Matrices in Quantum Computing

A 2-qubit Entangler Circuit

Huan Q. Bui

Colby College CLAS 2019

PHYSICS & MATHEMATICS
Statistics

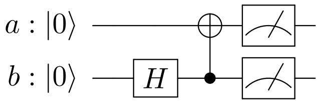
Class of 2021

April 29, 2019

Contents

1 Introduction

Quantum computing and quantum information science are rapidly growing fields in recent years. Progress is made everyday in not only in the physics labs but also in the mathematical foundation, most of which is linear algebra. The matrix theory behind quantum computing is voluminous, and thus it would be insane for me to talk about all kinds of matrices in quantum computing. So today, I would like to draw your attention to one of the simplest and basic yet interesting circuits in all of quantum computing: the 2-qubit entanglement circuit, and some of the matrix theory behind it.



2 Presentation layout

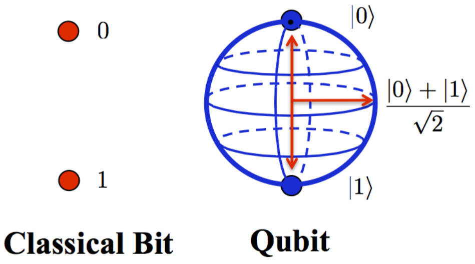
Here's my plan for this presentation. I will

1. first provide some background on quantum computing. I will introduce some of the basic objects in quantum computing such as qubits, quantum circuits, quantum gates, and entanglement.
2. And as I introduce these objects, the mathematical concepts of the Kronecker product and tensor product will come up, as they are the building blocks of the theory of quantum computation. I will talk about what these matrix theories are, how they work, and, time permitting, "why" they are required.
3. Then, we will walkthrough the 2-qubit entanglement circuit that I have just shown. We will see how the tensor product appears and how we can use it "see" what the circuit actually does.
4. Finally, I will show how you can actually implement this circuit (or any other circuit) on your browser and have an actual quantum computer at IBM execute the commands.

3 Qubits

So let us start with qubits. Qubits are kind of like classical bits, but not really. Classical bits can take values of only 0 and 1, "on" or "off", "up" or "down," while quantum bits can take any "superposition" of those two states, since qubits are quantum mechanical objects.

Most of the time, qubits are individual atoms that can be excited to the excited state or de-excited into the ground state. Mathematically, we can think



of these states as forming a basis for the space of all possible states of the atom, and the state of a qubit can be represented by the following linear combination:

$$a \begin{bmatrix} 1 \\ 0 \end{bmatrix} + b \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ represents the ground state, and $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ represents the excited state. The coefficients a, b are complex numbers such that the sum of the modulus squares is one:

$$|a|^2 + |b|^2 = 1,$$

the reason being, in quantum mechanics, it is require that the magnitude of the state vector is 1. In fact, in physics, it is often interpreted that $|a|^2$ is the probability of the qubit being in the ground state, and $|b|$ is the probability of the qubit being in the excited state.

Now, before we move on, for the sake of convenience, let us introduce a new notation for the basis states of a qubit. Let $|0\rangle$ represent the vector $\begin{bmatrix} 1 & 0 \end{bmatrix}^\top$ and $|1\rangle$ represent the vector $\begin{bmatrix} 0 & 1 \end{bmatrix}^\top$. The reason for doing this is that now we can just call the ground state the "zero" state and excited state the "one" state, giving us a nice way to relate the basis values of a classical bit. So, the state of single qubit can be written as

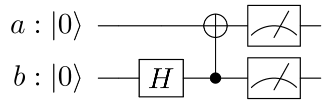
$$a\,|0\rangle + b\,|1\rangle\,.$$

4 Quantum gates

Another fundamental object in quantum computing is quantum gates. If we think of states of qubits as vectors, then quantum gates are modeled by linear transformations - matrices - on the state vectors of qubits. For example, one of the most common and important single-qubit quantum gates, is called the Hadamaar gate, which is in matrix form is

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

This gate is actually the first component on the quantum circuit we have seen:



The Hadamaar gate is denoted with an H. We can see what H does to, say $|0\rangle$:

$$H|0\rangle = H\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix},$$

which we can equivalently write as

$$\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{2}|1\rangle .$$

So what the Hadamaar gate does is putting a pure state into a perfect superposition, with exactly 50% chance of finding the qubit either state $|0\rangle$ or $|1\rangle$.

