# A Commentary on Quantum Computing and Communications: A Technical Basis

Jean Bacon and Jon Crowcroft

## Introduction

In December 2018 Jon Crowcroft gave a talk "QC for QCs" to our MCCRC project, available at: www.cl.cam.ac.uk/~jac22/talks/qc-for-qc.pdf.

Since this is a new area to most of the computer scientists in the project, as well as the lawyers, Jean Bacon, with Jon's help, wrote a companion paper "Quantum Computing and Communication: A Technical Basis". This is based on finding as much explanatory and tutorial material as possible and asking experts in quantum physics and computing for clarification (to whom, many thanks!). The current version of the paper is at: www.cam.ac.uk/~jmb25/MCCRC, listed as QC-introtech-withMathsandGates-2019-08-12.docx also as a pdf version. The dates in the titles are version numbers that change as updates are made. This commentary also appears on the page as QC-commentary-2019-08-26.docx and as a pdf.

Early versions of the paper attempted to explain Jon's slides without mathematics and quantum circuits. However, it's difficult to reach any level of understanding of the physics that is not superficial and the mathematical modelling helps to explain the assertions being made and capture constraints on system behaviour. It also became clear that to understand the operation of quantum computers it was necessary to introduce simple circuits. After trying the maths as "purple passages" that could be skipped, the current version of the paper has the maths and circuits embedded throughout, but restricts the maths to vectors, matrices and complex numbers. It should be possible to work through with A level maths, plus a little work to understand Dirac's bra-ket notation.

As we read around the subject we found that some important issues and insights were hardly ever highlighted in the material we found. We had to "discover the wood from the trees". These notes are therefore not attempting to be a text-only version of the paper but attempt to summarise these issues which often came as "lightbulb moments" or gradual realisations. In these notes we refer to section numbers in the paper for further reading.

An equation that remains in this commentary is a visualisation of Schrödinger's cat, based on Heisenberg's uncertainty principle. The state $|\Psi>$ of a quantum object (a qubit -- in this case, the cat) has probability ½ of being alive and probability ½ of being dead. When we look at the object (measure the qubit), we find out which.

$|\psi\rangle = \frac{1}{\sqrt{2}}|$
$\rangle + \frac{1}{\sqrt{2}}|$
$\rangle$

## Section 1: Mathematical representation of a quantum object or qubit

The first section sets up how to think about a quantum object or qubit, starting from a classical binary digit. In a classical system, the *physical representation of a bit in a computer* has the value 0 or 1. In a quantum system, the *physical representation of a quantum bit (qubit) is modelled as probabilistic,* resulting from the physical properties of spin, momentum etc. In a classical system, bits do not affect neighbouring bits. The power of quantum systems comes from the entanglement (coupling/interaction) property between qubits.

In a classical system, values can be copied and output throughout a process without affecting those values. Copying is ubiquitous in classical systems for example, to replicate bits for error correction. When transmitting data, *a copy* of a bit is sent, and further copies may be buffered en-route, until it is successfully received. Copies of data are routinely logged for security against failures and for audit purposes. In quantum systems copying the state of a qubit is impossible. Although this was known from the early days of quantum theory, the "no-cloning theorem" was published surprisingly late in 1982. In Section 2.5 two approaches to the proof are sketched using the properties of quantum operations that have been set up.

The paper starts by comparing a two-state quantum system with a classical system which has two stable states 0,1 and controlled transitions between them. In a two-state quantum system, a qubit is often described as *simultaneously* having the value 0 with some probability p and the value 1 with probability (1-p). Measuring a qubit yields 0 or 1, according to these probabilities, after which the previous state is lost (perhaps the wave becomes a particle on being measured). The idea that the probabilities must sum to 1, representing probability 1 as the sum of all possible measurement outcomes, carries through to any number of qubits and permeates the modelling mathematics.

