Roby Mann
Contemporary Physics Fall 2019
Professor Morgan

How to Best Arrange Guests at a Dinner Party

Section 1: History of Quantum Physics and Early Quantum Computing

Like many people, my first interaction with quantum physics began with the story of Schrodinger's cat - a nice story that illustrates two core quantum topics, the uncertainty principle and the collapse of the wavefunction (although further study of the subject is required to understand that these two topics are being presented). For many years, I confused quantum physics with something relating to time travel, wormholes, or nuclear explosions, some of the wackiest things in the universe. Only as a sophomore in college did I finally begin to refine my understanding of the subject and formalize it. It was at this point that I was introduced to quantum computing, the harnessing of quantum physics to perform computation in ways that are very different from classical computing.

In this section I will give a short history of quantum physics and quantum computing to lay a groundwork for the next sections, where I will cover some theoretical and applied aspects of quantum computing. First let us look at the origins of quantum physics.

Quantum theory takes over when classical newtonian physics breaks down at extremely small scales. Quantum theory offers the most compelling understanding of atoms and subatomic particles, and how they interact with each other. It begins with the assumption that the universe is discrete - that there is a smallest length, mass, time, temperature, and charge, the Planck units (some other physics theories allow for a continuous universe). From this, quantum physics begins to formalize how we talk about small objects, such as electrons. The central idea of quantum physics is that the state of these objects is not static and fixed, instead of being a particle or wave, most objects in the quantum world exhibit properties of both particles *and* waves. One property of wave-like objects is the uncertainty principle, which states that for certain observable qualities (like position and momentum), there is a limit to how precisely one can measure these qualities - with respect to each other. For example, an object with a very uniform wavelength measured over a long period of time has almost no distinct position (large Δx, meaning large variability of position) in space. But it does have a very well defined momentum (small Δp, meaning small variability of momentum). The opposite would be true if the object had a very sharp peak in wavelength at one location (small Δx) but zero wavelength elsewhere (large Δp). This object would have well defined positions and not well defined momentum.

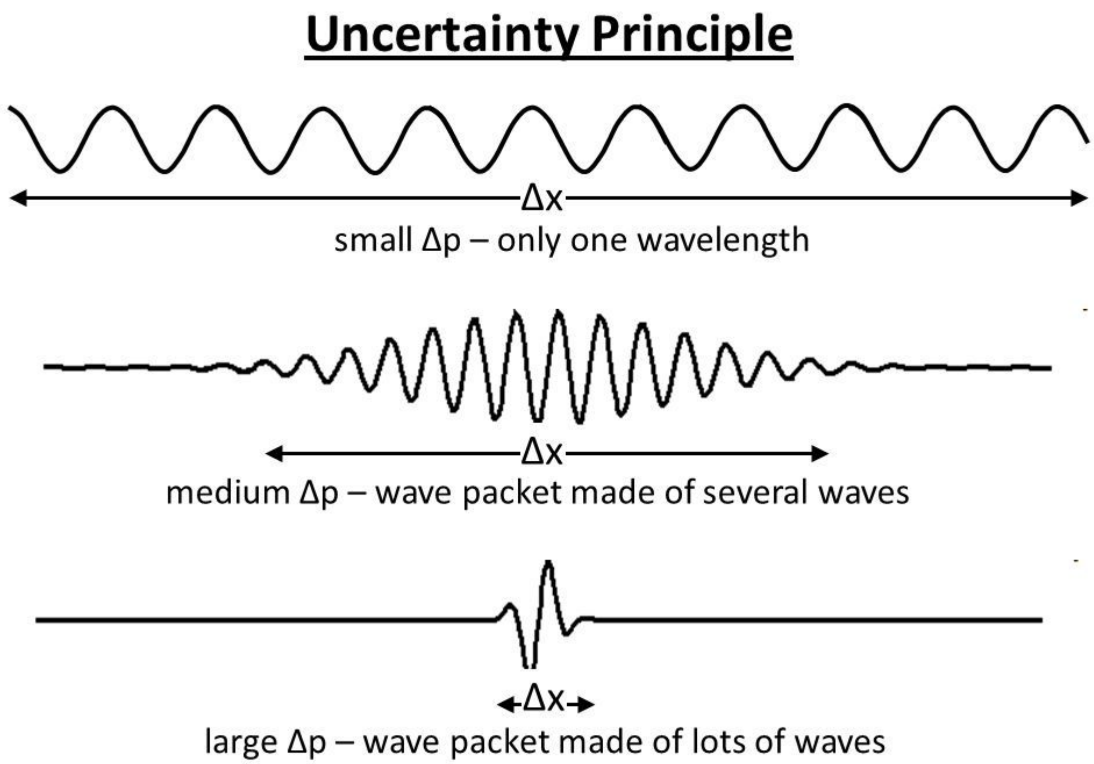

Figure 1: Graphical representation of uncertainty principle for momentum and position[1]

