18
Quantum computing and cryptography I

> *"I think I can safely say that nobody understands quantum mechanics."* , Richard Feynman, 1965

> *"The only difference between a probabilistic classical world and the equations of the quantum world is that somehow or other it appears as if the probabilities would have to go negative"*, Richard Feynman, 1982

There were two schools of natural philosophy in ancient Greece. *Aristotle* believed that objects have an *essence* that explains their behavior, and a theory of the natural world has to refer to the *reasons* (or "final cause" to use Aristotle's language) as to why they exhibit certain phenomena. *Democritus* believed in a purely mechanistic explanation of the world. In his view, the universe was ultimately composed of elementary particles (or *Atoms*) and our observed phenomena arise from the interactions between these particles according to some local rules. Modern science (arguably starting with Newton) has embraced Democritus' point of view, of a mechanistic or "clockwork" universe of particles and forces acting upon them.

While the classification of particles and forces evolved with time, to a large extent the "big picture" has not changed from Newton till Einstein. In particular it was held as an axiom that if we knew fully the current *state* of the universe (i.e., the particles and their properties such as location and velocity) then we could predict its future state at any point in time. In computational language, in all these theories the state of a system with n particles could be stored in an array of $O(n)$ numbers, and predicting the evolution of the system can be done by running some efficient (e.g., $poly(n)$ time) deterministic computation on this array.

18.1 THE DOUBLE SLIT EXPERIMENT

Alas, in the beginning of the 20th century, several experimental results were calling into question this "clockwork" or "billiard ball" theory of the world. One such experiment is the famous double slit experiment. Here is one way to describe it. Suppose that we buy one of those baseball pitching machines, and aim it at a soft plastic wall, but put a *metal barrier with a single slit* between the machine and the plastic wall (see Fig. 18.1). If we shoot baseballs at the plastic wall, then some of the baseballs would bounce off the metal barrier, while some would make it through the slit and dent the wall. If we now carve out an additional slit in the metal barrier then more balls would get through, and so the plastic wall would be *even more dented*.

So far this is pure common sense, and it is indeed (to my knowledge) an accurate description of what happens when we shoot baseballs at a plastic wall. However, this is not the same when we shoot *photons*. Amazingly, if we shoot with a "photon gun" (i.e., a laser) at a wall equipped with photon detectors through some barrier, then (as shown in Fig. 18.2) in some positions of the wall we will see *fewer* hits when the two slits are open than one only ones of them is!.[1] In particular there are positions in the wall that are hit when the first slit is open, hit when the second gun is open, but are *not hit at all when both slits are open!*.

It seems as if each photon coming out of the gun is aware of the global setup of the experiment, and behaves differently if two slits are open than if only one is. If we try to "catch the photon in the act" and place a detector right next to each slit so we can see exactly the path each photon takes then something even more bizarre happens. The mere fact that we *measure* the path changes the photon's behavior, and now this "destructive interference" pattern is gone and the number of times a position is hit when two slits are open is the sum of the number of times it is hit when each slit is open.

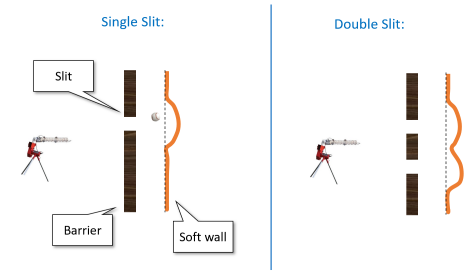

Figure 18.1: In the "double baseball experiment" we shoot baseballs from a gun at a soft wall through a hard barrier that has one or two slits open in it. There is only "constructive interference" in the sense that the dent in each position in the wall when both slits are open is the sum of the dents when each slit is open on its own.

[1] A nice illustrated description of the double slit experiment appears in this video.

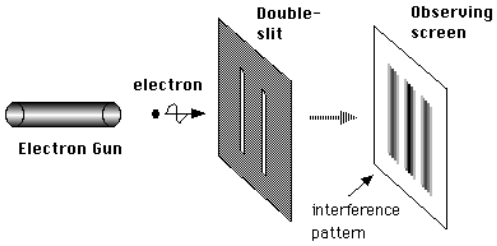

Figure 18.2: The setup of the double slit experiment in the case of photon or electron guns. We see also *destructive* interference in the sense that there are some positions on the wall that get *fewer* hits when both slits are open than they get when only one of the slits is open. Image credit: Wikipedia.

