The state of Quantum Computing in 2025

Adam Prior.[1]

1. *UrbanFox, Dublin, Ireland*

Abstract
This paper provides an accessible overview of the current state of quantum computing technologies as of September 2025. Aimed at readers with a technical background but no prior experience in quantum computing, it reviews foundational concepts, historical advancements, and leading hardware and software platforms. The paper also discusses practical challenges, potential applications, and the outlook for future developments in the field.

Keywords: Quantum Computing; Qubits; Decoherence; robotics

1. Introduction

Define Quantum Advantage here. Define Some other terms here.

todo: add to historical section that Shor's algorithm is somewhat of a benchmark for quantum computing

2. Historical Advancements in Quantum Computing

The first mentions of quantum computation can be traced as far back as the early as 1980s. In 1980, mathematician Yuri Manin discussed the concept of a quantum computer in his paper 'Computable and Uncomputable'[**Manin1980**]. Richard Feynman, in 1982, published a paper 'Simulating physics with computers' introducing the idea of simulating quantum systems using quantum computers[**Feynman1982**], highlighting the limitations of classical computers at simulating the exponentially growing state space of quantum systems and how quantum systems can be more efficiently simulated.

Given these initial few years of defining work, the field began to gain traction, followed by another *10 years* of foundational theoretical developments. In 1985, David Deutsch further developed the quantum computing theory by rigorously defining the Universal Quantum Turing Machine (the quantum analog of a classical Turing machine) capable of performing any computation that a classical computer can, but more efficiently for certain problems[**Deutsch1985**]. Further theoretical advancements continued with contributions from, reversible quantum computation by Paul Benioff between 1985-1987[**BBenioff1987**], the first proposal of a physically realizable quantum computer using atoms and photons in 1998[**Yamamoto1998**], and attempts to define quantum complexity classes in 1989[**Bernstein1993**] which would eventually lead to the definition of Bounded-Error Quantum Polynomial Time (BQP) problems, an important class of problems in complexity theory that can be efficiently solved by quantum computers in polynomial time.

It was not until 1992 that the first quantum algorithm was proposed by Deutsch and Jozsa[**Deutsch1992**], demonstrating that quantum computers could solve certain problems more efficiently than classical computers (quantum advantage). While this algorithm has very limited practical applications, it paved the way for more impactful algorithms. The most famous of these is Shor's algorithm, proposed by Peter Shor in 1994[**Shor1994**]. Shor's algorithm is an efficient quantum algorithm for integer factorization in polynomial time, versus the best-known classical algorithms which run in sub-exponential time. This algoirithm in some sense 'proved' the potential of quantum computing to solve practical problems that are intractable for classical computers, particularly in the context of cryptography, as many encryption schemes rely on the difficulty of factoring large integers. Another significant algorithm is Grover's algorithm, proposed by Lov Grover two years later[**Grover1996**], providing quadratic speedup for unstructured search problems.

While these algorithms demonstrate the theoretical potential of quantum computers, they would be useless without the ability to assemble a physical quantum computer. Around the same time as these algorithmic developments, experimental researchers were developing the first real qubits and quantum gates. The first experimental demonstration of a quantum logic gate was realised in 1995 by a team using an electromagnetic trap to confine ions in place to form a two-qubit system. The energy levels of the ions are used to represent the qubit states ($|0\rangle$ and $|1\rangle$), while laser pulses are used to manipulate these states. The team successfully implemented a controlled-NOT (CNOT) gate, which is a fundamental quantum gate universal quantum gate used in many quantum algorithms[1]. One might define this point in time as the take-off point in the history of quantum computing as it was demonstrated, roughly in parallel, to be both theoretically computationally advantageous (for certain classes of problems) and experimentally realisable and a large acceleration in research and development followed. Over the next decade experiments would demonstrate more complex gates, and increased amount of qubits (8 qubit registers by December 2005)[**https://www.nature.com/articles/nature04279**], and even new quantum systems that can implement qubits such as superconducting states, nuclear spins, quantum dots and even photon polarization[1]. Theoretical advancements also continued, with the development of quantum error correction codes and various proofs of principle for quantum algorithms that further highlighted feasability and limitations.

