Modern Computation

A Unified Approach

S. S. Chandra
Ph.D
June 2019

Draft
Version: 0.5

“Computing Science is no more about computers
than Astronomy is about telescopes.”

Edsger W. Dijkstra (1930-2002)

CHAPTER 6

Quantum Computation

> "Until computers and robots make quantum advances, they basically remain adding machines: capable only of doing things in which all the variables are controlled and predictable."
>
> Michio Kaku (1947-)

We have seen from previous chapters that the notion of computation can be reduced to the encoding of symbols. Computation is then the evolution of the symbols to another set of symbols in a definite, predictable and prescribed way. For example in Turing machines, we maintain a series of fixed states and a tape of symbols to do the computation. In these previous computation models, we are not concerned with how long it takes and that we have infinite time, instead we are concerned with whether computations are possible or not and whether the computation will complete or halt.

In an ideal computational system however, we would like to compute as much as possible within the shortest time possible. If the computation has multiple branches, we would like to explore all branches, since we are likely not to know which is the correct one. We could do many computations at once if we run multiple sub-computations to speed things up, i.e. compute lots of different chunks of the problem in parallel. This might be a kin to using multiple heads in a Turing machine and, as we saw in previous chapters, this does not make our computer more powerful in terms of computation capability. But what if we could compute all possible branches and chunks all at once and in a single computation? Is such a system even possible? Even if it is possible, how would you even design and build such a computer? In this chapter, we will show that such as computer is indeed possible and discuss the theory behind them. More details about quantum computation can also be found in Moore and Mertens [2011, Chapter 15].

6.1 Quantum Mechanics

The development of quantum mechanics (QM) comes from the simple goal of trying to understand the fundamental constituents of matter and being able to make predictions of physical

systems in nature. Traditionally, we have understood all natural phenomenon before the 20th century in one of two main ways: waves or particles.

6.1.1 Wave or Particle?

We experience sound and ripples in ponds and lakes as waves, where vibrations in material or fluids are carried across space and time as pressure or density changes. We know these wave properties so well, we use them intuitively to create music with different instruments that utilise vibrations via many different mechanisms such as strings, membranes and cavities. The music we create are made up of specific set of waves that form the fundamental building blocks of all music called harmonics. These harmonics are the simplest set of waves that are possible on a string tied at both ends and are called standing waves as a result as they resonate and continue to "ring" out as you pluck the string. Trying to create waves of other types results in sound that simply dies out too fast to be heard or just sound unpleasant. Interestingly, these harmonics only exist when the total number of maximum displacements from the resting position (so called toughs) in the wave is a whole number (see Figure 6.1).

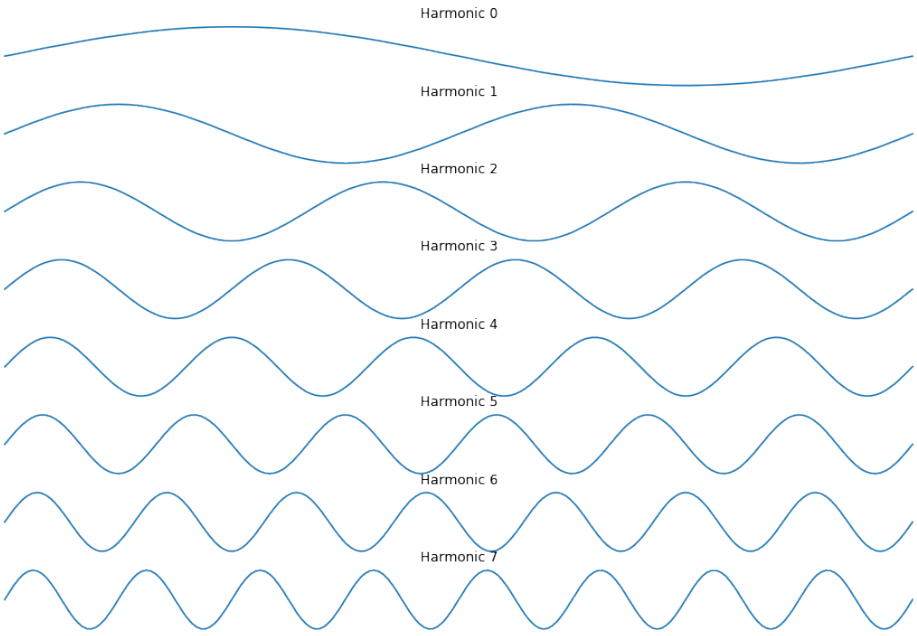

Figure 6.1: The standing waves that make the harmonics of a musical system.

Waves also exhibit another important property called superposition. Superposition is the process of creating a linear combination or weighted sum of elements, so that a superposition of a set of harmonics is a weighted sum of those harmonics. For example, the superposition $\mathbf{r}$ of two vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ can be written as $\mathbf{r} = \boldsymbol{a}\,\mathbf{e}_1 + \boldsymbol{b}\,\mathbf{e}_1$. In fact, this is how we write vectors in 2D to represent position, where the vectors $\mathbf{e}_1$ and $\mathbf{e}_2$ are the basis vectors for the $\boldsymbol{x}$ and $\boldsymbol{y}$ dimensions.

We also observe the particle or rigid body interaction in nature by way of collisions of these bodies, such as billiard balls and objects falling in gravity. These particles or objects exhibit a change in motion via exchanges of momentum $\boldsymbol{p} = \boldsymbol{mv}$, where $\boldsymbol{m}$ is the mass of the particle and $\boldsymbol{v}$ it's velocity. We also imagine planets around our sun and other celestial objects with the same ideas.

Therefore based on these experiences, we ask the obvious questions: Does all matter exist as waves or as particles at the fundamental level? What experiments confirm either hypothesis? The answer to those questions are that matter exists as neither of them (or in a form that is effectively both of them at once, depending on your point of view) and experiments confirm that matter can have both properties! This relatively new form leads to unexpected behaviour that we have no intuition for, which is usually what makes QM a challenging theory to learn.