> P You should read the paragraphs above more than once and make sure you appreciate how truly mind boggling these results are.

18.2 QUANTUM AMPLITUDES

The double slit and other experiments ultimately forced scientists to accept a very counterintuitive picture of the world. It is not merely about nature being randomized, but rather it is about the probabilities in some sense "going negative" and cancelling each other!

To see what we mean by this, let us go back to the baseball experiment. Suppose that the probability a ball passes through the left slit is p_L and the probability that it passes through the right slit is p_R. Then, if we shoot N balls out of each gun, we expect the wall will be hit $(p_L + p_R)N$ times. In contrast, in the quantum world of photons instead of baseballs, it can sometimes be the case that in both the first and second case the wall is hit with positive probabilities p_L and p_R respectively but somehow when both slits are open the wall (or a particular position in it) is not hit at all. It's almost as if the probabilities can "cancel each other out".

To understand the way we model this in quantum mechanics, it is helpful to think of a "lazy evaluation" approach to probability. We can think of a probabilistic experiment such as shooting a baseball through two slits in two different ways:

- When a ball is shot, "nature" tosses a coin and decides if it will go through the left slit (which happens with probability p_L), right slit (which happens with probability p_R), or bounce back. If it passes through one of the slits then it will hit the wall. Later we can look at the wall and find out whether or not this event happened, but the fact that the event happened or not is determined independently of whether or not we look at the wall.
- The other viewpoint is that when a ball is shot, "nature" computes the probabilities p_L and p_R as before, but does *not* yet "toss the coin" and determines what happened. Only when we actually look at the wall, nature tosses a coin and with probability $p_L + p_R$ ensures we see a dent. That is, nature uses "lazy evaluation", and only determines the result of a probabilistic experiment when we decide to *measure* it.

While the first scenario seems much more natural, the end result in both is the same (the wall is hit with probability $p_L + p_R$) and so the question of whether we should model nature as following the first scenario or second one seems like asking about the proverbial tree that falls in the forest with no one hearing about it.

However, when we want to describe the double slit experiment with photons rather than baseballs, it is the second scenario that lends itself better to a quantum generalization. Quantum mechanics associates a number α known as an *amplitude* with each probabilistic experiment. This number α can be *negative*, and in fact even *complex*. We never observe the amplitudes directly, since whenever we *measure* an event with amplitude α, nature tosses a coin and determines that the event happens with probability $|\alpha|^2$. However, the sign (or in the complex case, phase) of the amplitudes can affect whether two different events have *constructive* or *destructive* interference.

Specifically, consider an event that can either occur or not (e.g. "detector number 17 was hit by a photon"). In classical probability, we model this by a probability distribution over the two outcomes: a pair of non-negative numbers p and q such that $p + q = 1$, where p corresponds to the probability that the event occurs and q corresponds to the probability that the event does not occur. In quantum mechanics, we model this also by pair of numbers, which we call *amplitudes*. This is a pair of (potentially negative or even complex) numbers α and β such that $|\alpha|^2 + |\beta|^2 = 1$. The probability that the event occurs is $|\alpha|^2$ and the probability that it does not occur is $|\beta|^2$. In isolation, these negative or complex numbers don't matter much, since we anyway square them to obtain probabilities. But the interaction of positive and negative amplitudes can result in surprising *cancellations* where somehow combining two scenarios where an event happens with positive probability results in a scenario where it never does.

If you don't find the above description confusing and unintuitive, you probably didn't get it. Please make sure to re-read the above paragraphs until you are thoroughly confused.

Quantum mechanics is a mathematical theory that allows us to calculate and predict the results of the double-slit and many other experiments. If you think of quantum mechanics as an explanation as to what "really" goes on in the world, it can be rather confusing. However, if you simply "shut up and calculate" then it works amazingly well at predicting experimental results. In particular, in the double slit experiment, for any position in the wall, we can compute numbers α and β such that photons from the first and second slit hit that position with probabilities $|\alpha|^2$ and $|\beta|^2$ respectively. When we open both slits, the probability that the position will be hit is proportional to $|\alpha + \beta|^2$, and so in particular, if $\alpha = -\beta$ then it will be the case that, despite being hit when *either* slit one or slit two are open, the position is *not hit at all* when they both are. If you are confused by quantum mechanics, you are not alone: for decades people have been trying to come up with explanations for "the underlying reality" behind quantum mechanics, including Bohmian Mechanics, Many Worlds and others. However, none of these interpretations have gained universal acceptance and all of those (by design) yield the same experimental predictions. Thus at this point many scientists prefer to just ignore the question of what is the "true reality" and go back to simply "shutting up and calculating".

