The Quantum Computational Model

Spencer Little

August 2, 2019

1 Abstract

The foundation of modern information technology rests on the classical bit. With some requisite knowledge about logic gates and circuits it becomes clear just how natural the mathematical model for bits is. We have two definite, deterministic states that cycle through many series of logic gates which produce the desired computations. Yet as we approach the inevitable end of Moore's law one troubling question remains; *how can we continue to improve the efficiency of our processors if our transistors are so small that electrons are tunnelling across them?* Quantum computation offers a solution to this problem.

The quantum bit (qubit) is a far more complicated unit of information that is build upon the principles of quantum mechanics. To harvest it's power and understand it's promise we must look deeper than the comforting binary world classical computers reside in. The following paper describes a mathematical framework for representing quantum bits and qubit mappings using linear algebra. It will be shown how to model quantum probabilities, construct logic gates, and ultimately implement a simple quantum algorithm: Deutch's Algorithm.

2 Introduction

The quantum world requires a different paradigm to model computation. This paper contains a survey of the theory behind quantum computation and the mathematics that describes it. Namely it will be shown how qubits can be represented as unit vectors in $\mathbb{C}^2$ and how qubit mappings can be described as unitary matrices. After some discussion of quantum probabilities and qubit models, Dirac notation will be explored. Finally, a humble but seminal glimpse into the potential of quantum computing will be discussed; Deutch's Algorithm.

The technical insights in this paper draw from a series of lecture slides (developed by Kevin Resch at the University of Waterloo Department of Phsyics and Astronomy) and a compilation of lecture transcripts from Dave Bacon's CSE 599 (Quantum Computing) course at the University of Washington. See the references section for a comprehensive list of resources.

3 Quantum States

Perhaps the most apt analogy between classical computing and quantum computing is that of the difference between the candle and the light bulb. Quantum computing is a fundamentally different technology. Classical bits are easily represented as elements from the alphabet $(0, 1)$. It is not difficult to see why such a representation is convenient given it's physical incarnation; high/low voltages over wire. However, if we want to compute with particles we no longer have the luxury of classical probability. The quantum world is more properly described in the language of states which are represented by unit vectors with coefficients from the complex plane.

The conventional model for a qubit is a unit vector in $\mathbb{C}^2$. The vectors corresponding to the two classical binary states are the basis $e_1 = (1, 0)^T, e_2 = (0, 1)^T$ and a linear combination, ψ, with coefficients $c_1, c_2 \in \mathbb{C}$ is a valid qubit as long as $|\psi| = 1$ or equivalently $|c_1|^2 + |c_2|^2 = 1$. So our new model looks like

$$\psi = c_1 e_1 + c_2 e_2 \tag{1}$$

The probability of the qubit being in either state is given by the square of the respective coefficient.

To develop some intuition for this model I will briefly describe an experimental setup involving the use of the spatial path of a photon to model a quantum bit. For the purposes of this paper I will not be discussing the physical intricacies of these experiments. I will focus only on the superficial insights necessary to generalize a mathematical model. The following calculations have been adapted from a series of lecture slides made available from the Kevin Resch at the University of Waterloo's Institute for Quantum Computing [1].

In a Mach-Zehnder interfermoter photons travel through a series of two beamsplitters. After the first beamsplitter they are reflected by a mirror to direct their path into the second beamsplitter. Such devices are useful for studying quantum phenomena. As such they also provide a convenient model for a qubit in the form of the path that a given photon takes through the device.

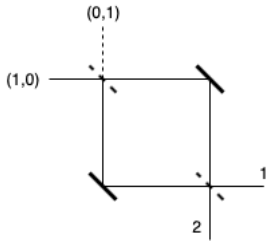

Figure 1: A diagram depicting the possible spatial paths through a Marc-Zehnder interferometer.

Let $(1, 0)$ denote the state of our "qubit" when it enters the MZ interferomter from the left. Our objective is to consider how each split in the path affects the

state of the qubit. After entering the first split the state is transformed by the beamsplitter (which we assume to be a random 50/50 split). We can model this transition with the following matrix multiplication.

$$\psi = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\begin{pmatrix}1\\0\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} \tag{2}$$

The matrix used to model this 50/50 transition is known as the Hadamard gate. The Hadamard gate, H, is a particularly useful transformation that creates a uniform, but random probability distribution. This property of the Hadamard gate makes it useful for creating unbiased coin flips. When applied to a constant qubit, the measurement of that qubit will represent a perfectly random coin toss.