6.1.2 Wavefunctions

All fundamental matter in the universe exists in a quantum state or a superposition of quantum states that can be described by a wavefunction ψ. A wavefunction can be thought of as a wave packet, a bundle of harmonics weighted and summed together to form an envelope of localised energy. Figure 6.2 shows an example of two wavefunctions interacting.

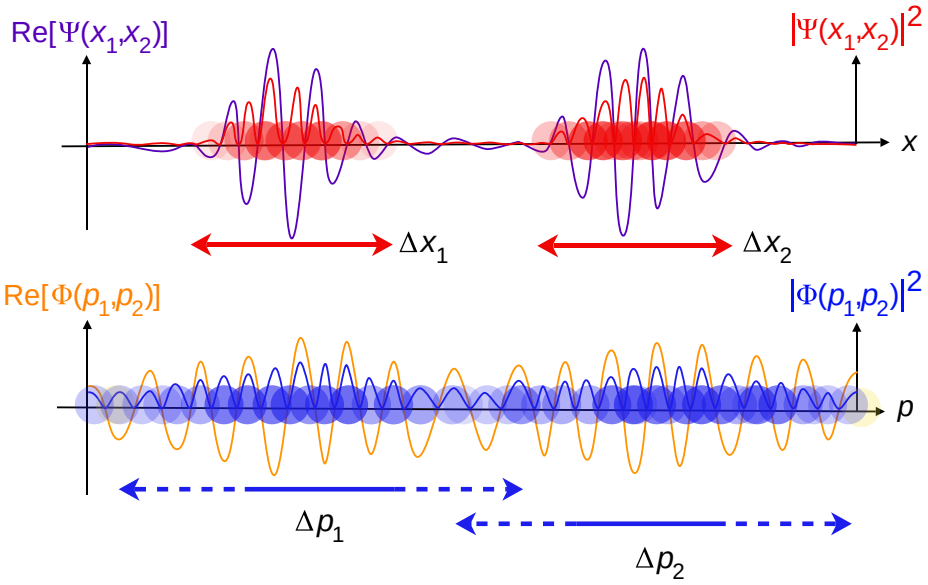

Figure 6.2: An example of the wavefunctions ψ_1 and ψ_2 as wave packets while interacting and their corresponding momenta. Original figure by Maschen.

The wavefunction model of matter explains how matter can behave as a particle or as a wave. Consider the set of waves that are superimposed to create the wavefunction. Since the constituents are waves and will exhibit wave-like properties, the summation will also. But also consider the spatial extent $\Delta\boldsymbol{x}$ of the wavefunction, it can be seen to effectively represent a particle of finite size. Indeed the length of the wavefunction

$$\boldsymbol{P}(\boldsymbol{x}) = |\psi|^2 \tag{6.1}$$

represents the probability of the particle location $\boldsymbol{P}(\boldsymbol{x})$, that is the chances of finding the particle at that location upon measurement. This is often referred to as the probability density as it relates to how 'tightly packed' the packet is with respect to space (and time if the wavefunction happens to be time dependent as well). For example, one of the simplest wavefunctions are the Gaussian wave packets, whose probability density is the normal distribution. It effectively says that the particle will be found in space according to the normal distribution when a measurement is made.

The interesting concept here is the 'measurement' being made. In the macroscopic scale, making a measurement has inconsequential effect on the outcome of the measurement itself. Measuring the length of a table is hardly affected by the act of the measurement. However, at the quantum scale (typically less than an Angstrom, which is 10^{-10}m), the act of using light to 'see' the electron can cause the electron to change state or even be ejected from the matter being observed.

In QM, there are one of two consequences of making a measurement. The accepted consequence until recently has been that the wavefunction collapses into one of the outcomes that is possible according to the probability density. The collapse is the outright destruction of the quantum state to produce this outcome. This leads to a famous paradox called the Schrödinger's cat.

Imagine a cat inside an opaque box. Inside the box is a radioactive substance, whose decay is known to be a quantum process that emits an alpha particle (via quantum tunnelling, a concept that will be discussed later) according to its probability density. A detector is present on the far side of the box that effectively triggers the poisoning of the cat if an alpha particle is measured by the detector. How does one know if the cat is alive or dead without opening the box? In essence, the cat is in a superposition of states, simultaneously alive and dead until a measurement is made. Opening the box will trigger the collapse of the system into one of the possible states, i.e. the cat will be found dead or alive. This is often the conundrum that QM systems create when trying to conduct experiments and quantum computation will be no exception.

6.1.3 Uncertainty Principle

We can however note another weird phenomenon of the wavefunction representation of matter. Observe what happens when we add more and more frequencies to the wave packet. It will become more and more localised having a smaller and smaller $\Delta\boldsymbol{x}$. Note that the energy of the wavefunction is directly proportionate to the different frequencies present. Using the wave nature of sound and the harmonics, having higher and higher frequencies, we must induce more and more energy into the string in order for it to vibrate faster. Thus, as we add more frequencies, it increases the $\Delta\boldsymbol{E}$, which we will show later as being equivalent to an increase in momenta $\Delta\boldsymbol{p}$. Conversely, if we only use a single frequency, i.e. use a single sine curve to represent a wavefunction, we know the energy exactly as there is only a single frequency, but as it is well known that a sine curve propagates to infinity in both the positive and negative $\boldsymbol{x}$ directions, so that it is as poorly localised as possible. This give and take nature of QM is

known as Heisenberg's uncertainty principle and it broadly states that

$$\Delta \boldsymbol{x} \Delta \boldsymbol{p} \geqslant \frac{\hbar}{2} \tag{6.2}$$

or equivalently

$$\Delta \boldsymbol{E} \Delta \boldsymbol{t} \geqslant \frac{\hbar}{2} \tag{6.3}$$

Here $\hbar = {}^{\boldsymbol{h}}/_{2\pi}$, pronounced 'h bar' and $\boldsymbol{h} = 6.62610^{-34}$ m^2 Kg/s and is known as Planck's constant. It is this constant that is the very definition of QM, appearing in nearly all expressions describing quantum phenomenon found in nature by QM.