Some of the counterintuitive properties that arise from amplitudes or "negative probabilities" include:

- **Interference** - As we see here, probabilities can "cancel each other out".
- **Measurement** - The idea that probabilities are negative as long as "no one is looking" and "collapse" to positive probabilities when they are *measured* is deeply disturbing. Indeed, people have shown that it can yield to various strange outcomes such as "spooky actions at a distance", where we can measure an object at one place and instantaneously (faster than the speed of light) cause a difference in the results of a measurements in a place far removed. Unfortunately (or fortunately?) these strange outcomes have been confirmed experimentally.
- **Entanglement** - The notion that two parts of the system could be connected in this weird way where measuring one will affect the other is known as *quantum entanglement*.

Again, as counter-intuitive as these concepts are, they have been experimentally confirmed, so we just have to live with them.

Remark 18.1 — Complex vs real, other simplifications. If (like the author) you are a bit intimidated by complex numbers, don't worry: you can think of all amplitudes as *real* (though potentially *negative*) numbers without loss of understanding. All the "magic" of quantum computing already arises in this case, and so we will often restrict attention to real amplitudes in this chapter.

We will also only discuss so-called *pure* quantum states, and not the more general notion of *mixed* states. Pure states turn out to be sufficient for understanding the algorithmic aspects of quantum computing.

More generally, this chapter is not meant to be a complete description of quantum mechanics, quantum information theory, or quantum computing, but rather illustrate the main points where these differ from classical computing.

18.2.1 Quantum computing and computation - an executive summary.

One of the strange aspects of the quantum-mechanical picture of the world is that unlike in the billiard ball example, there is no obvious algorithm to simulate the evolution of n particles over t time periods in $poly(n,t)$ steps. In fact, the natural way to simulate n quantum particles will require a number of steps that is *exponential* in n. This is a

huge headache for scientists that actually need to do these calculations in practice.

In the 1981, physicist Richard Feynman proposed to "turn this lemon to lemonade" by making the following almost tautological observation:

> *If a physical system cannot be simulated by a computer in T steps, the system can be considered as performing a computation that would take more than T steps*

So, he asked whether one could design a quantum system such that its outcome y based on the initial condition x would be some function $y = f(x)$ such that **(a)** we don't know how to efficiently compute in any other way, and **(b)** is actually useful for something.[2] In 1985, David Deutsch formally suggested the notion of a quantum Turing machine, and the model has been since refined in works of Detusch and Josza and Bernstein and Vazirani. Such a system is now known as a *quantum computer*.

For a while these hypothetical quantum computers seemed useful for one of two things. First, to provide a general-purpose mechanism to simulate a variety of the real quantum systems that people care about. Second, as a challenge to the theory of computation's approach to model efficient computation by Turing machines, though a challenge that has little bearing to practice, given that this theoretical "extra power" of quantum computer seemed to offer little advantage in the problems people actually want to solve such as combinatorial optimization, machine learning, data structures, etc..

To a significant extent, this is still true today. We have no real evidence that quantum computers, when built, will offer truly significant[3] advantage in 99 percent of the applications of computing.[4] However, there is one cryptography-sized exception: In 1994 Peter Shor showed that quantum computers can solve the integer factoring and discrete logarithm in polynomial time. This result has captured the imagination of a great many people, and completely energized research into quantum computing.

This is both because the hardness of these particular problems provides the foundations for securing such a huge part of our communications (and these days, our economy), as well as it was a powerful demonstration that quantum computers could turn out to be useful for problems that a-priori seemed to have nothing to do with quantum physics.

At the moment there are several intensive efforts to construct large scale quantum computers. It seems safe to say that, in the next five years or so there will not be a quantum computer large enough to factor, say, a 1024 bit number. However, some quantum computers have

[2] As its title suggests, Feynman's lecture was actually focused on the other side of simulating physics with a computer, but he mentioned that as a "side remark" one could wonder if it's possible to simulate physics with a new kind of computer - a "quantum computer" which would "not [be] a Turing machine, but a machine of a different kind". As far as I know, Feynman did not suggest that such a computer could be useful for computations completely outside the domain of quantum simulation, and in fact he found the question of whether quantum mechanics could be simulated by a classical computer to be more interesting.

