-a quantum machine learning software for material design and discovery," *Journal of chemical theory and computation*, vol. 19, 08 2023.

[177] C. Degen, F. Reinhard, and P. Cappellaro, "Quantum sensing," *Reviews of Modern Physics*, vol. 89, no. 3, jul 2017. [Online]. Available: https://doi.org/10.1103%2Frevmodphys.89.035002

[178] N. Aslam, H. Zhou, and E. e. a. Urbach, "Quantum sensors for biomedical applications." *Nat Rev Phys*, vol. 5, p. 157–169, 2023. [Online]. Available: https://www.baesystems.com/en-us/definition/what-is-quantum-sensing

[179] T. Ohshima, "Toward real application of quantum sensing and metrology," *Front. Quantum. Sci. Technol. Sec. Quantum Sensing and Metrology*, vol. 1, p. 157–169, 2022. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frqst.2022.998459/full

[180] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, p. 120–126, feb 1978. [Online]. Available: https://doi.org/10.1145/359340.359342

[181] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, 1994, pp. 124–134.

[182] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, vol. 26, no. 5, pp. 1484–1509, oct 1997. [Online]. Available: https://doi.org/10.1137%2Fs0097539795293172

[183] M. Campagna, L. Chen, Ö. Dagdelen, J. Ding, J. K. Fernick, N. Gisin, D. Hayford, T. Jennewein, N. Lütkenhaus, M. Mosca, B. Neill, M. Pecen, R. Perlner, G. Ribordy, J. M. Schanck, D. Stebila, N. Walenta, W. Whyte, and Z. Zhang, "Quantum safe cryptography and security: An introduction, benefits, enablers and challengers," ETSI (European Telecommunications Standards Institute), Tech. Rep., June 2015. [Online]. Available: http://www.etsi.org/images/files/ETSIWhitePapers/QuantumSafeWhitepaper.pdf

[184] D. Stebila and M. Mosca, "Post-quantum key exchange for the internet and the open quantum safe project," in *Selected Areas in Cryptography - SAC 2016 - 23rd International Conference, St. John's, NL, Canada, August 10-12, 2016, Revised Selected Papers*, ser. Lecture Notes in Computer Science, R. Avanzi and H. M. Heys, Eds., vol. 10532. Springer, 2016, pp. 14–37. [Online]. Available: https://doi.org/10.1007/978-3-319-69453-5_2

[185] M. Bozzio, M. Vyvlecka, M. Cosacchi, C. Nawrath, T. Seidelmann, J. C. Loredo, S. L. Portalupi, V. M. Axt, P. Michler, and P. Walther, "Enhancing quantum cryptography with quantum dot single-photon sources," *npj Quantum Information*, vol. 8, no. 1, sep 2022. [Online]. Available: https://doi.org/10.1038%2Fs41534-022-00626-z

[186] R. Orús, S. Mugel, and E. Lizaso, "Quantum computing for finance: Overview and prospects," *Reviews in Physics*, vol. 4, p. 100028, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405428318300571

[187] L. Babai, P. Frankl, and J. Simon, "Complexity classes in communication complexity theory (preliminary version)," in *27th Annual Symposium on Foundations of Computer Science, Toronto, Canada, 27-29 October 1986*. IEEE Computer Society, 1986, pp. 337–347. [Online]. Available: https://doi.org/10.1109/SFCS.1986.15

Appendix A
Complexity Theory

This section provides an overview of complexity classes. A *promise problem* is a decision problem where the input is to be selected from all possible input strings. A problem with an explicitly defined structure or characteristics of input is defined in advance, making the problem environment-entrusted with certain types of promised input with no instance of input sent in unexpectedly or out of definition. All promise problems are assigned to a complexity class suitable to their nature and requirements. Formally, *complexity classes* are a group of computational problems that have similar resource-based complexity [187]. In classical computing, there exist the following classes:

- *Polynomial Class(P)*: It contains all decision problems that can be solved by a deterministic Turing machine using a polynomial amount of computation time, or polynomial time.
- *Non-deterministic Polynomial (NP)*: It is the collection of decision problems that can be solved by a non-deterministic machine in polynomial time but these problems of NP can be verified by a Turing machine in polynomial time.
- *NP-hard*: An NP-hard problem is at least as hard as the hardest problem in NP and it is a class of problems such that every problem in NP reduces to NP-hard. Since it takes a long time to verify their solution, all NP-hard problems are not in NP.
- *NP-complete*: A problem is NP-complete if it is both NP and NP-hard. NP-complete problems are the hardest problems in the NP space. If one could solve an NP-complete problem in polynomial time, then one could also solve any NP problem in polynomial time.

APPENDIX B
QUANTUM TECHNICAL CONCEPTS AND DEFINITIONS

A. Hilbert Space

In Quantum mechanics the state of a physical system is represented by a vector in the Hilbert space, which is a complex vector space with an inner product. In the finite-dimensional complex vector spaces that come up in quantum computation and quantum information, the Hilbert space is exactly the same thing as an inner product space (both terms can be used interchangeably)[2].

B. State Space and State Vector

The Hilbert space is associated with any isolated physical system, known as the state space of the system [36]. The system is completely described by its state vector, which is a unit vector in the system's state space.

C. Schrodinger's Equation

The Schrödinger's equation (see Equation (4)) describes the evolution of the state ψ of a closed quantum system over time t [36].

$$i\hbar\frac{\partial\psi}{\partial t} = \hat{H}\psi \tag{4}$$

D. Hermitian Matrix/Operator

In the Schrödinger's equation, $\hat{H}$ is the hermitian operator, known as the Hamiltonian of the closed system. An operator *A* whose adjoint is *A* is known as a Hermitian or self-adjoint operator. It is a complex square matrix that is equal to its own conjugate transpose.

1) Hamiltonian: In principle, if we know the Hamiltonian of a system, together with the Planck's constant ($\hbar$) in the Schröndinger's equation, we can understand the complete dynamics of the physical system. In reality, determining the Hamiltonian of a physical system is a very difficult task (hard problem). Quantum computing represents a valid avenue toward realizing such physical systems with computations closer to real systems. In quantum computation and information, we usually consider the Hamiltonian as a starting point. Since the Hamiltonian is a Hermitian operator, it has spectral decomposition with eigenvalues corresponding to the normalized eigenvectors. The states $|E\rangle$ are conventionally referred to as energy eigenstates or sometimes stationed states and *E* is the energy of the state. The lowest energy state is the ground energy with the corresponding eigenstate known as the ground state.

E. Observable

Projective measurements are a special case of general measurements that have the ability to perform unitary transformations. A projective measurement is described by an observable (*M*) and a Hermitian operator on the state space of the system being observed. The observable has a spectral decomposition. The completeness relation, whose probabilities sum to 1, is applicable to projective measurements and makes the probability constraint valid. This way of computing measurements is related to the Heisenberg uncertainty principle [36].

F. Heisenberg's Uncertainty Principle

This principle states that we cannot know both the position and speed of a particle, such as a photon or electron, with perfect accuracy. The more we accurately know the position of the particle, the less we know about its speed, and vice versa.

[2]For infinite dimensions, the Hilbert space satisfies additional technical restrictions above and beyond inner product spaces [36].