Although a given qubit can only be measured once, after which its state is lost, its representation in terms of probabilities implies that, if instead, we were able to carry out the measurement repeatedly we would get values 0 and 1 in proportion to the probabilities. For example, if the probabilities of 0 and 1 were each ½, half the measurements would yield 0 and half 1 over a large number of measurements. In quantum experiments, this idea is carried through to measuring a long stream of qubits in which the measurements are deemed to be probabilistic. If the counts of 0s and 1s are not as expected from the probabilities over large numbers, this might indicate tampering with the stream.

In quantum physics, a qubit can be modelled as a *vector* with unit length (representing probability 1) on the *Bloch sphere,* which dates from the 1940s and is due to the physicist Felix Bloch (1905 – 1983). The Bloch sphere seems a natural intuition for modelling a qubit but is not used much by modern researchers. It certainly adds complexity when we examine it in detail for analogies with classical systems. Its radius is a complex number and it represents a two-dimensional vector space, both of which have to be explained.

In classical systems, *orthogonal axes* (at right angles and therefore independent of each other) i.e. x,y,z axes, are used to represent any point in 3D space. The representation is the projection of the point on each of the three axes. A point on the x axis has representation (x,0,0) meaning that its projection on the y and z axes is zero, similarly points on the y axis have representation (0,y,0) and z axis (0,0,z). Points on a unit sphere are constrained to have $x^2 + y^2 + z^2 = 1$.

Surprisingly, the maths for the *vectors* of quantum systems shows that, for example, the directions **up** and **down** on the Bloch sphere are orthogonal in this vector space world. Indeed, any line through the centre of the Bloch sphere represents a pair of orthogonal vectors. If these vectors are constrained to be of length 1 so as to be on the unit sphere, they are called *orthonormal*. Also surprisingly, *only two orthonormal vectors* are needed as axes to represent any vector on the unit sphere in this 2D vector space. This is because the vectors are complex therefore have two components each. Any vector on the unit sphere can be represented in terms of (say) **up** and **down**.

Paul Dirac (who lectured in Cambridge in the 1960s) introduced the bra-ket notation to represent vectors in this quantum state-space. The vectors |0> (**up** on the Bloch sphere and $\binom{1}{0}$ as a vector) and |1> (**down** on the Bloch sphere and $\binom{0}{1}$ as a vector) are called the *standard or computational basis*, i.e. |0>, |1> are the standard axes for a single qubit, like x,y,z for a classical bit.

Introducing the maths for the above, any vector $|\Psi>$ on the Bloch sphere i.e. the state of any qubit, can be written as a combination of |0>, |1>: $|\Psi> = \alpha\ |0> + \beta\ |1>$ where $\alpha$ and $\beta$ are complex numbers but the modulus (length), $|\alpha|^2+|\beta|^2=1$ (because the vectors are on the unit sphere). α and β are called the probability amplitudes of the vector. Any attempt to measure the state results in $|0>$ with probability $|\alpha|^2$ and $|1>$ with probability $|\beta|^2$ and the total probability is $|\alpha|^2 + |\beta|^2 = 1$. The equation on the cover page showing Schrödinger's cat, gives an example: $\left|\frac{1}{\sqrt{2}}\right|^2+\left|\frac{1}{\sqrt{2}}\right|^2$ = ½ + ½ = 1. A vector with this property (the probabilities sum to 1) is said to be *normalised*, and this concept generalises to modelling the combined state of any number of qubits. The expression for state $|\Psi> = \alpha\ |0> + \beta\ |1>$ is called a *superposition* of states.

Although n bits in a classical computer can represent $2^n$ possible states (all 0 to all 1), only one of these states is represented at any one time. In classical probabilistic (Bayesian) programming, the logic of the program can keep track of probabilities and the results can be output with associated probabilities. This could possibly be at the level of individual bits but is more likely to be associated with n-bit numbers. The approach does not reduce the number of steps needed for an algorithm to produce a result.

The power of QC is that in some sense, before measurement, n bits can hold the whole range of probabilities of values simultaneously (infinitely many), because each qubit is a linear superposition of |0> and |1>. Each operation on the bits carried out by a quantum computing program can therefore be seen as operating on all the probability patterns at once. The challenge is to be able to manipulate, select, collapse and read the one bit pattern that represents the answer to the problem at hand. Quantum algorithms work to increase the probability of the required answer relative to other possible results. After measuring n qubits, the resulting number of patterns of 0 and 1 is the same as for n classical bits, i.e. $2^n$ patterns.