6.1.4 Planck's Constant

At the beginning of the 20th century, a new phenomena was observed call the photoelectic effect. Classical physics predicted that shining light on the metals would induce a current proportional to the intensity of the light if light was modelled as a wave phenomena. The more the light incident on the metal, the more number of electrons would be excited and larger the current.

It turned out that the current was experimentally found and verified to be dependent on the frequency of the light and not the intensity. No matter how bright the light, current did not flow unless there was a minimum frequency of the light, a frequency dependent on the metal involved. Eventually it was found that the energy of the electron $\boldsymbol{E}$ given in electron volts (eV) was proportional to the incident light frequency $\boldsymbol{f}$ as

$$\boldsymbol{E} = \boldsymbol{hf} \tag{6.4}$$

where $\boldsymbol{h}$ is the same Planck's constant. In fact, these experiments of this photo-electric effect gave the first estimates of the constant and represented the slope of the line plotting the $\boldsymbol{E}$ and frequency $\boldsymbol{f}$, which was always the same regardless of the type of metal.

Einstein [1905] pointed out that this could only occur if light behaved as a particle having a 'quanta' or prescribed amount of energy that is transferred to the electron allowing current to flow. It would be twenty years later until Schrödinger [1926] would show that both wave and particle natures of light could be explained using wavefunctions and use it to predict the energy levels of Hydrogen. Throughout QM, one will see Planck's constant as a recurring theme that captures the quantum nature of matter. Schrödinger [1926] work would eventually be generalised to produce a more elegant mathematical framework, known as Dirac's notation, to explain the remaining quantum phenomena including to construct quantum computing.

6.2 Dirac's Notation

We have discussed how all matter is in a superposition of quantum states ψ or most precisely $\psi(\boldsymbol{x}, \boldsymbol{t})$, since the wavefunction is a function of space and time. For what follows, we will ignore the time aspect of the wavefunction, so that it is $\psi(\boldsymbol{x})$. In general, we will represent the state $\psi(\boldsymbol{x})$ has a complex-valued vector and operations between vectors will include inner products,

projections and outer products. For example, the inner product of two states ψ_1 and ψ_2, i.e. the projection of one state onto another, would be

$$\int_{-\infty}^{\infty} \psi_1(\boldsymbol{x})^* \psi_2(\boldsymbol{x}) \boldsymbol{dx} \tag{6.5}$$

as in general ψ_1 and ψ_2 are complex valued and $*$ represents complex conjugation. Note that complex conjugation is the operation of negation of all imaginary parts to allow complex numbers to be projected back into the real number line. Since the norm of the quantum state is the probability density, we also write

$$\int_{-\infty}^{\infty} \psi(\boldsymbol{x})^* \psi(\boldsymbol{x}) \boldsymbol{dx} = \int_{-\infty}^{\infty} |\psi(\boldsymbol{x})|^2 \boldsymbol{dx} = 1 \tag{6.6}$$

The above notation can get very complicated when dealing with a large number of states and many operations. It is thus desirable to give these states a special name when designing a compact mathematical notation for them.

Dirac [1930] designed just such a notation that we will utilise in this chapter to discuss quantum states in general. If we define a state vector $\boldsymbol{a}$, then the ***ket*** vector that represents $\boldsymbol{a}$ is denoted as $|\boldsymbol{a}\rangle$ and its corresponding ***bra*** that represents $\boldsymbol{a}^*$ is denoted as $\langle \boldsymbol{a}|$. Then we can represent superposition of state vectors $|\boldsymbol{a}\rangle$ and $|\boldsymbol{b}\rangle$ to give another state vector $|\boldsymbol{c}\rangle$ as

$$|\boldsymbol{c}\rangle = \alpha_1 |\boldsymbol{a}\rangle + \alpha_2 |\boldsymbol{b}\rangle \tag{6.7}$$

for arbitrary complex numbers α_1 and α_2. Thus, the inner product reduces to the simple "braket" statement

$$\langle \boldsymbol{a}|\boldsymbol{b}\rangle := \int_{-\infty}^{\infty} \boldsymbol{a}(\boldsymbol{x})^* \boldsymbol{b}(\boldsymbol{x}) \boldsymbol{dx} \tag{6.8}$$

and that

$$\langle \boldsymbol{a}|\boldsymbol{a}\rangle = 1 \tag{6.9}$$

In linear algebra, we can write any vector space via a set of basis vectors $\{\boldsymbol{e_i}\}$, where $\boldsymbol{i} = 1, \ldots, \boldsymbol{n}$ and $\boldsymbol{n}$ is the dimension of the space. The basis vectors are generally of length unity and orthogonal to each other. Orthogonality means the vectors are linearly independent and thus have no common components. In other words, the basis vectors satisfy the conditions that the norm is unity $\langle \boldsymbol{e_i}|\boldsymbol{e_i}\rangle = 1$ and the inner (or dot) product $\langle \boldsymbol{e_i}|\boldsymbol{e_j}\rangle = \delta_{\boldsymbol{ij}}$, where $\delta_{\boldsymbol{ij}}$ is the Dirac delta function defined as

$$\delta_{\boldsymbol{ij}} = \begin{cases} 0, & \boldsymbol{i} \neq \boldsymbol{j} \\ 1, & \boldsymbol{i} = \boldsymbol{j}. \end{cases} \tag{6.10}$$

This function simply states that the only vector that has a common component to a vector like $\boldsymbol{e_i}$ is the vector $\boldsymbol{e_i}$ itself. This set of basis vectors is often also referred to as an orthonormal set. We can simply write basis vectors compactly using direct notation for $|\boldsymbol{e_i}\rangle$ as $|\boldsymbol{i}\rangle$ so that

$$\langle \boldsymbol{i}|\boldsymbol{j}\rangle = \delta_{\boldsymbol{ij}} \tag{6.11}$$