[3] I am using the theorist' definition of conflating "significant" with "super-polynomial". As we'll see, Grover's algorithm does offer a very generic *quadratic* advantage in computation. Whether that quadratic advantage will ever be good enough to offset in practice the significant overhead in building a quantum computer remains an open question. We also don't have evidence that super-polynomial speedups *can't* be achieved for some problems outside the Factoring/Dlog or quantum simulation domains, and there is at least one company banking on such speedups actually being feasible.

[4] This "99 percent" is a figure of speech, but not completely so. It seems that for many web servers, the TLS protocol (which based on the current non-lattice based systems would be completely broken by quantum computing) is responsible for about 1 percent of the CPU usage.

been built that achieved tasks that are either not known to be achieved classically, or at least seem to require more resources classically than they do for these quantum computers. When and if such a computer is built that can break reasonable parameters of Diffie Hellman, RSA and elliptic curve cryptography is anybody's guess. It could also be a "self destroying prophecy" whereby the existence of a small-scale quantum computer would cause everyone to shift away to lattice-based crypto which in turn will diminish the motivation to invest the huge resources needed to build a large scale quantum computer.[5]

The above summary might be all that you need to know as a cryptographer, and enough motivation to study lattice-based cryptography as we do in this course. However, because quantum computing is such a beautiful and (like cryptography) counter-intuitive concept, we will try to give at least a hint of what it is about and how Shor's algorithm works.

[5] Of course, given that "export grade" cryptography that was supposed to disappear with 1990's took a long time to die, I imagine that we'll still have products running 1024 bit RSA when everyone has a quantum laptop.

18.3 QUANTUM 101

We now present some of the basic notions in quantum information. It is very useful to contrast these notions to the setting of *probabilistic* systems and see how "negative probabilities" make a difference. This discussion is somewhat brief. The chapter on quantum computation in my book with Arora (see draft here) is one relatively short resource that contains essentially everything we discuss here. See also this blog post of Aaronson for a high level explanation of Shor's algorithm which ends with links to several more detailed expositions. See also this lecture of Aaronson for a great discussion of the feasibility of quantum computing (Aaronson's course lecture notes and the book that they spawned are fantastic reads as well).

States: We will consider a simple quantum system that includes n objects (e.g., electrons/photons/transistors/etc..) each of which can be in either an "on" or "off" state - i.e., each of them can encode a single *bit* of information, but to emphasize the "quantumness" we will call it a *qubit*. A *probability distribution* over such a system can be described as a 2^n dimensional vector v with non-negative entries summing up to 1, where for every $x \in \{0,1\}^n$, v_x denotes the probability that the system is in state x. As we mentioned, quantum mechanics allows negative (in fact even complex) probabilities and so a *quantum state* of the system can be described as a 2^n dimensional vector v such that $\|v\|^2 = \sum_x |v_x|^2 = 1$.

Measurement: Suppose that we were in the classical probabilistic setting, and that the n bits are simply random coins. Thus we can describe the *state* of the system by the 2^n-dimensional vector v such that $v_x = 2^{-n}$ for all x. If we *measure* the system and see what the coins

came out, we will get the value x with probability v_x. Naturally, if we measure the system twice we will get the same result. Thus, after we see that the coin is x, the new state of the system *collapses* to a vector v such that $v_y = 1$ if $y = x$ and $v_y = 0$ if $y \neq x$. In a quantum state, we do the same thing: if we *measure* a vector v corresponds to turning it with probability $|v_x|^2$ into a vector that has 1 on coordinate x and zero on all the other coordinates.

Operations: In the classical probabilistic setting, if we have a system in state v and we apply some function $f : \{0,1\}^n \to \{0,1\}^n$ then this transforms v to the state w such that $w_y = \sum_{x:f(x)=y} v_x$.

Another way to state this, is that $w = M_f$ where M_f is the matrix such that $M_{f(x),x} = 1$ for all x and all other entries are 0. If we toss a coin and decide with probability $1/2$ to apply f and with probability $1/2$ to apply g, this corresponds to the matrix $(1/2)M_f + (1/2)M_g$. More generally, the set of operations that we can apply can be captured as the set of convex combinations of all such matrices- this is simply the set of non-negative matrices whose columns all sum up to 1- the *stochastic* matrices. In the quantum case, the operations we can apply to a quantum state are encoded as a *unitary* matrix, which is a matrix M such that $\|Mv\| = \|v\|$ for all vectors v.