So now the qubits probability distribution is $((\frac{1}{\sqrt{2}})^2, (\frac{1}{\sqrt{2}})^2) = (\frac{1}{2}, \frac{1}{2})$, which faithfully models the path of a photon through an MZ interferometer. After the first beamsplitter we define a phase, $\phi = \frac{2\pi\Delta L}{\lambda}$, where ΔL = path length and λ = wavelength.

Accounting for the phase

$$\psi = \begin{bmatrix}1 & 0\\ 0 & e^{i\phi}\end{bmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1\\1\end{pmatrix} = \frac{1}{\sqrt{2}}\begin{pmatrix}1\\e^{i\phi}\end{pmatrix} \tag{3}$$

Then considering the photon will once again pass a beamsplitter

$$\psi = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\frac{1}{\sqrt{2}}\begin{pmatrix}1\\e^{i\phi}\end{pmatrix} = \frac{1}{2}\begin{pmatrix}1+e^{i\phi}\\1-e^{i\phi}\end{pmatrix} \tag{4}$$

Finally, computing the probability of detection (making use of Eulers formula $e^{i\phi} = \cos\phi - i\sin\phi$)

$$\begin{aligned}|c_1|^2 &= \frac{1}{4}(1+e^{i\phi})(1+e^{i\phi})\\ &= \frac{1}{4}(1+2\cos\phi - 2i\sin\phi + \cos^2\phi + 2i\sin\phi\cos\phi - \sin^2\phi)\end{aligned} \tag{5}$$

$$\begin{aligned}|c_2|^2 &= \frac{1}{4}(1-e^{i\phi})(1-e^{i\phi})\\ &= \frac{1}{4}(1-2\cos\phi + 2i\sin\phi + \cos^2\phi + 2i\sin\phi\cos\phi - \sin^2\phi)\end{aligned} \tag{6}$$

The equations above may seem unruly however, since we are considering our photon to be a qubit the global phase does not have any observable consequences so we may let $\phi = 0$ [2][1]. Now we can see that $\psi = (1, 0)$. This result realistically describes the output of a MZ-interferometetr in certain circumstances; all photons end up taking one path after the second beamsplitter. For completeness I will note that this result can also be understood classically

[1] Dave Bacon in a transcript from lecture notes: "for quantum states, a global phase for the state never has any observable consequences...it is useful to always choose the global phase such that the coefficient of the $|0\rangle$ ket is real an non-negative."

as interference. However, if we are considering a quantum perspective (a single photon), we must introduce a quantum state like the one described above. This state describes a superposition, which is a way of representing a confusing property of quantum mechanics that shows that a single photon takes both paths through the MZ-interfermoter simultaneously.

Counterintuitive principles like these exemplify the quantitative challenges associated with constructing a computational model built upon quantum mechanics. Classical mathematical models are not capable of representing the intricacies of such confounding properties. In the coming sections the efficacy of the complex unit vector as a model for the qubit will be further elucidated.

4 Quantum Operations

4.1 Dirac Notation

Before we generalize the model proposed above it is necessary to take a detour into the world of Dirac (or bra-ket) notation. Dirac notation is widely accepted as the standard notation for representing quantum states and as such it is requisite for the representation of quantum systems.

From the discussion above we can see why it is convenient to express qubits as vectors in a complex vector space. More specifically we will represent qubits as vectors in a complex Hilbert space. Lending form Dave Bacon's lecture notes on this topic [3], the Hilbert space we will be making use of is the vector space $\mathbb{C}^N$ with the inner product $w = [w_0, w_1, \ldots, w_{n-1}]^T, v = [v_0, v_1, \ldots, v_{n-1}]^T$

$$\langle w, v \rangle = \sum_{i=0}^{N-1} w_i^* v_i \tag{7}$$

where w^* denotes the complex conjugate of w which is defined by $x, y \in \mathbb{R}, w = x + iy, w^* = x - iy$. Let this Hilbert space be denoted by $\mathcal{H}$.

We then define a vector in this space by $|v\rangle \in \mathcal{H}$, formally known as a ket. It is convenient to think of $|v\rangle$ as a column vector, $[v_0, v_1, \ldots, v_{n-1}]^T$ where $v_i \in \mathbb{C}$. The counterpart of the ket is the bra, which is a vector composed of elements from the dual vector space for our original Hilbert space, $\mathcal{H}^*$. Bras are denoted by $\langle w|$ and they can be thought of as row vectors $\langle w| = [w_0, w_1, \ldots w_{n-1}], w_i \in \mathcal{H}^*$. In the Hilbert space defined above every ket has a corresponding bra in which for each $v_i \in |v\rangle$ there exists $v_i^* \in \langle v|$. That is, each element of the bra is the complex conjugate of the corresponding element of the ket. This relationship also holds inversely.