## Section 2: Fundamental properties of quantum systems and how to model them

This section gives what experts present as the fundamental properties of quantum systems. It also introduces how they model quantum systems mathematically, including how to represent operations on qubits including measurement and transformation.

### Heisenberg's Uncertainty Principle (Section 2.1)

An unknown quantum state cannot be observed (measured) without being disturbed. Heisenberg initially had the properties of position and momentum of a particle in mind and described how any conceivable method of measuring a particle's position would disturb its conjugate property, its momentum, thus destroying its *coherence*. It is therefore impossible to simultaneously observe both properties with certainty.

A consequence of this is that once a quantum object has been measured it stays in the measured state. You have collapsed it to measure it -- it is no longer a complex vector representing a wave, but a binary value 0 or 1 (a particle?).

## Superposition (Section 2.2)

**Phases:** An analogy for *superposition* comes from the wave-like (analogue) properties of qubits in that waves can be added or can cancel each other out, depending on their phases. Quantum algorithms use this addition and cancellation to reinforce the required result both within a single qubit and in multiple qubits. Using the Bloch sphere to understand phase, changing the phase angle of a vector so that it traverses a line of latitude, does not affect the probability associated with the qubit. Changing the phase angle of a vector so that it traverses a line of longitude allows it to take the whole range of probabilities from 0 to 1. The literature on quantum mechanics often mentions that a global phase change has no effect on the value of a qubit. I take this to mean that the whole qubit may be tilted with respect to some external measuring equipment without changing the internal relative phases (latitude and longitude).

**Equal superpositions:** After initialising the qubits to be used in a quantum algorithm to |0>, it is usual to create an *equal superposition,* with equal probability of measuring |0> or |1> (both equal to ½). As mentioned above, an example is shown on the cover page. (Recall that a general expression for the state |Ψ> of a qubit is |Ψ> = α |0> + β |1> where $|\alpha|^2+|\beta|^2 = 1$, because the probabilities of measuring |0> or |1> must sum to 1. An equal superposition is $|\Psi\rangle=\frac{1}{\sqrt{2}}|0\rangle+\frac{1}{\sqrt{2}}|1\rangle$, where $|\alpha|^2+|\beta|^2 = ½ + ½ = 1$.)

Superposition of states and equal superpositions are also defined for multiple qubits. Two qubits can represent 4 states, three qubits 16 equally superposed states, and so on (n qubits represent $2^{2^n}$ equally superposed states). The probability amplitudes with the n terms of equal superpositions are each $\frac{1}{\sqrt{2^n}}$.

For two qubits (n=2), a general expression of state is a linear combinations of the computational bases |00> |01> |10> |11>.

$$|\Psi>=\alpha_0|00>+\alpha_1|01>+\alpha_2|10>+\alpha_3|11> \text{ where } \textstyle\sum_{i=0}^{3}|\alpha_i|^2=1$$

As the values of $\alpha_i$ vary continuously, infinitely many states result.
For an equal superposition $\alpha_i = ½$ since $|\alpha_i|^2 = ½^2 = ¼$ and $\sum_{i=0}^{3}|\alpha_i|^2 = ¼ + ¼ + ¼ + ¼ = 1$

## Entanglement (Section 2.3)

Entanglement is essential to achieving quantum computing and communication, which is not obvious on a first reading of the subjects. It is perhaps the least intuitive property of quantum systems. It is *the entanglement property,* that allows qubits to influence each other, that yields the large state space of quantum computing, with its potential for simultaneously capturing multiple steps of complex algorithms.

Certain pairs (or more generally groups) of qubits can be inextricably interconnected; independent but correlated. There is no suggestion of communication between the objects; the effect is instantaneous. This "spooky action at a distance" (according to Einstein) is called the Einstein, Podolsky, Rosen (EPR) paradox. The paradox is that if this related behaviour were due to communication between the entangled objects it would be faster than the speed of light.