5 Multiple qubits

Like classical circuits, quantum circuits require multiple qubits to operate. And so a natural question to ask now is how do we represent the quantum state of 2-qubits with the same vector. Let us say that the qubit 1 has state

$$a|0\rangle + b|1\rangle = \begin{bmatrix}a\\b\end{bmatrix}$$

and qubit 2 has state

$$c|0\rangle + d|1\rangle = \begin{bmatrix}c\\d\end{bmatrix}.$$

It is clear that we can neither add, nor "multiply" these "pairs" in any traditional way in order to get a vector of unit length. It turns out that if we do the following seemingly "magical" multiplication procedure,

$$\begin{bmatrix}a\\b\end{bmatrix} \boxtimes \begin{bmatrix}c\\d\end{bmatrix} = \begin{bmatrix}a\begin{bmatrix}c\\d\end{bmatrix}\\ b\begin{bmatrix}c\\d\end{bmatrix}\end{bmatrix} = \begin{bmatrix}ac\\ad\\bc\\bd\end{bmatrix}.$$

We can check quite easily that this new vector has unit length. Applying this procedure to the basis states $|0\rangle$, $|1\rangle$, we actually get a basis for the vector space that contains all possible quantum states of two qubits:

$$|0\rangle \boxtimes |0\rangle = \begin{bmatrix}1\\0\\0\\0\end{bmatrix}, |0\rangle \boxtimes |1\rangle = \begin{bmatrix}0\\1\\0\\0\end{bmatrix}, |1\rangle \boxtimes |0\rangle = \begin{bmatrix}0\\0\\1\\0\end{bmatrix}, |1\rangle \boxtimes |1\rangle = \begin{bmatrix}0\\0\\0\\1\end{bmatrix},$$

and so it is convenient for us to write

$$\begin{aligned}|00\rangle &= |0\rangle \boxtimes |0\rangle\\ |01\rangle &= |0\rangle \boxtimes |1\rangle\\ |10\rangle &= |1\rangle \boxtimes |0\rangle\\ |11\rangle &= |1\rangle \boxtimes |1\rangle\end{aligned}$$

where the first slot indicates the (pure) state of the first qubit, and the second slot indicates the (pure) state of the second qubit. And so it is not so hard to see that

$$\begin{bmatrix}a\\b\end{bmatrix} \boxtimes \begin{bmatrix}c\\d\end{bmatrix} = ac\,|00\rangle + ad\,|01\rangle + bc\,|10\rangle + bd\,|11\rangle\,.$$

6 Elementariness & Entanglement

With this, we can see that given two qubits with defined states, this "product" always gives us a new two-qubit joint state, and we call this joint state **elementary**. Now, we might wonder whether any joint state can be expressed as this special product of two single-qubit states. It turns out that the answer is no.

This is a subtle fact. So, let us consider an analogue to this, with polynomials. Let us consider $p(x)$ a polynomial in x and $q(y)$ a polynomial in y. It is clear that $p(x) \cdot q(y)$ is a polynomial of both x and y. However, this polynomial with both x and y,

$$xy + 1,$$

cannot be written as a product of a polynomial in x and one in y.

Back to qubits. Consider the following example. This joint state

$$\frac{1}{\sqrt{2}}\begin{bmatrix}1\\0\\0\\1\end{bmatrix} = \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle$$

while can written as a linear combination of products of basis states, cannot be written as a single product of two single-qubit states. These qubits are now in a state we call **entanglement**.

Why this terminology? If we stare at the state vector above for a bit, we can see that the two-qubit is either in the state $\ker 00$ or $|11\rangle$ with equal probability. So we can make a measurement on the first qubit, for example, and see $|0\rangle$, then we immediately know with high very high confidence that the state of the second qubit is $|0\rangle$ without measuring it. And because this feels very much like the two qubits are somehow "coordinating" they are said to be "entangled."