Elementary operations: Of course, even in the probabilistic setting, not every function $f : \{0,1\}^n \to \{0,1\}^n$ is efficiently computable. We think of a function as efficiently computable if it is composed of polynomially many elementary operations, that involve at most 2 or 3 bits or so (i.e., Boolean *gates*). That is, we say that a matrix M is *elementary* if it only modifies three bits. That is, M is obtained by "lifting" some 8×8 matrix M' that operates on three bits i, j, k, leaving all the rest of the bits intact. Formally, given an 8×8 matrix M' (indexed by strings in $\{0,1\}^3$) and three distinct indices $i < j < k \in \{1, \ldots, n\}$ we define the *n-lift of M' with indices i, j, k* to be the $2^n \times 2^n$ matrix M such that for every strings x and y that agree with each other on all coordinates except possibly i, j, k, $M_{x,y} = M'_{x_i x_j x_k, y_i y_j y_k}$ and otherwise $M_{x,y} = 0$. Note that if M' is of the form M'_f for some function $f : \{0,1\}^3 \to \{0,1\}^3$ then $M = M_g$ where $g : \{0,1\}^n \to \{0,1\}^n$ is defined as $g(x) = f(x_i x_j x_k)$. We define M as an *elementary stochastic matrix* or a *probabilistic gate* if M is equal to an n lift of some stochastic 8×8 matrix M'. The quantum case is similar: a *quantum gate* is a $2^n \times 2^n$ matrix that is an N lift of some unitary 8×8 matrix M'. It is an exercise to prove that lifting preserves stochasticity and unitarity. That is, every probabilistic gate is a stochastic matrix and every quantum gate is a unitary matrix.

Complexity: For every stochastic matrix M we can define its *randomized complexity*, denoted as $R(M)$ to be the minimum number T such that M is can be (approximately) obtained by combining T elementary probabilistic gates. To be concrete, we can define $R(M)$ to be the minimum T such that there exists T elementary matrices $M_1, \ldots, M_T$ such that for every x, $\sum_y |M_{y,x} - (M_T \cdots M_1)_{y,x}| < 0.1$. (It can be shown that $R(M)$ is finite and in fact at most 10^n for every M; we can do so by writing M as a convex combination of function and writing every function as a composition of functions that map a single string x to y, keeping all other inputs intact.) We will say that a probabilistic process M mapping distributions on $\{0,1\}^n$ to distributions on $\{0,1\}^n$ is *efficiently classically computable* if $R(M) \leq poly(n)$. If M is a unitary matrix, then we define the *quantum complexity* of M, denoted as $Q(M)$, to be the minimum number T such that there are quantum gates $M_1, \ldots, M_T$ satisfying that for every x, $\sum_y |M_{y,x} - (M_T \cdots M_1)_{y,x}|^2 < 0.1$.

We say that M is *efficiently quantumly computable* if $Q(M) \leq poly(n)$.

Computing functions: We have defined what it means for an operator to be probabilistically or quantumly efficiently computable, but we typically are interested in computing some function $f : \{0,1\}^m \to \{0,1\}^\ell$. The idea is that we say that f is efficiently computable if the corresponding operator is efficiently computable, except that we also allow to use extra memory and so to embed f in some $n = poly(m)$. We define f to be *efficiently classically computable* if there is some $n = poly(m)$ such that the operator M_g is efficiently classically computable where $g : \{0,1\}^n \to \{0,1\}^n$ is defined such that $g(x_1, \ldots, x_n) = f(x_1, \ldots, x_m)$. In the quantum case we have a slight twist since the operator M_g is not necessarily a unitary matrix.[6] Therefore we say that f is *efficiently quantumly computable* if there is $n = poly(m)$ such that the operator M_q is efficiently quantumly computable where $g : \{0,1\}^n \to \{0,1\}^n$ is defined as $g(x_1, \ldots, x_n) = x_1 \cdots x_m \| (f(x_1 \cdots x_m) 0^{n-m-\ell} \oplus x_{m+1} \cdots x_n)$.

[6] It is a good exercise to verify that for every $g : \{0,1\}^n \to \{0,1\}^n$, M_g is unitary if and only if g is a permutation.