Up to this point, it is well known that quantum computers suffer from an extreme scaling problems. Qubits are extremely sensitive to their environment and suffer badly from quantum decoherence which limits the rate at which additional qubits can be added to a system before the entire system becomes unstable highlighting why it took so long to scale beyond only a few qubits. Table 1 summarizes some of the key leaps in qubiit counts over the years.

3. Fundamentals of Quantum Computing

In order to properly understand the current state of quantum computing, it is important to first understand some of the fundamental concepts. What exactly is a qubit? What is superposition and entanglement? What is a quantum gate?

3.1. Qubits and their construction

For the average reader without a background in quantum mechanics, but with at least some technical background in computing, a qubit can be thought of as the quantum analog of a classical bit. When we use the term 'classical', we specifically mean non-quantum, as in the type of computing that underpins all modern computers today. A classical bit represents the fundamental unit of information in classical computing and is binary in nature, meaning that a bit exists in either one of two states, usually represented as 0 or 1. In classical physics, it is straightforward to encode a bit's state as either 0 or 1. Usually this is done using voltage levels, where a high voltage might represent a 1 and a low voltage a 0. An important point to state at this point is that a classical bit can only exist in one of these two states at any given time. In classical physics there is no such thing as being both high and low voltage at the same time. It is either one or the other.

Similarly a qubit is also a binary unit of information, but it operates according to the principles of quantum mechanics. There are many quantum mechanical systems that can be used to physically realise a qubit, and generally, any two-level quantum system can technically be used as a qubit. Consider, for example, the spin angular momentum of an electron. This 'spin' can take strictly two values: 'up' or 'down', which can be used to represent the 0 and 1 states of a qubit. Other physical systems that can be used to realise qubits include the polarization states of photons, energy levels of atoms or ions, and superconducting circuits. We are not limited to just electronic spins but a rich variety of physical systems can be used to implement qubits, each with its own advantages and challenges. Such systems include but are not limited to:

- Trapped Ions electronic states
- Superconducting states (charge, flux, or phase)
- Photonic Qubits (polarization states, number of photons, etc)
- Quantum Dots (electron localization states, spin)
- Nuclear Spins (nuclear magnetic resonance spin)

When we consider that the underlying quantum system is being used as just a different way to encode a binary state, it might be tempting to think of a qubit as just a more complex representation of a classical bit. However, the actual power of of quibits lies in their ability to exist in a state called a *quantum superposition*, which is a fundamental principle of quantum mechanics.

3.2. Quantum States, Superposition and Entanglement

In the realm of quantum mechanics we represent the state of a quantum system using a special notation called Bra-ket notation. In this notation, the up state is represented as $|0\rangle$ and the down state as $|1\rangle$. As mentioned previously, however, unlike a classical bit, a qubit can exist in a combination (superposition) of both states simultaneously. For those who are mathematically inclined, this means that a qubit can be in a state that is a linear combination of $|0\rangle$ and $|1\rangle$, represented mathematically as:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{1}$$

where ψ just means 'the state of the qubit'. The coefficients α and β are complex numbers that represent the probability amplitudes of the qubit being in the $|0\rangle$ and $|1\rangle$ states, respectively. This expression for the superposition, given that the complex coefficients are continuous values, show us that the qubit has an uncountably infinite amount of states it can be in. The probabilities of measuring the qubit to be in either state are given by the squares of the magnitudes of these coefficients. In some sense they represent the 'weight' of each state in the superposition, the likelihood that

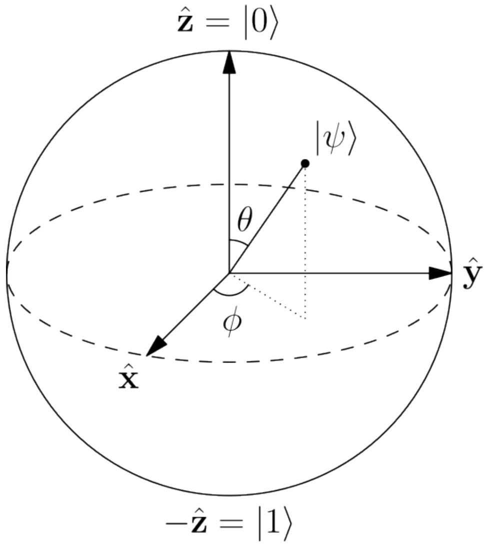

Figure 1: The Bloch Sphere representation of a qubit. Any point on the surface of the sphere represents a valid qubit state, with $|0\rangle$ and $|1\rangle$ at the poles. The angles θ and ϕ determine the specific superposition state.