This property, the uncertainty principle, is one of many quantum behaviors that deviate from classical behaviors. Another key concept in quantum physics is superposition, the fact that an object can exist in a combination of classical states. For example, let's suppose you could create a classical electron (bear with me here - I know that this cannot exist, but for the purpose of this paper, let's assume it can). This classical electron could have two possible states, up or down. You can at any point in time measure its state and you would find that the electron exists completely in one of the two states. A quantum electron can exist in a superposition of states, meaning that the electron can be a little bit in the up state and a little bit in the down state. Mathematically, we represent the “up” state as $|1\rangle$, and the “down” state as $|0\rangle$, using dirac notation. A superposition then, is $\alpha|0\rangle + \beta|1\rangle$, where $\alpha, \beta \in$ C. α and β are coefficients that indicate how much of the total system is in state 0 or state 1, and $|\alpha|^2 + |\beta|^2 = 1$.

However, when you go to measure the state of that (quantum) electron, the wavefunction - the mathematics responsible for its particular behavior and orientation, collapse from a superposition of states to a single classical state. This fact makes measuring quantum systems challenging. We can’t simply take a photograph or slow the process down to have a look. Any external stimuli that is introduced into the system can have tremendous impact. The systems that are of interest to us are much more sensitive to slight fluctuations in their environment that their classical counterparts. Therefore we have to go to extreme measures to remove possible external stimuli.

Here begins our venture into the land of quantum computing. For many years, creating a quantum computer was impossible, since it requires cooling technology that we did not possess. The hotter a quantum system is, the more kinetic and vibrational energy it has. At a certain point, if the system is too hot, it becomes impossible to source information from. This quality is called decoherence, and it is a core issue of quantum computing. When handling these systems, any bump in energy can cause the quantum object (qubit) to abruptly change state, resulting in a total loss of the solution. Even after accounting for as many variables as possible, the quantum objects have a certain time interval after which they will decohere, or lose their information. In order to create stable quantum systems we had to figure out ways to cool a relatively large area to sub 1 degree Kelvin, create near perfect vacuums, and shield the processors from outside radiation. The most recent creation from D-Wave Systems, a Canadian company that manufactures “commercial” quantum computers (I say “commercial” with quotation marks because one of these puppies will set you back a cool 15 million dollars), cools their processor to 0.015K, has 50,000 times less radiation than earth’s magnetic field, and operates at a pressure 10 billion times lower than our atmosphere. Only under these extreme conditions can the quantum processes occur.

Section 2: Quantum Computing and Theory

I will shortly get into how quantum computers process information, but first let's talk about the basic structures of a quantum computer. In a classical computer we process information with bits, the movement of electricity through wires that can either be on or off, representing 0's and 1's. In the quantum world we have quantum bits, or qubits. These structures can represent both 0 and 1 at the same time, through superposition. How do we create a qubit? Fundamentally, a qubit is built from any system that can exist in a superposition of two states. You could make a qubit out of a dining room table if that table could be both right side up and up side down (that would make for interesting dinner parties, eh?). Traditionally, the qubit is either an atom or electron (where the 0 and 1 states are the nuclear or electron spin), or a quantum Josephson junction (although many other types of qubits do exist). There are pros and cons to each method. A single electron has a much longer decoherence time, around 60ms in ideal conditions[4], than other means of quantum computing. However this electron also requires a completely neutral background environment to exist in. Scientists have embedded a single phosphorus atom in a mass of purified silicon 28, a magnetically neutral isotope of silicon[5]. This phosphorus atom has a lone electron in its outermost valence shell. This electron is then acted on by superconducting magnets that can directly control the spin of that electron. As one brash commenter on a YouTube video[6] rightfully said, "that's a lot of work for one f-ing bit"! For Josephson junctions, less space is required, but they have their own limitations. A Josephson junction is a thin layer of superconducting material sandwiched between two larger layers of insulating material. I wish I better understood the electricity and magnetism physics behind these phenomena, but that is beyond the scope of this paper. For our purposes, the Josephson junction acts as qubit because a magnetic flux through a loop generates either a clockwise or counterclockwise current (or both at the same time - superposition!). The decoherence time is much shorter however, around 30-40 μs. Using Josephson junctions, D-Wave is able to construct an entire (high qubit count) quantum computer that can fit in a small room - a feat that would be very challenging to achieve via other means of quantum computing.