Quantum and classical computation: The way we defined what it means for a function to be efficiently quantumly computable, it might not be clear that if $f : \{0,1\}^m \to \{0,1\}^\ell$ is a function that we can compute by a polynomial size Boolean circuit (e.g., combining polynomially many AND, OR and NOT gates) then it is also quantumly efficiently computable. The idea is that for every gate $g : \{0,1\}^2 \to \{0,1\}$ we can define an 8×8 unitary matrix M_h where $h : \{0,1\}^3 \to \{0,1\}^3$ has the form $h(a,b,c) = a, b, c \oplus g(a,b)$. So, if f has a circuit of s gates, then we can dedicate an extra bit for every one of these gates and then run the corresponding s unitary operations one by one, at the end of which we will get an operator that computes the mapping

$x_1, \ldots, x_{m+\ell+s} = x_1 \cdots x_m \| x_{m+1} \cdots x_{m+s} \oplus f(x_1, \ldots, x_m) \| g(x_1 \ldots x_m)$ where the the $\ell + i^{th}$ bit of $g(x_1, \ldots, x_n)$ is the result of applying the i^{th} gate in the calculation of $f(x_1, \ldots, x_m)$. So this is "almost" what we wanted except that we have this "extra junk" that we need to get rid of. The idea is that we now simply run the same computation again which will basically we mean we XOR another copy of $g(x_1, \ldots, x_m)$ to the last s bits, but since $g(x) \oplus g(x) = 0^s$ we get that we compute the map $x \mapsto x_1 \cdots x_m \| (f(x_1, \ldots, x_m) 0^s \oplus x_{m+1} \cdots x_{m+\ell+s})$ as desired.

The "obviously exponential" fallacy: A priori it might seem "obvious" that quantum computing is exponentially powerful, since to compute a quantum computation on n bits we need to maintain the 2^n dimensional state vector and apply $2^n \times 2^n$ matrices to it. Indeed popular descriptions of quantum computing (too) often say something along the lines that the difference between quantum and classical computer is that a classic bit can either be zero or one while a qubit can be in both states at once, and so in many qubits a quantum computer can perform exponentially many computations at once. Depending on how you interpret this, this description is either false or would apply equally well to *probabilistic computation*. However, for probabilistic computation it is a not too hard exercise to show that if $f : \{0,1\}^m \to \{0,1\}^n$ is an efficiently computable function then it has a polynomial size circuit of AND, OR and NOT gates.[7] Moreover, this "obvious" approach for simulating a quantum computation will take not just exponential time but *exponential space* as well, while it is not hard to show that using a simple recursive formula one can calculate the final quantum state using *polynomial space* (in physics parlance this is known as "Feynman path integrals"). So, the exponentially long vector description by itself does not imply that quantum computers are exponentially powerful. Indeed, we cannot *prove* that they are (since in particular we can't prove that *every* polynomial space calculation can be done in polynomial time, in complexity parlance we don't know how to rule out that $P = PSPACE$), but we do have some problems (integer factoring most prominently) for which they do provide exponential speedup over the currently best *known* classical (deterministic or probabilistic) algorithms.

[7] It is a good exercise to show that if M is a probabilistic process with $R(M) \leq T$ then there exists a probabilistic circuit of size, say, $100Tn^2$ that approximately computes M in the sense that for every input x, $\sum_{y \in \{0,1\}^n} |\Pr[C(x) = y] - M_{x,y}| < 1/3$.

18.3.1 Physically realizing quantum computation

To realize quantum computation one needs to create a system with n independent binary states (i.e., "qubits"), and be able to manipulate small subsets of two or three of these qubits to change their state. While by the way we defined operations above it might seem that one needs to be able to perform arbitrary unitary operations on these two or three qubits, it turns out that there several choices for *universal sets* - a small constant number of gates that generate all others. The

biggest challenge is how to keep the system from being measured and *collapsing* to a single classical combination of states. This is sometimes known as the *coherence time* of the system. The threshold theorem says that there is some absolute constant level of errors τ so that if errors are created at every gate at rate smaller than τ then we can recover from those and perform arbitrary long computations. (Of course there are different ways to model the errors and so there are actually several threshold *theorems* corresponding to various noise models).

There have been several proposals to build quantum computers:

- Superconducting quantum computers use super-conducting electric circuits to do quantum computation. These are currently the devices with largest number of fully controllable qubits.