Now we can interpret the model introduced in equation 1 using dirac notation

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle \tag{8}$$

where $\alpha, \beta \in \mathbb{C}$ and $|0\rangle = (1, 0)^T, |1\rangle = (0, 1)^T$. This equation formally describes a single qubit as a vector in a complex Hilbert space. This definition will form the foundation for the following sections.

The inner product between two vectors, $|v\rangle, |w\rangle$ is denoted by $\langle v, w\rangle = \langle v|w\rangle$. Notice that now if we consider $\langle v|$ to be a bra vector and $|w\rangle$ to be a ket vector the inner product can be interpreted much like it was introduced in equation 7.

$$\langle v|w\rangle = [v_0^*, v_1^*, \ldots, v_{n-1}^*] \begin{bmatrix} w_0 \\ w_1 \\ \vdots \\ w_{n-1} \end{bmatrix} = \sum_{i=0}^{n-1} v_i^* w_i \tag{9}$$

It is convenient to note that since $\langle v|$ is a column vector and $|w\rangle$ is a row vector the operation described above is simply matrix multiplication.

So we have now defined a qubit as a two level quantum system represented by a vector in a complex Hilbert space. Next we will explore qubit transformations; an integral step in constructing the quantum computational model.

4.2 Qubit Mappings

Now that we have introduced a way of representing qubits the next step is to define a way to transform the bits. Since any transformation that we make use of must map a unit vector to a unit vector and preserve the inner product defined above, the transformation must be linear. Transformations such as these are described by unitary matrices. Formally, a square complex matrix is defined as unitary if

$$U^H = U^{-1} \tag{10}$$

That is, if the conjugate transpose of the matrix is also it's inverse. The conjugate transpose of a matrix is defined by transposing the conjugate matrix (which is formed by taking the complex conjugate of each respective element in U). We can now defined a single qubit gate as a linear transformation that maps $\mathbb{C}^2 \to \mathbb{C}^2$ and preserves this inner product. This transformation can be represented by a unitary matrix. As it turns out we have already introduced a qubit gate in section 3: the Hadamard gate.

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{11}$$

As we observed while following the photon through the MZ interferometer the Hadamard gate creates a uniform superposition. It is also interesting to note that the Hadamard gate squares to the inverse, $H^2 = I$, and thus is considered reversible. These properties make it a very useful operation for constructing quantum algorithms.

Some other important gates are the Pauli operators. There are four total but here we will only discuss the Pauli X gate (NOT).

$$\sigma_1 = \sigma_x = X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{12}$$

The Pauli X gate behaves much like the classical NOT gate. It maps $|0\rangle \rightarrow |1\rangle$ and vice versa. It is interesting to note that all Pauli operators square to the identity [2]. This hints at a powerful prospect of quantum computing: logical reversibility. Unfortunately, reversibility is beyond the scope of this paper.

The final feature we will explore before demonstrating the power of quantum computing with Duetch's Algorithm is a representation of multiple qubits.

4.3 Tensor Products

To represent multiple qubits we will make use of the tensor product. Denoted by $|v\rangle \otimes |w\rangle$, the tensor product is an operation on two vectors or vector spaces, in this case $\mathcal{H} \rightarrow \mathcal{H}$, that creates an $m \times n$ dimensional space, where m and n are the respective dimensions of the initial vector spaces. The new space has a basis that is a combination of the tensor products of the basis vectors of the original two spaces. To develop some intuition for what tensor products look like an example is provided below.

$$(1,2,3)^T \otimes (2,3)^T = \begin{bmatrix} 1\times 2 \\ 1\times 3 \\ 2\times 2 \\ 2\times 3 \\ 3\times 2 \\ 3\times 3 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \\ 4 \\ 6 \\ 6 \\ 9 \end{bmatrix} \tag{13}$$

Each vector element of the first vector is multiplied by the second vector and the results are stacked into a column to form the product. The result can also be interpreted as a matrix corresponding to the linear transformation defined by the tensor product operation. However, for the purposes of this paper we will only be considering the products to be vectors.