7 Kronecker product

Now that we are able to represent a system of qubits with a single vector. What about matrices that act on these vectors? It turns out that if $\mathcal{A}$ is some matrix that acts on the first qubit, and $\mathcal{B}$ is some matrix that acts on the second qubit, then the corresponding matrix that acts on the two-qubit state is given by

$$\mathcal{A}\left|a\right\rangle \boxtimes \mathcal{B}\left|b\right\rangle = (\mathcal{A}\otimes\mathcal{B})(\left|a\right\rangle \boxtimes \left|b\right\rangle)$$

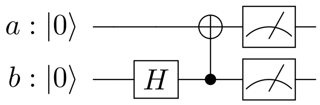
where "$\otimes$" is called the **Kronecker product**. And the way the Kronecker product is defined is kind of similar to how the magical product "$\boxtimes$" was defined before. The Kronecker product of two matrices is defined as follows. If we are given two matrices

$$\mathcal{A} = \begin{bmatrix} m & n \\ o & p \end{bmatrix} \qquad \mathcal{B} = \begin{bmatrix} q & r & s \\ t & u & v \\ w & x & y \end{bmatrix}$$

then the Kronecker product of $\mathcal{A}$ and $\mathcal{B}$ is a 6×6 matrix:

$$\mathcal{A}\otimes\mathcal{B} = \begin{bmatrix} m\begin{bmatrix} q & r & s \\ t & u & v \\ w & x & y \end{bmatrix} & n\begin{bmatrix} q & r & s \\ t & u & v \\ w & x & y \end{bmatrix} \\ o\begin{bmatrix} q & r & s \\ t & u & v \\ w & x & y \end{bmatrix} & p\begin{bmatrix} q & r & s \\ t & u & v \\ w & x & y \end{bmatrix} \end{bmatrix} = \begin{bmatrix} mq & mr & ms & nq & nr & ns \\ mt & mu & mv & nt & nu & nv \\ mw & mx & ms & nw & nx & ny \\ oq & or & os & pq & pr & ps \\ ot & ou & ov & pt & pu & pv \\ ow & ox & oy & pw & px & pys \end{bmatrix}$$

We can look back at our circuit diagram:



and check that this definition of the Kronecker product makes sense, i.e., the left hand side where we have single-qubit gates acting on individual qubits:

$$\begin{aligned} I\left|0\right\rangle \boxtimes H\left|1\right\rangle &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 \\ 0 \end{bmatrix} \boxtimes \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}\begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} \boxtimes \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \end{bmatrix} \\ &= \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \end{aligned}$$

is equal to the right hand side, where we have a two-qubit gate acting on a two-qubit state vector:

$$(I\otimes H)(\left|0\right\rangle \boxtimes \left|0\right\rangle) = \begin{bmatrix} \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} & \mathcal{O} \\ \mathcal{O} & \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \end{bmatrix}\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}.$$

8 Some properties & Elementariness

So it appears that the $\otimes$ product is very similar to the $\boxtimes$ product. In fact, we can notice a few things:

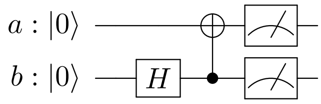
1. Both are **bilinear**, i.e., linear in both the first and second argument.
2. Both are **distributive**, left and right. We in fact have seen these properties before when we wrote

$$\begin{bmatrix} a \\ b \end{bmatrix} \boxtimes \begin{bmatrix} c \\ d \end{bmatrix} = ac\,|00\rangle + ad\,|01\rangle + bc\,|10\rangle + bd\,|11\rangle\,.$$

   By tweaking the left hand side, the bilinear and distributive properties become very clear:

$$(a\,|0\rangle + b\,|1\rangle) \boxtimes (c\,|0\rangle + d\,|1\rangle) = ac\,|00\rangle + ad\,|01\rangle + bc\,|10\rangle + bd\,|11\rangle\,.$$