- At Harvard, Lukin's group is using cold atoms to implement quantum computers.

- Trapped ion quantum computers Use the states of an ion to simulate a qubit. People have made some recent advances on these computers too. For example, an ion-trap computer was used to implement Shor's algorithm to factor 15. (It turns out that $15 = 3 \times 5$:))

- Topological quantum computers use a different technology, which is more stable by design but arguably harder to manipulate to create quantum computers.

These approaches are not mutually exclusive and it could be that ultimately quantum computers are built by combining all of them together. At the moment, we have devices with about 100 qubits, and about 1% error per gate. Such restricted machines are sometimes called "Noisy Intermediate-Scale Quantum Computers" or "NISQ". See this article by John Preskil for some of the progress and applications of such more restricted devices. If the number of qubits is increased and the error is decreased by one or two orders of magnitude, we could start seeing more applications.

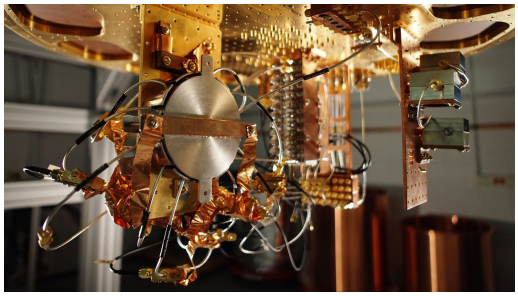

Figure 18.3: Superconducting quantum computer prototype at Google. Image credit: Google / MIT Technology Review.

18.3.2 Bra-ket notation

Quantum computing is very confusing and counterintuitive for many reasons. But there is also a "cultural" reason why people sometimes find quantum arguments hard to follow. Quantum folks follow their own special notation for vectors. Many non quantum people find it ugly and confusing, while quantum folks secretly wish they people used it all the time, not just for non-quantum linear algebra, but also for restaurant bills and elemntary school math classes.

The notation is actually not so confusing. If $x \in \{0,1\}^n$ then $|x\rangle$ denotes the x^{th} standard basis vector in 2^n dimension. That is $|x\rangle$ 2^n-dimensional column vector that has 1 in the x^{th} coordinate and zero everywhere else. So, we can describe the column vector that has α_x in the x^{th} entry as $\sum_{x\in\{0,1\}^n} \alpha_x |x\rangle$. One more piece of notation that is useful is that if $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$ then we identify $|x\rangle|y\rangle$ with $|xy\rangle$ (that is, the 2^{n+m} dimensional vector that has 1 in the coordinate corresponding to the concatenation of x and y, and zero everywhere else). This is more or less all you need to know about this notation to follow this lecture.[8]

A quantum gate is an operation on at most three bits, and so it can be completely specified by what it does to the 8 vectors $|000\rangle, \ldots, |111\rangle$. Quantum states are always unit vectors and so we sometimes omit the normalization for convenience; for example we will identify the state $|0\rangle + |1\rangle$ with its normalized version $\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$.

[8] If you are curious, there is an analog notation for *row* vectors as $\langle x|$. Generally if u is a vector then $|u\rangle$ would be its form as a column vector and $\langle u|$ would be its form as a row product. Hence since $u^\top v = \langle u, v\rangle$ the inner product of u and b can be thought of as $\langle u||v\rangle$. The *outer product* (the matrix whose i, j entry is $u_i v_j$) is denoted as $|u\rangle\langle v|$.

18.4 BELL'S INEQUALITY

There is something weird about quantum mechanics. In 1935 Einstein, Podolsky and Rosen (EPR) tried to pinpoint this issue by highlighting a previously unrealized corollary of this theory. They showed that the idea that nature does not determine the results of an experiment until it is measured results in so called "spooky action at a distance". Namely, making a measurement of one object may instantaneously effect the state (i.e., the vector of amplitudes) of another object in the other end of the universe.

Since the vector of amplitudes is just a mathematical abstraction, the EPR paper was considered to be merely a thought experiment for philosophers to be concerned about, without bearing on experiments. This changed when in 1965 John Bell showed an actual experiment to test the predictions of EPR and hence pit intuitive common sense against the quantum mechanics. Quantum mechanics won: it turns out that it *is* in fact possible to use measurements to create correlations between the states of objects far removed from one another that cannot be explained by any prior theory. Nonetheless, since the results of these experiments are so obviously wrong to anyone that has ever sat in an armchair, that there are still a number of Bell denialists arguing that this can't be true and quantum mechanics is wrong.