Entanglement was a controversial concept from the start. Physicists have investigated whether the phenomenon could be explained by "hidden local variables", i.e., that the entangled bits somehow have a hidden plan (metadata) on how they will behave under all measurement possibilities. In 1964, John S. Bell produced a theorem to the effect that observed quantum results could not be explained by hidden local variables. Experiments have been carried out since, to validate the theorem as well as to close claimed loopholes in the various experimental setups. To date, experimental results have favoured the quantum (not hidden variable) explanation. The phenomenon is already in practical use in quantum communication.

Section 2.3 spells out the infinitely many states two qubits can be in, most of which are entangled states. The idea is that in non-entangled states, the values of the two bits are separable, for example in

$\frac{1}{\sqrt{2}}|00> + \frac{1}{\sqrt{2}}|01>$, the first bit has to be 0, and it can therefore be factorised as $|0>(\frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>)$.

in $\frac{1}{\sqrt{2}}|01> + \frac{1}{\sqrt{2}}|11>$, the second bit has to be 1, so it can be written $(\frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>)|1>$.

An example of two entangled states is $\frac{1}{\sqrt{2}}|00> \pm \frac{1}{\sqrt{2}}|11>$ where we know the two bits are the same, but only on measuring one do we know its value, and instantly know the value of the other. This expression cannot be factorised into two bits. Another example is $\frac{1}{\sqrt{2}}|01> \pm \frac{1}{\sqrt{2}}|10>$ where the values have to be opposite. These four states are known as the **Bell States** after John S. Bell (1928 – 1990). In the modelling, there are infinitely many possible states as the probability amplitudes vary, constrained by the condition that the sum of the squares of the probability amplitudes' moduli = 1.

So, the potential power of quantum computing comes from exploiting entanglement. This means that the technologies used to fabricate qubits have to be capable of interaction (coupling) with each other, unlike neutrinos, for example. The downside of this need to interact is that the qubits are subject to unwanted interactions; they lose accuracy by existing in an external environment. This is called *decoherence*. For this reason, quantum computers have to be shielded from environmental interactions by being kept at a temperature close to absolute zero.

Another implication of this exposure to interaction is loss of accuracy during operation or communication. Classical data can easily be cleaned to 0 or 1, since a bit must at all stages be precisely one or the other and small changes can be detected and corrected. An example is that a bit stream on a communication channel can be corrected into a square-wave form representing 0s and 1s. Because qubits have a superposed state until measured, it is impossible to tell whether small errors have been introduced during operation, so errors can accumulate. There is therefore an overwhelming need for quantum error correction. Multiple copies of classical bits are used for error correction in classical computing but making copies of a qubit is not an option, due to the no-cloning theorem.

### Measurement and transformation of qubits (Section 2.4)

Measurement has been one of the most controversial and difficult areas of quantum mechanics since the early days and is important for quantum computing. In the two-slit experiment of early quantum physics, a single particle can behave like a wave, as though it passes through *both* slits, because interference patterns are observed on a light-sensitive screen beyond the slits. Interference patterns are caused by reinforcement and cancellation when waves are combined, as we see for classical waveforms. However, if a detector is placed on the path to either or both slots, while allowing the particle to continue, then the interference effects disappear – the act of measurement has collapsed the quantum object into a particle.

Measurement and transformation are the essential operations of quantum computing:

- measurement of qubits to read out the end results of algorithms;

- quantum algorithms use transformations of qubits without measurement.

We first revisit representation/measurement bases which were introduced above as orthogonal axes used to represent superpositions. For background, Schrödinger's equation represents the progression of a quantum system, without taking account of any external environmental interaction. Bases are stable solutions to Schrödinger's equation which are called eigenvectors. A set of orthogonal (independent) eigenvectors can be the bases for representing qubits. For example, |0> and |1> are orthogonal; |0> has no component in the |1> direction and vice versa. A general state |Ψ> = α|0> + β|1> has projections α and β on these bases. Measuring the state results in $|0>$ with probability $|\alpha|^2$ and $|1>$ with probability $|\beta|^2$ where the total probability is $|\alpha|^2 + |\beta|^2 = 1$. Section 2.4 attempts to explain the mathematical modelling of measurement.