3. Both are **associative**, which can easily be checked.
4. Both are **NOT commutative**. For example, $|01\rangle \neq |10\rangle$.
5. It also turns out that the business with elementariness also carries over into Kronecker products of matrices. What I mean is that, say, given two 2×2 matrices, their Kronecker product is a 4×4 matrix (which is what we just did). But not every 4×4 matrix can be written as a Kronecker product of two 2×2 matrices. An example can be readily found in our quantum circuit:



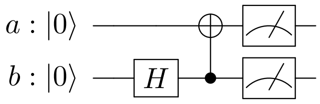
   The last gate before the measurements is called the *Control* – *NOT* gate. It is a conditional bit-flip gate, conditioned on the second qubit. In matrix form,

$$CNOT_b = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \qquad \longrightarrow \begin{cases} |00\rangle \to |00\rangle \\ |10\rangle \to |10\rangle \\ |01\rangle \to |11\rangle \\ |11\rangle \to |01\rangle \end{cases}$$

   Quantum gates that are not elementary are also called "entangled."

9 Entanglement circuit

So, we look back at the quantum circuit



we realize that we have seen how every component works. We don't have to worry so much about what the measurement gates. Technically speaking, they are not exactly quantum gates, since what they do is projecting the state vector onto one of the basis states $|0\rangle$ or $|1\rangle$. Nevertheless, since we already know how each quantum gate in this circuit functions, we can sort of work through the circuit and see why this circuit does what it does - entangling two qubits.

1. First, the identity gate doesn't do anything to the first qubit. But the second qubit is subjected to the Hadamard which puts it in a perfect superposition. So we have

$$\begin{aligned} a &: |0\rangle \to |0\rangle \\ b &: |0\rangle \to \frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\,. \end{aligned}$$

2. Then, the $CNOT$ gate acts on the two-qubit state:

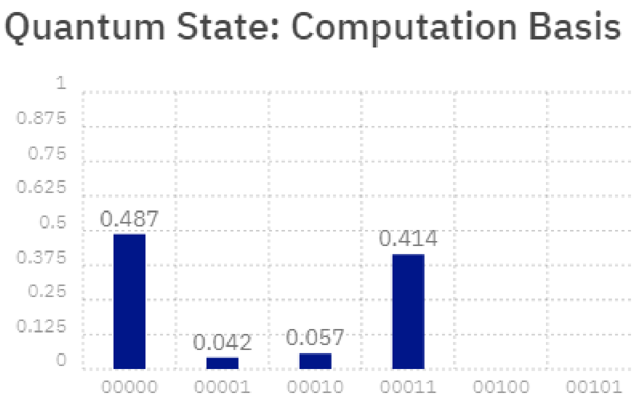
$$CNOT_b\left[|0\rangle \boxtimes \left(\frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle\right)\right] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

which is exactly what we have said before:

$$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}}\,|00\rangle + \frac{1}{\sqrt{2}}\,|11\rangle\,,$$

and entangled state.

There's a website by IBM called the IBM-Q experience that lets us upload/design a quantum circuit and have an actual quantum computer at IBM execute the circuit. Here is what I got when I ran this circuit: As you can see, the probabilities of measuring $|00\rangle$ and $|11\rangle$ are roughly equal. One thing to notice is that there are errors, which is due to the quantum mechanical - probabilistic nature of the computation.



10 Why quantum computers?

At this point, it would be natural to question the fidelity of quantum computations. Are answers reliable? And why quantum computation? How is quantum computation "better" than classical computation with bits and deterministic processes? The main advantage of quantum computation is resources. Because one qubit can store much more information than a single classical bit can due to its quantum mechanical-probabilistic nature, quantum computers can do certain things much faster and more efficiently than classical computers can such as factoring numbers or optimization problems or where quantum parallelism proves superior.

11 Tensor Products

Finally, there is one more piece to my talk, that is the connection between the "magical product" denoted as $\boxtimes$ and the Kronecker product, $\otimes$. As you might have guessed, or had an intuitive feel for, the $\boxtimes$ and $\otimes$ are very much the same - because they are! In fact, they are instances, or representations, of what's called the **tensor product**. So, the notation $\boxtimes$ is just my way of not confusing myself and you guys. In reality, the tensor product is denoted by $\otimes$.