measuring the qubit will give that particular state, as in quantum mechanics, the act of measurement causes the qubit to 'collapse' into one of the base states (either $|0\rangle$ or $|1\rangle$) with probabilities determined by these coefficients. It is this superposition that enables quantum computers to perform certain computations more efficiently than classical computers as these complex coefficients can interfere constructively or destructively during quantum computations, allowing quantum algorithms to amplify the probabilities of correct answers while suppressing incorrect ones.

The nature of superposition also leads to the concept of *entanglement*, another fundamental property of quantum mechanics. When two qubits become entangled, the state of one qubit becomes directly correlated to the state of the other, regardless of the distance between them. This means that measuring one qubit instantly determines the state of the other qubit. Entanglement is a key resource in many quantum algorithms and protocols, enabling phenomena such as quantum teleportation and superdense coding. It is also a crucial component in achieving quantum advantage, as it allows for correlations that are not possible in classical systems as there is no classical analog to entanglement, it is a purely quantum mechanical phenomenon.

As the coefficients α and β are complex numbers, they have both a magnitue and a phase, and it is this phase that allows for interference effects to occur due to relative phase differences between qubit states. Maintining stable relative phases is critical for the proper functioning of quantum algorithms, and is measured by a property called coherence time, which is the time over which a qubit can maintain its quantum state before loss of information to the environment (decoherence) occurs. The global coherence of a multi-qubit system is particularly important, as quantum algorithms often rely on the interference of amplitudes across multiple qubits to achieve their computational advantage. Two important metrics used to quantify the coherence of qubits are τ_1 and τ_2 times. τ_1, or relaxation time, measures how long a qubit can stay in an excited state before it decays to its ground state due to energy loss to the environment. τ_2, or dephasing time, measures how long a qubit can maintain its phase coherence before interactions with the environment cause it to lose its quantum information. Both τ_1 and τ_2 times are critical for the performance of quantum computers, as they determine how long quantum information can be reliably stored and manipulated.

3.3. Quantum Gates

It's all well and good to be able to create qubits and understand their properties, in order to perform computations we need a way to manipulate these qubits and perform quantum gate operations on them. Quantum gates are the quantum analogs of classical logic gates. While classical logic gates could be interpreted as performing operations that flip or combine bits in their restrictied classical space, quantum gates are represented mathematically by matrices operating on the state (vectors) of the qubits, performing flips, rotations, entanglements and other transformations in the much larger quantum state space. This makes more sense when considering the geometric representation of a qubit on the Bloch sphere (see Figure. 1), where quantum gates can be visualized as rotations of the qubit state vector on this sphere.

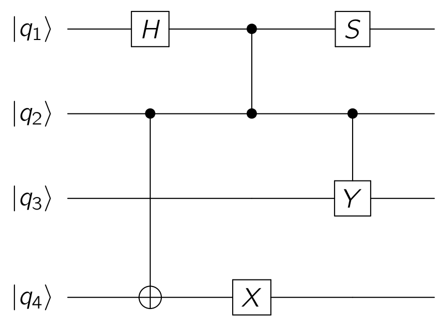

Figure 2: Example of a simple 4-qubit quantum circuit. Each horizontal line represents a qubit, and the symbols show basic operations that can be performed on them such as the Hadamard gate (H), controlled-NOT gate (CNOT), phase gate (S), Pauli-X gate (X), and Pauli-Y gate (Y).