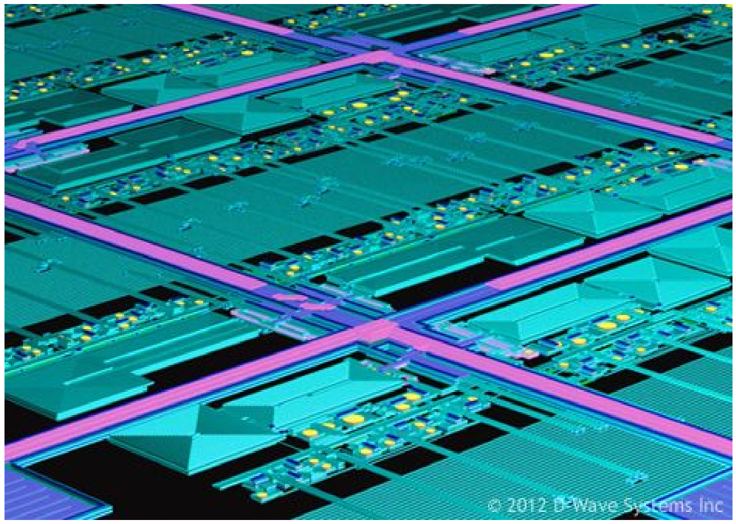

Figure 2: C.A.D. schematic of D-Wave's quantum chipset. The yellow dots are Josephson junctions[2]

When we have multiple qubits interacting with each other, we see an astonishing result. Due to their superposition nature, each qubit can represent an entire sphere of values (referred to as a bloch sphere), and thus there is an uncountably infinite number of possible values that α and β can assume. (If α and β are the angles Θ and ϕ, in spherical coordinates, then with only those two numbers you can specify any location on the surface of a unit sphere.)

In my research I found that the most abundant and accessible source of information on quantum computing was the YouTube channel hosted by D-Wave Systems (just to reiterate, D-Wave is a company that builds quantum computers!). They have over 90 videos covering topics from coding on quantum computers, to tours of their facilities, to fundamental quantum theory. Most of my understanding of the subject is informed by their videos. The style of quantum computing that they specialize in is quantum annealing, others styles include quantum gate modeling, and universal quantum computers.

The goal of quantum annealing is to solve two types of problems, optimization and probabilistic sampling problems. Optimization, for example, is quite straightforward and easy to understand. Say you are trying to seat guests at a dinner party and you want to find the best arrangement of 10 guests. Since there are 10 factorial ways to arrange these guests, finding the single best option is pretty tricky. So how do we express this problem as a physics problem? We can employ a technique called energy minimization, where the goal is to set up the problem as an energy distribution (think of some complicated potential energy graph). If the computer is programmed correctly, the smallest energy value will correspond to the most correct solution of a given problem. Quantum computers are significantly better at solving these types of problems than classical computers.

Alright - I've blabbed on long enough about how a hypothetical quantum computer would solve these problems, but how do real quantum computers do it? In quantum annealing, we have no control over the state of qubits, so after we set them up, the system evolves in time until it reaches its final state. The process starts with the setup. First, each individual qubit is prepared in a specific superposition. Next, entanglement is introduced. Unfortunately I haven't quite had the time to explore entanglement as much as I would have liked to, but in short, two particles can be entangled, and afterwards, their combined state cannot be described by measurements made on only one of the particles. Once entanglement is introduced, the entire group of qubits are physically connected - their individual states are dependent on the states of others. Next, the system experiences a bias, or a shift in the potential landscape that is energetically favorable to one region versus another. If you had a square well system (zero potential between $0 < x < a$, and an infinite potential at $x = 0$ and $x = a$), you could apply a linear shift to that well, creating an imbalance of potential. A particle trapped in that well would favor being in the region of lower potential energy. Finally, the system has been fully prepared and is ready to be run. Now, we wait. Literally. In quantum annealing, the entangled qubits are left alone to evolve in their environment until they each come to rest in a classical state[7]. This process is very well documented and described in D-Wave's YouTube videos, but in essence, during the anneal (over a specified time interval), the qubits will be free to modulate between states and shift around in their configurations. At the end, once the qubits have decohered and have returned to classical states, the particular configuration should be the most correct answer. D-Wave has reported that 95% of their tests return the correct result. Not too shabby.

In quantum gate modeling, we do have control over the state of the qubits. By applying a perturbation to the system, in the form of electromagnetic radiation, we can excite the qubit at the specific frequency required to cause it to jump to the next energy level. The specific frequency is $\omega = \frac{Ea - Eb}{\hbar}$, where E_a is the higher energy level that you want to either jump to or jump from, and E_b is the lower energy level. At higher levels, we essentially have control over the qubits in the same way that classical logic gates control the flow of information between bits and bytes. We can apply certain rotations and translations to the bloch sphere, which manipulates the orientation of the qubits. Some of the more prominent gates used are in Figure 3 below.