4.4 Multiple Qubits

There are four possible configurations of a system with two qubits $|0\rangle \otimes |0\rangle, |0\rangle \otimes |1\rangle, |1\rangle \otimes |0\rangle$, and $|1\rangle \otimes |1\rangle$. However it is common practice to drop the tensor product and append the qubits within a ket so the qubits above can be written as $|00\rangle, |01\rangle, |10\rangle$, and $|11\rangle$. Extending the definition of a qubit described in equation 8 we arrive at

$$|\psi\rangle = \alpha_{00}|00\rangle + \alpha_{01}|01\rangle + \alpha_{10}|10\rangle + \alpha_{11}|11\rangle \tag{14}$$

where $\alpha_{ij} \in \mathbb{C}$ and $|\alpha_{00}|^2 + |\alpha_{01}|^2 + |\alpha_{10}|^2 + |\alpha_{11}|^2 = 1$. We now have a model capable of representing multiple qubits. When considering multiple qubits the dimension of our vectors increases exponentially. For instance, two qubits will be represented by a tensor product of dimension $2^2 = 4$, a vector representing three qubits will have dimension $2^3 = 8$ and so fourth. The dimension of the transformation matrices scales proportionally as well. Consider the CNOT gate

(short for controlled NOT gate, an operation involving two bits where a NOT gate is applied conditionally depending on the value of a control bit).

$$C_X = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{15}$$

The CNOT gate will flip the second bit if the control bit is 1. If the control bit is 0 there will be no effect. We can observe this by applying C_X to $|10\rangle$

$$C_X|10\rangle = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = |11\rangle \tag{16}$$

In this circumstance our control bit was $|1\rangle$ and the affected bit was $|0\rangle$ which was transformed to $|1\rangle$. The CNOT gates effect can be summarized as follows

$$\begin{gathered} |00\rangle \to |00\rangle \\ |01\rangle \to |01\rangle \\ |10\rangle \to |11\rangle \\ |11\rangle \to |10\rangle \end{gathered}$$

We now have a method for representing and manipulating multiple qubits. The journey from a binary probability distribution, to a single quantum state, through unitary matrices representing logic gates, and finally to generalizations involving multiple qubits has lead us to a place where we can now investigate the power of quanutm computing through a colloquial example of the algorithmic ingenuity possible in the quantum realm.

5 Duetch's Problem

David Duetch was one of the first people to actualize the promise of quantum computing [4]. Duetch was able to demonstrate the superiority of quantum processes through the lense of a simple algorithmic problem. Strictly speaking the method that will be described in this section is not actually an algorithm, nonetheless the approach involves computational ingenuity so it has been regarded as such in this paper.

Duetch's problem is described as follows [2]. We defined a function $f(x, y)$ that operates on two bits. Then we consider a black box that implements the function. The objective is to determine what function is implemented by the box by querying the box the fewest number of times. There are four possible functions that the box could be implementing:

$$\begin{gathered} f_1(x, y) \to (x, y) \\ f_2(x, y) \to (x, !y) \\ f_3(x, y) \to (x, x \oplus y) \\ f_4(x, y) \to (x, x \oplus !y). \end{gathered}$$

The problem is defined in such a way that our goal is to determine whether the function implemented belongs to $S_1 = \{f_1, f_2\}$ or $S_2 = \{f_3, f_4\}$. Thinking for a moment we can see why this task will take at least two queries if we are thinking from a classical perspective. However, if we implement this problem with the quantum model discussed above we can solve Duetch's problem in a single query. First, consider how we can represent each function described above using a 4×4 unitary matrix. For instance, f_3 is simply the CNOT gate described in equation 15. The other transformation are defined as follows [2]

$$\begin{aligned} U_1 &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} & U_2 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ U_3 &= C_X & U_4 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{17}$$

To solve Duetch's problem we will begin in the state $|01\rangle$. The first step is to apply the 4×4 Hadamard gate. This gate is represented by the matrix.

$$H \otimes H = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \tag{18}$$

Applying this transformation to $|01\rangle$

$$|\phi\rangle = (H \otimes H)|01\rangle = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix} \tag{19}$$

Then rewriting this as a two qubit system

$$|\phi\rangle = \frac{1}{2} \begin{pmatrix} 1 \\ -1 \\ 1 \\ -1 \end{pmatrix} = \frac{1}{2} \left(\begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 1 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \right) \tag{20}$$