In practice, the experimental implementation of quantum gates depends on the physical system used to realise the qubits. For example, in superconducting qubits, microwave pulses are used to manipulate the energy levels of the qubits, while in trapped ion systems, laser pulses are used to control the internal states of the ions. Common single-qubit gates include the Pauli-X, Y, and Z gates (which correspond to bit-flip and phase-flip operations), as well as the Hadamard gate (which creates superposition). Multi-qubit gates, such as the CNOT (Controlled-NOT) gate, are essential for creating entanglement between qubits and enabling more complex quantum operations. Given that gate operations are actual physical operations, they take a finite duration of time. The time taken to perform a gate operation, known as the gate time τ_g, is important in the context of quantum algorithms, as it must be much shorter than the coherence times (τ_1 and τ_2)(see practical challenges section).

3.4. Decoherence, Errors, and Correction

In an actual quantum computer, we are far from the ideal mathematical situation of qubits being perfectly coherent, noise free, and decoupled from the environment. In practice, the quantum states required are incredibly delicate, generally requiring the system to be supercooled to near absolute zero temperatures, and isolated from all external influences. Even still, it is impossible to remove all sources of noise, and as such, the system is subject to decoherence. Decoherence is the proccess by which a quantum system loses its quantum properties due to interactions with its environment. This can occur through various mechanisms, such as thermal fluctuations, electromagnetic interference, or imperfections in the qubit fabrication process. Decoherence leads to the loss of superposition and entanglement, effectively collapsing the qubit states into classical states and destroying the quantum information they carry. This is not a limitation of hardware alone, but a fundamental property of quantum mechanics itself. As stated earlier, the coherence times τ_1 and τ_2 are critical metrics that quantify how long a qubit can maintain its quantum state before decoherence occurs. As we add more qubits to the system, the global coherence of the multi-qubit becomes exponentially more fragile and susceptible to decoherence, making it increasingly challenging to maintain the quantum information across the entire system. Decoherence ultimately results in the occurrence of errors during quantum computations as the global state collapses unpredictably.

Decoherence is not the only source of error in quantum computations. The quantum gate operations themselves are also subject to imperfect operations and thus have some probability of failure. For example, attempting to perform an X gate on a qubit in the $|0\rangle$ state might not always successfully flip it to the $|1\rangle$ state due to imperfections in the control pulses or interactions with the environment during the gate operation. This is known as gate error. Gate fidelity is a measure of how accurately a quantum gate performs the intended operation, and it is typically quantified using metrics such as the average gate fidelity. As a rule of thumb, gate fidelities above 99% are generally considered good, but for practical quantum computing, even higher fidelities (e.g., 99.9% or better) are often required[**99.9percent-required-fidelity**].

To mitigate the effects of decoherence and gate errors, quantum error correction techniques are employed. Error correction involves encoding the quantum information across multiple physical qubits to create logical qubits that are more robust against errors. This is typically done using quantum error-correcting codes, such as the surface code or the Steane code, which can detect and correct certain types of errors without directly measuring the quantum information itself (which would collapse the state). Implementing error correction requires additional qubits and gate operations, which adds complexity to the quantum computer. The overhead for error correction can be significant, often requiring an order of magnitude more physical qubits to create a single logical qubit. However, this is essential for achieving fault-tolerant quantum computing, where computations can be performed reliably even in the presence of noise and errors.

4. Practical Challenges in Quantum Computing

4.11. Decoherence and Noise

As stated in the previous section, the fragility of quantum states plays a huge role in the accuracy of quantum computations. Qubits interact unavoidably with their environments, leading to decoherence that destroys superpositions and entanglement, which are essential for quantum advantage and reducing the error rates of quantum gates. Minimizing decoherence and noise is one of the biggest challenges in building practical quantum computers. The scaling effect of decoherence and noise is particularly problematic as the number of qubits increases, making it exponentially more difficult to maintain coherence across the entire system as we enter the realm of hundreds to thousands of qubits.

4.2. Gate Fidelity and Speed

Given the coherence windows τ_1 and τ_2, these set a hard limit on how many gate operations can be performed before the quantum information is lost. As a real example in trapped ion systems, which can have exceptionally long coherence times of the order of seconds or longer, single-qubit gate times can be on the order of microseconds, while two-qubit gates can take tens to hundreds of microseconds. For a 50 second coherence time and $500\mu s$ gate time, this allows for only 100,000 gate operations before decoherence occurs and destroys the computation. When considering a practical algorithm like Shors algorithm for integer factorization, this number seems far more than enough when considering the trvial example of factoring the number 15 (4-bits) into its factors of 3 and 5. But entering the regime of RSA encryption where we are dealing with 2048-bit numbers to be factorized, we are looking at the order of trillions of gate operations. Thus, faster gate operations are desirable to maximize the number of operations that can be performed within the coherence time. However, increasing gate speed often comes at the cost of reduced gate fidelity, as faster gates can be more susceptible to control errors and noise. Thus, there a balance must be struck between gate speed and fidelity that must be carefully managed to optimize the performance of quantum computers.