When we make measurements of physical systems, the wavefunctions of these systems are sampled discretely, so that our state vectors are discrete also, so that we can use linear algebra

to describe them. The state vectors can then be defined in terms of basis vectors as a discrete superposition. For example, the two state vectors

$$|\boldsymbol{a}\rangle = \sum_{\boldsymbol{i}=0}^{\boldsymbol{n}} \alpha_{\boldsymbol{i}} |\boldsymbol{e_i}\rangle , \qquad |\boldsymbol{b}\rangle = \sum_{\boldsymbol{i}=0}^{\boldsymbol{n}} \beta_{\boldsymbol{i}} |\boldsymbol{e_i}\rangle \tag{6.12}$$

and likewise

$$\langle \boldsymbol{a}| = \sum_{\boldsymbol{i}=0}^{\boldsymbol{n}} \alpha_{\boldsymbol{i}}^* \langle \boldsymbol{e_i}| , \qquad \langle \boldsymbol{b}| = \sum_{\boldsymbol{i}=0}^{\boldsymbol{n}} \beta_{\boldsymbol{i}}^* \langle \boldsymbol{e_i}| \tag{6.13}$$

The resulting ***kets*** are column vectors and ***bras*** are row vectors

$$\langle \boldsymbol{a}| = (\alpha_1^*, \ldots, \alpha_{\boldsymbol{n}}^*) , \qquad |\boldsymbol{a}\rangle = \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{\boldsymbol{n}} \end{pmatrix} \tag{6.14}$$

This changes our inner product definitions to discrete ones

$$\begin{aligned} \langle \boldsymbol{a}|\boldsymbol{b}\rangle &= \sum_{\boldsymbol{i}=0,\boldsymbol{j}=0}^{\boldsymbol{n}} \alpha_{\boldsymbol{i}}^* \beta_{\boldsymbol{j}} \langle \boldsymbol{e_i}|\boldsymbol{e_j}\rangle \\ &= \sum_{\boldsymbol{i}=0}^{\boldsymbol{n}} \alpha_{\boldsymbol{i}}^* \beta_{\boldsymbol{i}} \end{aligned} \tag{6.15}$$

This also means that all the operations involving the state vectors are vector and matrix multiplications. Thus, changing states from one state to another is done by the use of operators

$$\boldsymbol{\Omega}\, |\boldsymbol{a}\rangle = |\boldsymbol{b}\rangle \tag{6.16}$$

and the operators are matrices that define the act of measurement or computation. We can see this from the equation (6.16) and (6.12) that

$$\sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \boldsymbol{\Omega}\, \alpha_{\boldsymbol{j}} |\boldsymbol{j}\rangle = \sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \beta_{\boldsymbol{j}} |\boldsymbol{j}\rangle \tag{6.17}$$

We contract the ***kets*** by multiplying with a ***bra*** $\langle \boldsymbol{i}|$ both sides resulting in

$$\sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \langle \boldsymbol{i}|\, \boldsymbol{\Omega}\, \alpha_{\boldsymbol{j}} |\boldsymbol{j}\rangle = \sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \langle \boldsymbol{i}|\, \beta_{\boldsymbol{j}} |\boldsymbol{j}\rangle \tag{6.18}$$

which can be further simplified by taking out the scalars

$$\sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \langle \boldsymbol{i}|\, \boldsymbol{\Omega}\, |\boldsymbol{j}\rangle\, \alpha_{\boldsymbol{j}} = \sum_{\boldsymbol{j}=0}^{\boldsymbol{n}} \langle \boldsymbol{i}|\boldsymbol{j}\rangle\, \beta_{\boldsymbol{j}} \tag{6.19}$$

We can now define the matrix elements

$$\boldsymbol{\Omega} := \Omega_{\boldsymbol{ij}} := \langle \boldsymbol{i}|\, \boldsymbol{\Omega}\, |\boldsymbol{j}\rangle \tag{6.20}$$

allowing us to finally write

$$\sum_{j=0}^{n} \Omega_{ij}\alpha_j = \beta_i \tag{6.21}$$

where summation over the same indices j result in a contraction from a matrix to a vector, leaving the vector with index i.

Finally, an important operation in linear algebra is the outer product or tensor product, which can be represented in Dirac notation as

$$P = |a\rangle \langle b| = a \otimes b \tag{6.22}$$

The result is a matrix P which becomes a projection operator with components $|a\rangle$ and $\langle b|$. In quantum computing, it is often more convenient to represent the outer product of k n-dimensional vectors as a n^k-dimensional vector instead to represent a superposition. Later we shall use this property to create k-bit systems, whose states are 2^k-dimensional vectors.

6.3 Qubits

We will begin by formulating classical or conventional computing using a operator like approach via linear algebra.

6.3.1 Classical Bits

Consider a simple single bit system that consists of just two states 0 and 1, we can write this system as

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \qquad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{6.23}$$

We can view this notation as representing a vertical switch with states $|0\rangle$ and $|1\rangle$ representing the on and off positions of that switch. We can operate on these states by using operators or gates represented as matrices and their application to states as matrix multiplication. For a 1-bit system there are only four operations possible:

1. Identity

$$\begin{aligned} I|0\rangle &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= |0\rangle \end{aligned} \tag{6.24}$$

$$\begin{aligned} I|1\rangle &= \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.25}$$

2. Negation

$$\begin{aligned} \boldsymbol{N}|0\rangle &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.26}$$

$$\begin{aligned} \boldsymbol{N}|1\rangle &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= |0\rangle \end{aligned} \tag{6.27}$$

3. Set State to $|0\rangle$

$$\begin{aligned} 0\,|0\rangle &= \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= |0\rangle \end{aligned} \tag{6.28}$$

$$\begin{aligned} 0\,|1\rangle &= \begin{pmatrix} 1 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \\ &= |0\rangle \end{aligned} \tag{6.29}$$

4. Set State to $|1\rangle$

$$\begin{aligned} 1\,|0\rangle &= \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.30}$$

$$\begin{aligned} 1\,|1\rangle &= \begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.31}$$

Now consider a 2-bit system, so that we have the four possible states $|00\rangle$, $|10\rangle$, $|01\rangle$ and

$|11\rangle$. Using our notation, we can write them as

$$|00\rangle = |0\rangle\langle 0| = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \tag{6.32}$$

$$|01\rangle = |0\rangle\langle 1| = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{6.33}$$

$$|10\rangle = |1\rangle\langle 0| = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \tag{6.34}$$

$$|11\rangle = |1\rangle\langle 1| = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} \tag{6.35}$$

$$|v\rangle \otimes |w\rangle = \begin{pmatrix} \alpha \\ \beta \end{pmatrix} \otimes \begin{pmatrix} \delta \\ \gamma \end{pmatrix} = \begin{pmatrix} \alpha\delta \\ \alpha\gamma \\ \beta\delta \\ \beta\gamma \end{pmatrix} \tag{6.36}$$

where $|\boldsymbol{a}\rangle\langle\boldsymbol{b}|$ is the outer product (also known as a tensor product $\boldsymbol{a} \otimes \boldsymbol{b}$) of the states $\boldsymbol{a}$ and $\boldsymbol{b}$ discussed in section 6.2. In addition to our operators for the single bit system, we introduce another operator called the conditional NOT or CNOT

$$\boldsymbol{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{6.37}$$

that only flips or applies NOT to one of the bits when one of the chosen bits is in the $|1\rangle$ state.