I wish that I had more time and knowledge to continue my exploration into the various forms of quantum computing and how they work, but since I am constrained in many ways, I will move on now to discuss some applications of quantum computing and the future of quantum computers.

Operator	**Gate(s)**		**Matrix**
Pauli-X (X)	$\boxed{X}$	$\oplus$	$\begin{bmatrix}0 & 1\\1 & 0\end{bmatrix}$
Pauli-Y (Y)	$\boxed{Y}$		$\begin{bmatrix}0 & -i\\i & 0\end{bmatrix}$
Pauli-Z (Z)	$\boxed{Z}$		$\begin{bmatrix}1 & 0\\0 & -1\end{bmatrix}$
Hadamard (H)	$\boxed{H}$		$\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$
Phase (S, P)	$\boxed{S}$		$\begin{bmatrix}1 & 0\\0 & i\end{bmatrix}$
$\pi/8$ (T)	$\boxed{T}$		$\begin{bmatrix}1 & 0\\0 & e^{i\pi/4}\end{bmatrix}$
Controlled Not (CNOT, CX)	$\bullet$ / $\oplus$		$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$
Controlled Z (CZ)	$\bullet$ / $\boxed{Z}$	$\bullet$ / $\bullet$	$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&-1\end{bmatrix}$
SWAP		$\times$ / $\times$	$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$
Toffoli (CCNOT, CCX, TOFF)	$\bullet$ / $\bullet$ / $\oplus$		$\begin{bmatrix}1&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0\\0&0&1&0&0&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&1&0&0&0\\0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&1&0\end{bmatrix}$

Figure 3: Prominent gates used in quantum gate model computing

Section 3: Applications of Quantum Computing

Quantum computers are not a replacement for classical computers. Many tasks that we require computers to perform on a daily basis are not suited for quantum machines. For example, browsing the web, playing games, or watching videos are all tasks that are well suited for classical computers. A quantum computer would be either just as efficient, or less efficient. But there are many things that quantum computers can do much faster than classical ones. For example, given a very large number, N, find its prime factors. Classically, we would have to check a random number, a, for its greatest common divisor with N. If it returns a value different than 1, then we have found the non-trivial factor of N, and we are done! But if it returns 1, then we have not found a proper factor and we essentially repeat the process (and believe me, it takes a long while). In a quantum computer, we have shor's algorithm, an extremely powerful tool that allows us to make better than random guesses, much better in fact. Shor's algorithm states that from our guess g, the next guess of $g^{p/2} \pm 1$ will be much better (by much better I mean much closer to the answer we are looking for). The difficulty then is finding this p. Unfortunately I could not find an explanation of this p calculation that was not over my head, but in short - it's very easy for quantum computers to find this p value. You might be asking yourself, sure we can find the prime factors of large integers, but what does that do for us? It just so happens that a common form of encryption, used for very sensitive data such as communication messages, bank information, locations, etc, uses large integer factorization as their encryption method. It's a good thing those quantum computers are so expensive, otherwise hackers would have a hay-day! In reality, there are other forms of encryption and our quantum technologies are not yet ready to go about hacking into any and all encryptions, but we are making great strides towards more powerful and capable machines. Recently, Google's quantum computer named Sycamore performed a calculation in 200 seconds that would take the world's most powerful classical supercomputer 2.5 days to solve! This is 1000 times faster than classical computing, and that's not even the upper bound of speed. So no, sadly, the future will not be little tiny quantum computers in our pockets, but we will have large quantum computers to calculate how to best arrange all the guests at your next dinner party, so that there is no bickering and everyone has a nice time.

Figures:

1. https://www.reddit.com/r/askscience/comments/789ptn/how_do_you_interpret_heisenbergs_uncertainty/
2. https://www.dwavesys.com/tutorials/background-reading-series/introduction-d-wave-quantum-hardware
3. https://en.wikipedia.org/wiki/Quantum_logic_gate#/media/File:Quantum_Logic_Gates.png

Works Cited:

4. https://en.wikipedia.org/wiki/Transmon
5. https://arxiv.org/ftp/cond-mat/papers/0512/0512705.pdf
6. https://www.youtube.com/watch?v=zNzzGgr2mhk
7. https://www.youtube.com/watch?v=tnikftltqE0

Works Consulted:

8. https://www.youtube.com/user/dwavesystems
9. https://en.wikipedia.org/wiki/Quantum_computing