Both measurement and transformation are represented by the operation of a matrix on a state vector. For a single qubit these are 2x2 matrices. *Hermitian matrices* are associated with measurement in physical systems since they have real eigenvalues and orthogonal eigenvectors which can act as bases. For measurement operations, the matrix operations that achieve measurement separate out α and β. This is also known as projection since α is the projection of the state on |0> and β is the projection on |1>. The final result is scalar 0 or 1 depending on the probabilities $|\alpha|^2$ and $|\beta|^2$.

If an unknown qubit is measured, the orientation of the measuring apparatus is random relative to the qubit's behavior, such as orbit, spin, polarisation etc. For example, a photon can be polarised in two orientations, rectilinear and diagonal. This is used in quantum communication, as described in Section 3. If the measurement happens to use the correct orientation, the result is a 100% correct 0 or 1, otherwise the result is a random 0 or 1 with an associated probability.

Therefore, when qubits are used in quantum computation, they are initialised to |0> according to a known orientation so that they can be operated on and eventually be measured. This can be achieved by measuring some qubit, then using the resulting |0> directly or inverting the resulting |1> to become |0> using a quantum NOT gate. The orientation and axis of measurement are then maintained through all the transformations comprising the quantum algorithm. Only by such means can a meaningful answer be arrived at.

### Transformation

A transformation must conserve certain properties of the object and move it between viable states.
In quantum physics:

- The allowed evolution of quantum systems must ensure that the sum of probabilities of all possible outcomes of any event always equals 1. In linear algebra terms, the transformation conserves inner products and so lengths and probabilities.
- All operations must be reversible, i.e., applying an operator twice brings the system back to where it was. This means that no information can be lost during a transformation, and that the number of inputs to a quantum gate is equal to the number of outputs (unlike many classical gates such as AND, OR, XOR).
- All operators are linear, that is, they can be modelled as operating on the components of a state $\alpha|0>$ then $\beta|1>$ in turn.

Any transformation operator that has these properties is represented by a *unitary matrix*. A little more of the maths of unitary operators is given in the paper and summarised in its appendix, repeated here as an appendix for reference.

Three famous unitary operators (matrices) that achieve unitary transformations for a single qubit are the **Pauli** operators, due to Wolfgang Pauli (1900 – 1958).

$$\mathrm{X} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \mathrm{Y}= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

Each of these has a corresponding quantum gate. Note that the identity operator $\mathrm{I} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ is also unitary, and that $\mathrm{X\,X} = \mathrm{I}$, $\mathrm{Y\,Y} = \mathrm{I}$ and $\mathrm{Z\,Z} = \mathrm{I}$.

## Nonclonability (Section 2.5)

It follows from the properties of unitarity and linearity that a quantum state $|\Psi> = \alpha|0> + \beta|1>$ cannot accurately be copied (cloned). Although this was known from the early days, a mathematical proof was not published until 1982. Section 2.5 sketches how either linearity or unitarity can be used to prove the nonclonability theorem.

As mentioned, a big problem for quantum computers is decoherence and the general build-up of errors. Error correction technology usually relies on taking the majority value of multiple copies, e.g., three or five copies. But the no-cloning theorem means that copies can't be taken. Again, entanglement comes to the rescue. Entangled groups of qubits can be created where all the qubits should have the same value. They can be used to test whether any qubit has diverged from the majority value, allowing error detection and correction.

## Quantum gates and circuits (Section 4.2)

Quantum logic gates are the building blocks for quantum circuits in most quantum computers. (The DWave computer instead uses a process called quantum annealing). Quantum gates can operate on one, two or three qubits. The number of qubits in the input and output of a gate must be equal because of the no-loss, reversibility requirement.