We can then apply this to our two bit system states to obtain

$$\begin{aligned}
\boldsymbol{C}|00\rangle &= \boldsymbol{C}|0\rangle\langle 0| \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} \\
&= |00\rangle \qquad (6.38) \\
\boldsymbol{C}|01\rangle &= \boldsymbol{C}|0\rangle\langle 1| \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} \\
&= |01\rangle \qquad (6.39) \\
\boldsymbol{C}|10\rangle &= \boldsymbol{C}|1\rangle\langle 0| \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \\
&= |11\rangle \qquad (6.40) \\
\boldsymbol{C}|11\rangle &= \boldsymbol{C}|1\rangle\langle 1| \\
&= \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} \\
&= |10\rangle \qquad (6.41)
\end{aligned}$$

We are now ready to generalise our formalism to qubits.

6.3.2 Quantum Bits

A qubit is a is a bit that is in a superposition of two states (say on and off or 1 and 0), rather than exclusively in either one of two states as in a classical or binary bit. In other words, a qubit is in a continuum of the two possible states 0 and 1. A qubit still collapses into a binary bit, but like all quantum systems, the final state has a probabilistic outcome given by its probability density. In our Dirac notation, the qubit can be represented as

$$|\boldsymbol{q}\rangle = \begin{pmatrix} \mu \\ \nu \end{pmatrix} \qquad (6.42)$$

where μ and ν are arbitrary complex numbers, so that $\langle \boldsymbol{q}|\boldsymbol{q}\rangle = 1$ and has a probability density of $|\mu|^2 + |\nu|^2$. In other words, the qubit has a $|\mu|^2$ probability of collapsing into state 0 and $|\nu|^2$ probability of collapsing into state 1.

Notice that the classical bit defined in equation (6.23) is a subset of a qubit, but now that states 0 and 1 are replaced by complex numbers. Some example qubit states include

$$\begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \qquad \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}, \qquad \begin{pmatrix} \frac{1}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix} \tag{6.43}$$

The first of these has a probability density of $1/2 + 1/2$, which implies that the qubit state is equally in both the on and off states simultaneously. In contrast to a classical bit, $|0\rangle$ has a probability of 1 to collapse to the state 0 and probability of 0 to collapse to the state 1, while $|1\rangle$ has a probability of 0 to collapse to the state 1 and probability of 1 to collapse to the state 1.

Thus, qubits can hold multiple bit states at once during a computation before they collapse when we measure them. A series of qubits then can do parallel computations, so that each thread of that computation, and its corresponding intermediate results, is represented as one of the many superimposed states held in the qubits. In other words, each thread of the parallel computation exists as a single harmonic across the qubits in a cacophony of harmonics created by superposition that represents the entire computation.

To create multiple qubits, we follow the same process constructed in section 6.3.1. A two qubit system would have

$$\begin{aligned} |\boldsymbol{qr}\rangle = |\boldsymbol{q}\rangle \langle \boldsymbol{r}| &= \begin{pmatrix} \mu_1 \\ \nu_1 \end{pmatrix} \otimes \begin{pmatrix} \mu_2 \\ \nu_2 \end{pmatrix} \\ &= \begin{pmatrix} \mu_1\mu_2 \\ \mu_1\nu_2 \\ \nu_1\mu_2 \\ \nu_1\nu_2 \end{pmatrix} \end{aligned} \tag{6.44}$$

so that $|\mu_1\mu_2|^2 + |\mu_1\nu_2|^2 + |\nu_1\mu_2|^2 + |\nu_1\nu_2|^2 = 1$. For example, for the states

$$|\boldsymbol{q}\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix}, \qquad |\boldsymbol{r}\rangle = \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \tag{6.45}$$

will produce the following state

$$\begin{aligned} |\boldsymbol{qr}\rangle = |\boldsymbol{q}\rangle \langle \boldsymbol{r}| &= \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \otimes \begin{pmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \\ \frac{1}{2} \end{pmatrix} \end{aligned} \tag{6.46}$$

which has the probability density of $\frac{1}{4}+\frac{1}{4}+\frac{1}{4}+\frac{1}{4}=1$. In other words, there is a $\frac{1}{4}$ probability that the system will collapse to one of $|00\rangle$, $|10\rangle$, $|01\rangle$ or $|11\rangle$ states.

The operators that we have constructed so far work in exactly the same way as for the formulation in section 6.3.1. There are however, several operators that only make sense in the quantum realm, which then allows us to build new types of circuits that can exploit the quantum weirdness for computation.

6.4 Quantum Circuits

An important step in building qubits is to be able to convert between classical and quantum states.

6.4.1 Operators

The Hadamard operator $\boldsymbol{H}$ allows us to create the superposition of states of a qubit with its form

$$\boldsymbol{H}=\begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{pmatrix}=\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} \tag{6.47}$$

For example, we can convert the states $|0\rangle$ and $|1\rangle$ to superpositions

$$\boldsymbol{H}|0\rangle = \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}1\\0\end{pmatrix}=\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix} \tag{6.48}$$

$$\boldsymbol{H}|1\rangle = \begin{pmatrix}\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}}\end{pmatrix}\begin{pmatrix}0\\1\end{pmatrix}=\begin{pmatrix}\frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}}\end{pmatrix} \tag{6.49}$$

Operators applied in this way take a quantum state present on one point on the unit circle to another point on the same circle. Generally, the vectors and operators are in the set of complex numbers $\mathbb{C}$, so that states are actually points on the Bloch sphere, which is a unit 2-sphere, i.e. like a surface of a ball.