Gate fidelity itself is a critical factor, as even tiny errors in gate operations can accumulate over the course of a quantum computation, leading to incorrect results. Achieving high gate fidelities (e.g., above 99.9%)[**99.9percent-required-fidelity**] is essential for practical quantum computing, but this remains a significant challenge due to various sources of noise and imperfections in the control systems used to manipulate qubits.

3. Error Correction, Fault Tolerance and the Logical-Physical Qubit Gap Overhead problem: Leading schemes like surface codes or color codes require hundreds to thousands of physical qubits per logical qubit. Thresholds: Achieving fault tolerance requires error rates below code-specific thresholds (1 percent for 2D surface codes, lower for others). Connectivity constraints: Error-correcting codes often assume 2D nearest-neighbor layouts, which restricts architecture choices unless long-range couplers are engineered.

4. Qubit Connectivity and Architecture Physical layout: Current technologies (superconducting circuits, trapped ions, neutral atoms, spin qubits) impose specific geometrical constraints on connectivity. Scaling bottlenecks: Long-range entanglement and modular interconnects are in development, but practical quantum networks are immature. Cross-talk: Increasing qubit density introduces electromagnetic or motional interference that couples errors across the system.

5. Control and Calibration Complexity explosion: Each qubit requires individual calibration of frequencies, pulse shapes, and error mitigation parameters. Classical overhead: Classical control electronics (microwave sources, lasers, cryogenics) scale poorly in energy and complexity. Stability: Drift in hardware requires frequent recalibration, reducing uptime and computational efficiency.

6. Resource Requirements for Algorithms Logical vs physical gap: Algorithms like Shor's factoring or HHL require millions of error-corrected qubits, far beyond today's NISQ-era machines (<1,500 physical qubits). Noise-resilient algorithms: Near-term strategies (variational, hybrid quantum–classical methods) face barren-plateau problems and limited provable speedup. Compilation overhead: Translating abstract algorithms into native hardware gates introduces further depth and error accumulation.

7. Engineering and Systems Integration Cryogenics: Superconducting qubits require dilution refrigerators at 10 mK, which struggle to scale to millions of control lines. Laser and vacuum systems: Trapped ions and neutral atoms require stable optical setups that are difficult to miniaturize. Fabrication: Achieving uniform, scalable, and reproducible qubits remains an open challenge, with variability causing large yield losses.

5. Leading Quantum Hardware Platforms

6. Potential Applications of Quantum Computing

The journal of *Nordic Machine Intelligence* uses primarily the Vancouver style of references with numbers in square brackets in the text and a numbered list in the Reference section.[2] However, using the Harvard reference style will also be accepted in some cases. This document is a template for LATEX. Please use the electronic version of this document as a template when you produce your manuscript for submission to the Nordic Machine Intelligence. The paper size is A4 (21 × 29.7 cm). The introduction section of your paper should include the necessary background information, including an adequate review of earlier findings and the justification for conducting this study. 3.1. Qubits and their construction All figures should be numbered consecutively with the figure legend indented 0.5 cm on each side. See figure commented fig for an example. Figures may be in color or black and white and must be of such quality that they produce clear and sharp printouts on an ordinary (color). This document is a template for LATEX. Please use the electronic version of this document as a template when you produce your manuscript for submission to the Nordic Machine Intelligence. The paper size is A4 (21 × 29.7 cm). The introduction section of your paper should include the necessary background information, including an ade- quate review of earlier findings and the justification for conducting this study. 3.1. Qubits and their construction All figures should be numbered consecutively with the figure legend indented 0.5 cm on each side. See figure commented fig for an example. Figures may be in color or black and white and must be of such quality that they produce clear and sharp printouts on an ordinary (color)