Mathematically, a quantum gate performs a reversible, norm/length/probability-preserving transformation on the qubits. Quantum gates are therefore represented by unitary operators i.e. unitary matrices operating on the vectors that represent qubits. For a single qubit, unitary transformations correspond to rotations of the qubit (unit) vector on the Bloch sphere to specific superpositions. Quantum algorithms are designed to exploit possible cancellations and reinforcements as vectors are rotated.

Section 4.2 introduces a notation for describing quantum gates and circuits and presents them together with the corresponding matrix operators. For example, the quantum NOT gate is implemented by the Pauli X operator. This is equivalent to an exclusive OR (XOR) operation. Like all unitary and therefore linear operators, it can be applied to the terms of a general quantum state in turn, i.e. to $\alpha\ |0>$ and then $\beta\ |1>$.

As introduced above, when discussing superposition, a quantum computation usually starts by clearing a number of qubits to |0> then operating on them to create in each an equal superposition of states, $\frac{1}{\sqrt{2}}|0> + \frac{1}{\sqrt{2}}|1>$. The operator (and gate) that achieves this is called the **Hadamard (H)** operator. Its matrix representation is $\mathrm{H} = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$, which must be unitary of course (checking, $\mathrm{H\,H} = \mathrm{I}$).

**Entangled states:** An important operator is the controlled NOT or C-NOT gate which has two inputs and two outputs. If the first input (called the control) is |0>, the output leaves the two bits unchanged. If the first input is |1> the first output is equal to the first input and the second output is flipped. The second input is called the target. The importance of **C-NOT** is that it *can create entangled states* and, being reversible (as are all quantum gates), can separate out entangled states into the states that were originally entangled (see Sections 4.2.6 and 4.2.7). This separation process is called Bell State Measurement (BSM).

### Quantum algorithms (Section 5)

In 1994 Peter Shor published a quantum algorithm for factorizing prime numbers while at Bell Laboratories. (He became a professor of applied mathematics at MIT in 2003.) Much of existing cryptography relies on the inability of classical computers to factorise large prime numbers. The threat that current cryptography (PKI) will be broken by quantum computation's prime factorisation algorithm has led to the funding of many quantum computing projects worldwide.

Section 5.3 presents some of the work of Deutsch and Jozsa . The idea is to give an example of how a quantum computer might, in one operation, achieve a result that would take a classical computer many, in this case n/2 +1 operations, for n inputs and n outputs. The example is somewhat artificial but illustrates the point for a binary function, i.e., a function that takes a series of 0 or 1 as input and for each input yields 0 or 1 as output. We are told that the output is either constant (either entirely 1 or entirely 0, whatever the input) or balanced (with an equal number of 1 and 0, e.g., a balanced function might output whether each of a sequence of fixed length binary numbers was odd or even). Section 5.3 tabulates the possible functions, shows the maths and gives the circuit that achieves the result.

Section 5.4 shows Grover's algorithm for searching a list to find whether some item is present. The full analysis is not given. Section 5.5 introduces Shor's algorithm for prime factorisation but does not attempt to explain quantum Fourier transform algorithms or give the circuits to implement them. The paper would have doubled in size to explain these algorithms fully.

### Quantum computers are not general-purpose, stored program machines.

What is emerging from the description of quantum gates and algorithms is that quantum circuits are designed to implement quantum algorithms. The quantum computers envisaged so far are not like classical general-purpose, stored-program computers. An outline of how classical computers operate is given in Section 1 of the paper. Quantum circuits are built to achieve specific algorithms, similar to the wartime Colossus computer at Bletchley Park.

### Quantum communication (Sections 3, 5.1 and 5.2)

Another consequence of the threat that current cryptography could be broken by quantum prime factorisation is the development of *post-quantum cryptography;* cryptographic algorithms that don't rely on prime factorisation or other properties that are vulnerable to quantum algorithms. **Quantum key distribution (QKD)** can be seen as part of post-quantum cryptography, being a method of transmitting a secret key of any length to two communicating parties.