However, quantum operators must have special mathematical properties to ensure that total probability is preserved, that is it does not change the norm of the state vector though it may change its direction. It follows that a quantum operator $\boldsymbol{U}$ must satisfy

$$\boldsymbol{U}\boldsymbol{U}^{\dagger}=\boldsymbol{I} \tag{6.50}$$

where the $\dagger$ represents the Hermitian conjugate $\boldsymbol{U}^{\dagger}=(\boldsymbol{U}^{\boldsymbol{T}})^{*}$. It therefore also implies that $\boldsymbol{U}^{\dagger}=\boldsymbol{U}^{-1}$, i.e. the operator $\boldsymbol{U}$ is self-adjoint or it is its own inverse when in Hermitian conjugate form. Equivalently, the column vectors of the matrix representing $\boldsymbol{U}$ must be orthonormal and matrices of this form are therefore referred to as unitary matrices. We can see this in the presence of the negative value in the Hadamard operator $\boldsymbol{H}$ that gives it this property. All operations in a quantum computer are simply rotations or reflections applied by unitary operators in a high dimensional vector space.

Thus, because the Hadamard operator is unitary and self-adjoint, we can convert a qubit superposition, like the ones presented in equations (6.48) and (6.49), back into a classical state simply by applying the operator $\boldsymbol{H}$ again. This is indicative of many quantum computing operators and algorithms, allowing one to convert between probabilistic and deterministic output, and represents the notion of reversible computing.

6.4.2 Circuits

To build quantum computing circuits, we represent each qubit with a 'wire' and the operators as boxes or symbols on these wires. For example, we write the following to represent a qubit and the Hadamard operator $\boldsymbol{H}$ acting on that qubit as shown in figure 6.4.2. The result of this

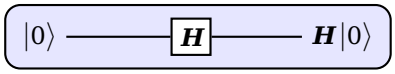

Figure 6.3: A quantum circuit for producing a superposition of states in equation (6.48).

operation will be the same as that shown in equation (6.48). We can also represent composition of operators $\boldsymbol{U}_2\boldsymbol{U}_1$ and the tensor product of operators $\boldsymbol{U}_1 \otimes \boldsymbol{U}_2$ of any two valid quantum operators $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ as boxes in series and parallel respectively as shown in figure 6.4.2. We

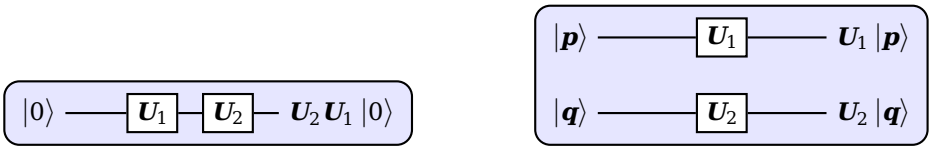

Figure 6.4: A quantum circuits for producing a composition and tensor products of operators.

can then create the four circuits for our the four operations on a single qubit using these diagrams.

However, out of the four operations on the single bit, the operations of Set $|0\rangle$ and Set $|1\rangle$ do not have a self adjoint, i.e. they do not obey equation (6.50) and are not reversible while the other two operations (identity and negation) are reversible. The Set operations overwrite the result regardless of the input that makes it impossible to recover the initial input without further information. We need to create a larger matrix that can still encode this operation and obey equation (6.50), thereby making these operations reversible. We can do so by using an additional qubit to encode the output (top wire) and input qubit (bottom wire), so that we can tell what happened and reverse the operation. This means all our four operations will have to be constructed using two wires and encode the operations accordingly. Once we use two wires, the Set operations are fairly simple and straight forward (see figure 6.4.2 on the next page). They simply involve two non-interacting wires assuming the output qubit is always $|0\rangle$ initially. By default, such a setup is already a Set $|0\rangle$ operation and we can use a simple NOT operator $\boldsymbol{X}$ to obtain the Set $|1\rangle$.

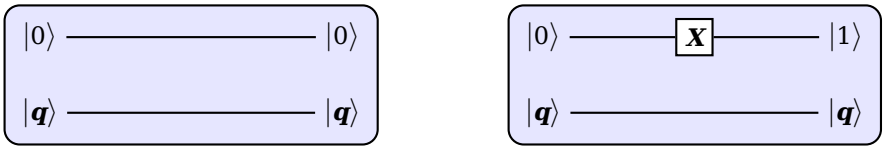

Figure 6.5: A quantum circuits for producing Set $|0\rangle$ and $|1\rangle$ operations.

To implement a simple identity operation, we require the use of the CNOT operator to create the circuit as shown in figure 6.4.2 on the left. The dark circle is the control bit and it decides

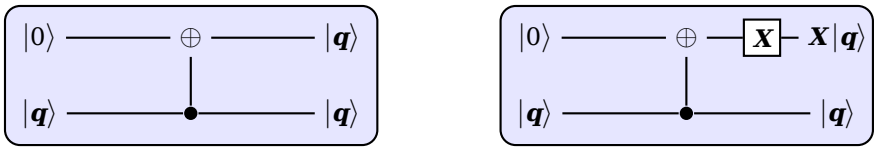

Figure 6.6: A quantum circuit for producing an identity and negation operations on states.

whether the output bit is negated or not. Thus, the circuit function as an identity operation by only flipping the output qubit to $|1\rangle$ when input bit is also $|1\rangle$, otherwise it remains as $|0\rangle$, which is also the same as the input in this case. The negation operation shown in figure 6.4.2 on the right is simply the negated identity result using the $\boldsymbol{X}$ (NOT) operator on the output qubit.

The circuits presented so far represent the basic notations required for build quantum algorithms. Now we have all the necessary machinery to build the algorithms that will allow us to harness the main power of quantum computing: exponential speed ups.