7. Future Outlook and Developments

This document is a template for LATEX. Please use the electronic version of this document as a template when you produce your manuscript for submission to the Nordic Machine Intelligence. The paper size is A4 (21 × 29.7 cm). The introduction section of your paper should include the necessary background information, including an ade- quate review of earlier findings and the justification for conducting this study. 3.1. Qubits and their construction All figures should be numbered consecutively with the figure legend indented 0.5 cm on each side. See figure commented fig for an example. Figures may be in color or black and white and must be of such quality that they produce clear and sharp printouts on an ordinary (color). This document is a template for LATEX. Please use the electronic version of this document as a template when you produce your manuscript for submission to the Nordic Machine Intelligence. The paper size is A4 (21 × 29.7 cm).

The introduction section of your paper should include the necessary background information, including an adequate review of earlier findings and the justification for conducting this study. 3.1. Qubits and their construction All figures should be numbered consecutively with the figure legend indented 0.5 cm on each side. See figure commented fig for an example. Figures may be in color or black and white and must be of such quality that they produce clear and sharp printouts on an ordinary (color)

References

1. Monroe C, Meekhof DM, King BE, Itano WM, and Wineland DJ. Demonstration of a Fundamental Quantum Logic Gate. Physical Review Letters 1995; 75. National Institute of Standards and Technology, Boulder, Colorado:4714–7. Available from: https://tf.nist.gov/general/pdf/140.pdf
2. Samples of Formatted References for Authors of Journal Articles. National Library of Medicine. 2018 Apr 26. Available from: http://www.nlm.nih.gov/bsd/uniform_requirements.html [Accessed on: 2020 Sep 27]
3. Halpern SD, Ubel PA, and Caplan AL. Solid-organ transplantation in HIV-infected patients. New England Journal of Medicine 2002; 347:284–7. DOI: 10.1000/182
4. Rose ME, Huerbin MB, Melick J, Marion DW, Palmer AM, Schiding JK, et al. Regulation of interstitial excitatory amino acid concentrations after cortical contusion injury. Brain Research 2002; 935(1-2):40–6. DOI: 10.1000/182
5. Murray PR, Rosenthal KS, Kobayashi GS, and Pfaller MA. Medical microbiology. 4th ed. St. Louis: Mosby, 2002. DOI: 10.1000/182
6. Meltzer PS, Kallioniemi A, and Trent JM. Chromosome alterations in human solid tumors. *The genetic basis of human cancer*. Ed. by Vogelstein B and Kinzler KW. New York: McGraw-Hill, 2002 :93–113. DOI: 10.1000/182
7. Vogelstein B and Kinzler KW, eds. The genetic basis of human cancer. New York: McGraw-Hill, 2002. DOI: 10.1000/182

Year	Qubit Count	Description
1995	2	First experimental demonstration of a quantum logic gate using trapped ions[1]
2001	7	Implementation of Shor's algorithm on a 7-qubit NMR quantum computer[**Vandersypen2001**]
2005	8	Demonstration of an 8-qubit register using trapped ions[**https://www.nature.com/articles/nature0427**
2011	5	D-Wave Systems announced a 512-qubit quantum annealer[**DWave2011**]
2016	16	IBM unveiled a 16-qubit superconducting quantum processor[**IBM2016**]
2017	20	Google announced a 20-qubit superconducting quantum processor[**Google2017**]
2019	53	Google's Sycamore processor achieved quantum supremacy with 53 qubits[**Arute2019**]
2020	65	Honeywell announced a 65-qubit trapped-ion quantum computer[**Honeywell2020**]
2021	127	IBM unveiled a 127-qubit superconducting quantum processor[**IBM2021**]
2022	433	IonQ announced a 433-qubit trapped-ion quantum computer[**IonQ2022**]
2023	1000+	Various companies announced plans for quantum processors with over 1000 qubits[**Various2023**]
2024	2000+	Continued advancements with processors exceeding 2000 qubits[**Various2024**]
2025	5000+	Leading companies project quantum processors with over 5000 qubits[**Various2025**]

Table 1: Key advancements in qubit counts from 1995 to 2025 and their corresponding platforms.