Recognizing the vectors above as qubits we can denote the above state as

$$|\phi\rangle = \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle) \tag{21}$$

We then consider each of the transformations defined above

$$\begin{aligned} U_1|\phi\rangle &= \frac{1}{2}(|00\rangle - |01\rangle + |10\rangle - |11\rangle) \\ U_2|\phi\rangle &= \frac{1}{2}(-|00\rangle + |01\rangle - |10\rangle + |11\rangle) \\ U_3|\phi\rangle &= \frac{1}{2}(|00\rangle - |01\rangle - |10\rangle + |11\rangle) \\ U_4|\phi\rangle &= \frac{1}{2}(-|00\rangle + |01\rangle + |10\rangle - |11\rangle) \end{aligned} \tag{22}$$

Finally we can apply $H \otimes H$ again to see

$$\begin{aligned} (H \otimes H)U_1|\phi\rangle &= |01\rangle \\ (H \otimes H)U_2|\phi\rangle &= -|01\rangle \\ (H \otimes H)U_3|\phi\rangle &= |11\rangle \\ (H \otimes H)U_4|\phi\rangle &= -|11\rangle \end{aligned} \tag{23}$$

We are now able to determine whether the function implemented is an element of S_1 or S_2 by observing the first bit in our system. While this may seem link a negligible improvement this result is actually quite significant. It was among the very first proofs that showed a tractable method for improving upon classical models using the quantum paradigm. This algorithm was later extended to the problem of searching and has been implemented on a physical quantum computer. [5]

6 Conclusion

Leaving behind the world of deterministic, binary states we forayed into a land of amplitudes and uncertainty. We saw how quantum bits can be represented by unit vectors in a complex Hilbert space, how these bits can be transformed by unitary matrices, and how we can combine these operations to construct a simple algorithm. The content of this paper barely scratches the surface of quantum computing. There are many fundamental aspects of the quantum computational model not discussed here. A more extensive paper may have included sections about how qubits can be interpreted geometrically within the context of the Bloch sphere, how logical reversibility is a foundation of the quantum model, and perhaps some discussion of Shor's algorithm.

The journey from classical, binary bits into the quantum realm is not a straight forward path. However, some think that the algorithmic challenges faced by computer scientists are not due to their lack of ingenuity but rather their predilection to compute in a paradigm that is at odds with the fundamental principles of the universe. While the principles of quantum mechanics seem to operate in ways that classical logic struggles to comes to terms with, it may be that this quantum logic forms the basis for the most powerful computational model possible. In this sense adapting to the quantum model shouldn't

be seen as transition to a foreign and unusual model. Rather this transition should be seen as shifting our computational foundation to align with the most fundamental phsyical discoveries of the 21^{st} century.

7 Reflection

I do not have any partners to assess but would say that my own contributions to my project were sufficient. This was an exciting project for me. I learned a lot and am left with more questions than answers, which I suppose is a good thing. I find it very exciting to finally be reaching a level in math where I can start to grasp things I previously saw as unapproachable. Learning about tensor products, complex vector spaces, and dirac notation was very interesting. I also found it refreshing to learn about how the math I am practicing can be applied to the real world in useful ways. It is always nice to be reminded of the importance of what I study.

If I had more time or were to do this project again I think I may have taken care to develop a deeper level of conceptual depth than I was able to display here. Due to time constraints I felt as though I had to restrain myself from focusing too much on the physics or computational theory aspects. I still don't quite understand how qubits are supposed to be used in computation or how quantum mechanics makes a more powerful computer. Nonetheless this project was intriguing and I appreciate the opportunity to apply the curriculum outside of the classroom.

References

[1] Kevin Resch. Mach-zehnder interferomter. *University of Waterloo IQC*, Jun 2009.

[2] Dave Bacon. One quibit, two qubit. *CSE 599d - Quantum Computing*, Jan 2006.

[3] Dave Bacon. Dirac notation and basic linear algebra for quantum computing. *CSE 599d - Quantum Computing*, Jan 2006.

[4] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London. A. Mathematical and Physical Sciences*, 400(1818):97–117, 1985.

[5] Stephan Gulde, Mark Riebe, Gavin PT Lancaster, Christoph Becher, Jürgen Eschner, Hartmut Häffner, Ferdinand Schmidt-Kaler, Isaac L Chuang, and Rainer Blatt. Implementation of the deutsch–jozsa algorithm on an ion-trap quantum computer. *Nature*, 421(6918):48, 2003.