6.5 Deutsch's Oracle Problem

Imagine now that we have a problem of trying to determine which of the four operations exist as a black box that we know nothing about. It can take inputs and produce outputs, but we are not told anything about the operator that resides inside the box. We are allowed to provide inputs and measure outputs, i.e. for inputs and output states $|0\rangle$, we may get the arbitrary states $|\boldsymbol{p}\rangle$ and $|\boldsymbol{q}\rangle$ (see figure 6.5). Can we determine what the operator is within the box with

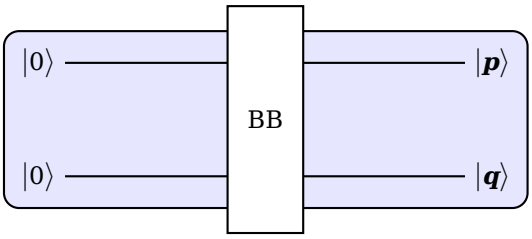

Figure 6.7: Deutsch's oracle problem with an black box (unknown) operator.

as few queries as possible?

Classically, for a single bit of information, we would require a total of 2^1 queries into the black box to determine what operation was actually applied. For a series of $\boldsymbol{n}$ bits, a total of $2^{\boldsymbol{n}}$ queries would be required. This is because for every bit input to the box, the bit could've have been negated or set to a state and the operation would be indistinguishable from each other. The only way to know is to try as many bits as uniquely possible to determine the operator in the box. However, as we shall see, with a quantum computer, we can always determine the operator present with only a single query resulting in an exponential speed up! For the purposes of this book, we will examine the single bit black box and solve the problem using our quantum circuits from the previous section.

Consider the four operations on a single bit: identity, negation, set $|0\rangle$ and $|1\rangle$. The last two are constant operations that do not take the input state into account. We can see this as the quantum circuit for these wires that do not interact unlike the first two operators that involve the CNOT gate. We can thus divide the operators up into two categories: variable and constant operators. Our goal is to use superposition of qubits to separate out the two types of operators. The variable type of operator should interact with the output and input bits to produce a different tensor product state than for those constant operators that do not interact.

Now consider the quantum circuit for Deutsch's problem as show in figure 6.5, where the $\boldsymbol{M}$ operator represents making the measurement. We can insert the relevant quantum circuits

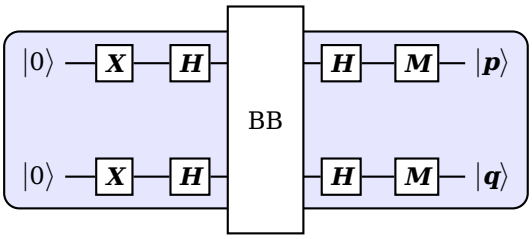

Figure 6.8: A quantum circuit for Deutsch's problem with the black box (unknown) operator.

corresponding to the four possible operations in place of the black box in figure 6.5 and analyse the outcomes. For Set $|0\rangle$ (left) and Set $|1\rangle$ (right) operators as shown in figure 6.5, we can replace the black box and evaluate what happens. Essentially, the Set $|0\rangle$ and Set $|1\rangle$ operations

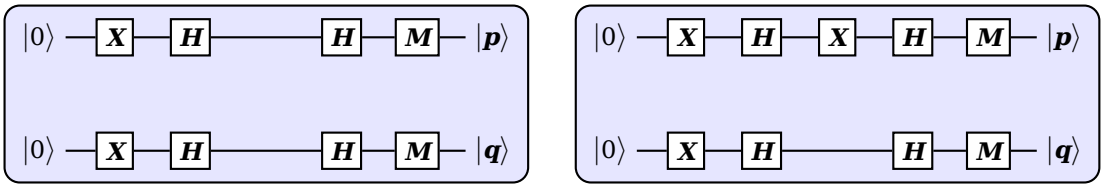

Figure 6.9: Quantum circuits for Deutsch's problem as Set $|0\rangle$ and $|1\rangle$ operators.

pass through without any changes as our previous circuit constructions have shown. We can write out the (tensor) product state of the these circuits to see the outcomes. Top and bottom

wires of the Set $|1\rangle$ will result in

$$\begin{aligned} \boldsymbol{HHX}|0\rangle &= \boldsymbol{X}|0\rangle \\ &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.51}$$

The product state is then just $|11\rangle$ as seen from equation (6.40). The bottom wire in Set $|1\rangle$ is the same as equation (6.51) and the top wire is simply

$$\begin{aligned} \boldsymbol{HXHX}|0\rangle &= \boldsymbol{HXH}|1\rangle \\ &= \boldsymbol{HX} \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix} = \boldsymbol{HX} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \\ &= \boldsymbol{H} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix} = \boldsymbol{H} \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} \\ &= \begin{pmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \end{pmatrix} \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{pmatrix} = \begin{pmatrix} 0 \\ -1 \end{pmatrix} \\ &= |1\rangle \end{aligned} \tag{6.52}$$

Since the probability density of the result becomes $0^2 + (-1)^2$ is the same as $|1\rangle$, the resultant state is $|1\rangle$. Thus when using the constant operators, the result is always $|11\rangle$.

Now examine the result if the black box operator is replaced with the variable operators: identity and negation as shown in figure 6.5. In both instances we use a CNOT gate, so we must

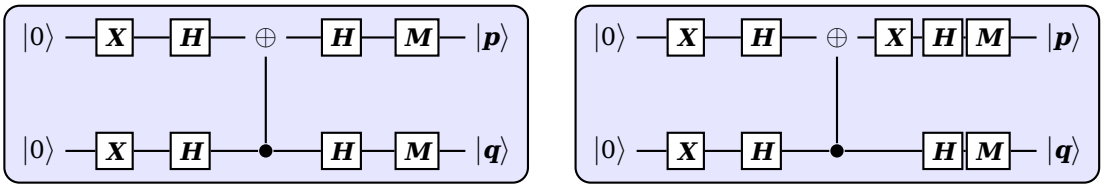

Figure 6.10: Quantum circuits for Deutsch's problem as variable operators identity and negation.