Section 3 describes quantum key distribution which, to detect and prevent eavesdropping, depends on the no-cloning theorem and the fact that quantum measurement destroys quantum state. Current implementations depend on entanglement. Communicating parties A and B are each sent a stream of photons. Each photon sent to A has its entangled pair sent to B.

Quantum communication has been demonstrated in the research domain and is moving to the implementation phase. As large networks are developed, it will be necessary to perform a function similar to repeaters in traditional networks, in which classical bits are cleaned and copied. Section 5.1 describes **quantum teleportation** whereby a qubit's state can be transferred (not copied) from A to a repeater, then from the repeater to B. The method depends on entanglement. **Superdense coding** is described in Section 5.2. Here, two measured bits can be transferred from A to B by means of a single entangled qubit transfer. Again, quantum entanglement is at the heart of the process.

Quantum communication is being shown to be practically viable. The properties of photons are such that the devices that manipulate them do not need to be kept at very low temperatures. Devices that polarise and measure the polarisation of photons are becoming routinely deployed.

## Projects, challenges and conclusions (Sections 6, 7, 8)

An overview of fabrication technologies and current quantum computing projects is covered in a related paper. A discussion of challenges for quantum computing to become feasible concludes this paper.

## Appendix: Some definitions from linear algebra

### Norm of a vector

The norm (length) of a vector |a> $=\sqrt{< a|a >}$

### Inner product of two vectors

The inner product of two vectors |a> and |b> = <a|b>

$|a> = \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix}$, $<a| = (\alpha_1{*}, \alpha_2{*})$ and $|b> = \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix}$, $<b| = (\beta_1{*}, \beta_2{*})$

$<a|a> = (\alpha_1{*}, \alpha_2{*}) \begin{pmatrix} \alpha_1 \\ \alpha_2 \end{pmatrix} = |\alpha_1|^2 + |\alpha_2|^2$ and $<a|b> = (\alpha_1{*}, \alpha_2{*}) \begin{pmatrix} \beta_1 \\ \beta_2 \end{pmatrix} = (\alpha_1{*}\, \beta_1 + \alpha_2{*}\, \beta_2)$.

### Commutative matrices

Two square matrices A, B commute if $AB = BA$

### Inverse of a matrix

The inverse $A^{-1}$ of a square matrix A is such that $AA^{-1}=I$

### Transpose of a matrix

The transpose of a matrix A is obtained by exchanging rows and columns, denoted $A^T$

### Conjugate transpose of a matrix

The conjugate transpose of a complex matrix A is obtained by taking the complex conjugate of each element and exchanging rows and columns, denoted $A^\dagger$ (sometimes $A^*$).

Note that if A is real then $A^\dagger = A^T$

### Normal matrix

A complex square matrix A is normal if it commutes with its conjugate transpose, $AA^\dagger = A^\dagger A$

**An eigenvector** v of a linear transformation A is a non-zero vector that changes by only a scalar factor when that linear transformation is applied to it, i.e. Av = cv, where (scalar) c is the **eigenvalue** associated with the eigenvector v.

### Hermitian matrix

A complex matrix A is Hermitian if $A^\dagger = A$

If A is real, $A^\dagger = A^T$. Note that $A^T = A$ is just the definition of a symmetric real matrix.

Hermitian matrices have real eigenvalues and their eigenvectors are always orthogonal for different eigenvalues, so they form a basis for the whole space.

### Unitary matrix

A complex matrix U is unitary if $UU^\dagger = U^\dagger U = I$, i.e. $U^\dagger = U^{-1}$ and $U^\dagger$ is also unitary.

The importance of unitary matrices in quantum mechanics is that they preserve norms, and thus probability amplitudes. U is a normal matrix with eigenvalues lying on the unit circle.

Unitary matrices are Hermitian and are those matrices with a complete set of orthonormal eigenvectors such that the corresponding eigenvalues are ±1.

The rows (and columns) of a Unitary matrix form a unitary basis, that is, each row (or column) is of length 1.

Given two complex vectors a and b, multiplication by U preserves their inner product:

$<Ua|Ub> = <a|U^\dagger U|b> = <a|I|b> = <a|b>$