So why exactly do we use the tensor product to describe the state of multiple qubits and what exactly is the tensor product? The answer to the first question is not so satisfying: because the tensor product is one that works, and that it is postulated in quantum mechanics that the joint state of two quantum states *is* the tensor product of the two individual quantum states.

The answer to the second question is a little more subtle and interesting. Like I said, the Kronecker product and $\boxtimes$ are instances of the tensor product. Kronecker products are special kind of product on matrices, same for $\boxtimes$, since we can always think about a vector as a very tall and very narrow matrix. On the other hand, the tensor product is one of vector spaces and of linear functions on vector spaces in general.

$$\begin{tikzcd}
\mathbf{V} \otimes \mathbf{W} \arrow[r, "\hat{f}"] \arrow[r, "linear"', phantom, shift right=1ex] & \mathbf{X} \\
\mathbf{V} \times \mathbf{W} \arrow[u, hook, "\phi"] \arrow[ur, "f"] \arrow[ur, "bilinear"', phantom, shift right=1ex] &
\end{tikzcd}$$

Roughly speaking, the tensor product of two vector spaces is a new vector space. But the tensor product is constructed in such a way that if somebody gives me a bilinear function from the Cartesian product space to some target space, then I immediately have a **unique** corresponding linear function from the tensor product space to the same target space.

Since we are working with operators, we can consider the case where $\mathbf{X}$ is $\mathbf{V} \otimes \mathbf{W}$ itself. Let's say we given an operator $\mathcal{L}$ on $\mathbf{V}$ and $\mathcal{M}$ on $\mathbf{W}$.

$$\begin{tikzcd}
\mathbf{V} \otimes \mathbf{W} \arrow[r, "\hat{f}"] \arrow[r, "linear"', phantom, shift right=1ex] & \mathbf{V} \otimes \mathbf{W} \\
\mathbf{V} \times \mathbf{W} \arrow[u, hook, "\phi"] \arrow[ur, "f"] \arrow[ur, "bilinear"', phantom, shift right=1ex] &
\end{tikzcd}$$

We know that $\mathcal{L} \otimes \mathcal{M}$ is an operator on $\mathbf{V} \otimes \mathbf{W}$, and that $\mathcal{L}[v] \otimes \mathcal{M}[w]$ is a bilinear function from $\mathbf{V} \times \mathbf{W}$ to $\mathbf{V} \otimes \mathbf{W}$. And so by the uniqueness in the definition, it must be true that

$$(\mathcal{L} \otimes \mathcal{M})(v \otimes w) = \mathcal{L}[v] \otimes \mathcal{M}[w].$$

This looks every much like how the Kronecker product is defined, as except here $\mathcal{L}$, $\mathcal{M}$ are general linear functions, not matrices. And while there is not a general definition for how to compute the tensor product of two linear functions, we don't have to worry so much because if possible, we can always choose a basis ν for $\mathbf{V}$ and ω for $\mathbf{W}$ with respect to which $\mathcal{L}$ and $\mathcal{M}$ can be represented by matrices. And so the matrix of $\mathcal{L} \otimes \mathcal{M}$ with respect to a basis τ of $\mathbf{V} \otimes \mathbf{W}$ created by tensor products of the elements of ν, ω can be calculated via the Kronecker product.

$$[\mathcal{L} \otimes \mathcal{M}]_{\tau \leftarrow \tau} = [\mathcal{L}]_{\nu \leftarrow \nu} \otimes [\mathcal{M}]_{\omega \leftarrow \omega}.$$

And so we're done.

12 Recap

And so, just to summarize, in this talk we have learned a few things from trying to figure out how a 2-qubit entanglement circuit works. We have seen what qubits and quantum gates are, and how the Kronecker product and matrix representations of these objects allow us to model a quantum computation and calculate numerical results. We also learn about entanglement and how entanglement manifests in the language of matrix theory. Finally, we look at a more general concept of the Kronecker product on matrices, called the tensor product.