use the product states $|00\rangle$ etc. throughout. Since both $\boldsymbol{X}$ and $\boldsymbol{H}$ to both qubits independently, we can simplify the expression so that only the CNOT operation is left to apply. Thus, we can

write the identity operation as

$$
\begin{aligned}
\boldsymbol{CHX}|00\rangle &= \boldsymbol{C}\left(\begin{pmatrix}\frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}}\end{pmatrix}\otimes\begin{pmatrix}\frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}}\end{pmatrix}\right) = \boldsymbol{C}\begin{pmatrix}\frac{1}{2}\\ -\frac{1}{2}\\ -\frac{1}{2}\\ \frac{1}{2}\end{pmatrix} \\
&= \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{pmatrix}\begin{pmatrix}\frac{1}{2}\\ -\frac{1}{2}\\ -\frac{1}{2}\\ \frac{1}{2}\end{pmatrix} = \begin{pmatrix}\frac{1}{2}\\ -\frac{1}{2}\\ \frac{1}{2}\\ -\frac{1}{2}\end{pmatrix} \\
&= \frac{1}{2}\begin{pmatrix}1\\ -1\\ 1\\ -1\end{pmatrix} = \left(\begin{pmatrix}\frac{1}{\sqrt{2}}\\ \frac{1}{\sqrt{2}}\end{pmatrix}\otimes\begin{pmatrix}\frac{1}{\sqrt{2}}\\ -\frac{1}{\sqrt{2}}\end{pmatrix}\right) \\
&= |01\rangle
\end{aligned}
\tag{6.53}
$$

Likewise apply a similar analysis for the negation circuit, we simply need to handle the additional NOT gate on the output bit. The result is similar to equation (6.52). Therefore, the resultant state remains $|01\rangle$ as with the identity case. Thus, we have shown that we can classify the unknown operator into one of the two categories as predicted but with only a single query. If the resulting state from input and output states $|0\rangle$ is $|11\rangle$, then it is a constant operator and if the resulting state is $|01\rangle$ it is a variable operator.

The above can be extended to $\boldsymbol{n}$ bits and is known as the Deutsch-Josza problem [Deutsch and Jozsa, 1992]. Although this problem is out of scope for this book, it can be seen that the superposition of states introduced by the Hadamard gate is the key in separating out the two categories. We can note that the different between constant and variable operations is the CNOT gate, so we can also view it as amplifying the difference between the categories and minimising the similarities between the categories. This is similar to how Shor [1994] constructed his algorithm for factoring large numbers, by using periodicity of roots of unity, we can cancel out the undesired categories to give us the answer we seek. These and other algorithms are set to benefit greatly when qubits are made more stable, so that quantum computers can become more readily available.

Abbreviations

1D one dimensional 19
2D two dimensional 11
3D three dimensional iv
GCD greatest common divisor 8
ACE automatic computing engine 18
FSM Finite State Machine iv
QM quantum mechanics 41

Bibliography

Berlekamp, E., J.Conway, R.Guy, 1982. Winning Ways for your Mathematical Plays. Vol. 2.

Church, A., 1932. A set of postulates for the foundation of logic. Annals of Mathematics 33 (2), 346–366.
URL https://doi.org/10.2307/1968337

Curry, H. B., Feys, R., 1958. Combinatory Logic, Volume I. North-Holland.

Deutsch, D., Jozsa, R., 1992. Rapid solution of problems by quantum computation. Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences 439 (1907), 553–558.
URL https://doi.org/10.1098/rspa.1992.0167

Diophantus, 100. Arithmetica. Springer-Verlag Berlin and Heidelberg GmbH & Co. KG (December 31, 1982).

Dirac, P. A. M., 1930. The Principles of Quantum Mechanics. Clarendon Press.

Einstein, A., 1905. Über einen die erzeugung und verwandlung des lichtes betreffenden heuristischen gesichtspunkt. Annalen der Physik 322 (6), 132–148.
URL https://doi.org/10.1002/andp.19053220607

Euclid, 300BCE. The Elements.

Euler, L., 1763. Theoremata Arithmetica Nova Methodo Demonstrata. Novi Commentarii Academiae Scientiarum Petropolitanae 8, 74–104.

Feynman, R., 2005. The Pleasure of Finding Things Out: The Best Short Works of Richard P. Feynman. Helix Books.

Gardner, M., 1970. Mathematical games: The fantastic combinations of john conway's new solitaire game "life". Scientific American (223), 120–123.
URL http://www.jstor.org/stable/24927642

Gauss, C. F., 1801. Disquisitiones Arithmeticae. Yale Univeristy Press.

Gödel, K., Dec 1931. Über formal unentscheidbare sätze der principia mathematica und verwandter systeme i. Monatshefte für Mathematik und Physik 38 (1), 173–198.
URL https://doi.org/10.1007/BF01700692

Klein, F., Mar 1893. Vergleichende betrachtungen über neuere geometrische forschungen. Mathematische Annalen 43 (1), 63–100.
URL https://doi.org/10.1007/BF01446615

Langton, C. G., 1986. Studying artificial life with cellular automata. Physica D: Nonlinear Phenomena 22 (1), 120 – 149, proceedings of the Fifth Annual International Conference.
URL https://doi.org/10.1016/0167-2789(86)90237-X

Moore, C., Mertens, S., 2011. The Nature of Computation. Oxford University Press, Inc., New York, NY, USA.

Schrödinger, E., Dec 1926. An undulatory theory of the mechanics of atoms and molecules. Phys. Rev. 28, 1049–1070.
URL https://doi.org/10.1103/PhysRev.28.1049

Shor, P. W., 1994. Algorithms for quantum computation: Discrete logarithms and factoring. In: Proceedings 35th annual symposium on foundations of computer science. IEEE, pp. 124–134.
URL https://doi.org/10.1109/SFCS.1994.365700

Sipser, M., 2013. Introduction to the theory of computation / Michael Sipser., 3rd Edition. Cengage Learning, Andover.

Turing, A. M., 1937. On computable numbers, with an application to the entscheidungsproblem. Proceedings of the London Mathematical Society s2-42 (1), 230–265.
URL https://doi.org/10.1112/plms/s2-42.1.230