So, what is this Bell's Inequality? Suppose that Alice and Bob try to convince you they have telepathic ability, and they aim to prove it via the following experiment. Alice and Bob will be in separate closed rooms.[9] You will interrogate Alice and your associate will interrogate Bob. You choose a random bit $x \in \{0,1\}$ and your associate chooses a random $y \in \{0,1\}$. We let a be Alice's response and b be Bob's

[9] If you are extremely paranoid about Alice and Bob communicating with one another, you can coordinate with your assistant to perform the experiment exactly at the same time, and make sure that the rooms are sufficiently far apart (e.g., are on two different continents, or maybe even one is on the moon and another is on earth) so that Alice and Bob couldn't communicate to each other in time the results of their respective coins even if they do so at the speed of light.

response. We say that Alice and Bob win this experiment if $a \oplus b = x \wedge y$. In other words, Alice and Bob need to output two bits that *disagree* if $x = y = 1$ and *agree* otherwise.[10]

Now if Alice and Bob are not telepathic, then they need to agree in advance on some strategy. It's not hard for Alice and Bob to succeed with probability $3/4$: just always output the same bit. Moreover, by doing some case analysis, we can show that no matter what strategy they use, Alice and Bob cannot succeed with higher probability than that:[11]

Theorem 18.2 — Bell's Inequality. For every two functions $f, g : \{0,1\} \to \{0,1\}$, $\Pr_{x,y \in \{0,1\}}[f(x) \oplus g(y) = x \wedge y] \leq 3/4$.

Proof. Since the probability is taken over all four choices of $x, y \in \{0,1\}$, the only way the theorem can be violated if if there exist two functions f, g that satisfy

$$f(x) \oplus g(y) = x \wedge y$$

for all the four choices of $x, y \in \{0,1\}^2$. Let's plug in all these four choices and see what we get (below we use the equalities $z \oplus 0 = z$, $z \wedge 0 = 0$ and $z \wedge 1 = z$):

$$\begin{aligned} f(0) \oplus g(0) &= 0 \quad \text{(plugging in } x = 0, y = 0) \\ f(0) \oplus g(1) &= 0 \quad \text{(plugging in } x = 0, y = 1) \\ f(1) \oplus g(0) &= 0 \quad \text{(plugging in } x = 1, y = 0) \\ f(1) \oplus g(1) &= 1 \quad \text{(plugging in } x = 1, y = 1) \end{aligned}$$

If we XOR together the first and second equalities we get $g(0) \oplus g(1) = 0$ while if we XOR together the third and fourth equalities we get $g(0) \oplus g(1) = 1$, thus obtaining a contradiction. ■

An amazing experimentally verified fact is that quantum mechanics allows for "telepathy".[12] Specifically, it has been shown that using the weirdness of quantum mechanics, there is in fact a strategy for Alice and Bob to succeed in this game with probability larger than $3/4$ (in fact, they can succeed with probability about 0.85, see Lemma 18.3).

18.5 ANALYSIS OF BELL'S INEQUALITY

Now that we have the notation in place, we can show a strategy for Alice and Bob to display "quantum telepathy" in Bell's Game. Recall that in the classical case, Alice and Bob can succeed in the "Bell Game" with probability at most $3/4 = 0.75$. We now show that quantum mechanics allows them to succeed with probability at least 0.8.[13]

[10] This form of Bell's game was shown by Clauser, Horne, Shimony, and Holt.

[11] Theorem 18.2 below assumes that Alice and Bob use *deterministic* strategies f and g respectively. More generally, Alice and Bob could use a *randomized* strategy, or equivalently, each could choose f and g from some *distributions* $\mathcal{F}$ and $\mathcal{G}$ respectively. However the *averaging principle* (**??**) implies that if all possible deterministic strategies succeed with probability at most $3/4$, then the same is true for all randomized strategies.

[12] More accurately, one either has to give up on a "billiard ball type" theory of the universe or believe in telepathy (believe it or not, some scientists went for the latter option).

[13] The strategy we show is not the best one. Alice and Bob can in fact succeed with probability $\cos^2(\pi/8) \sim 0.854$.

Lemma 18.3 There is a 2-qubit quantum state $\psi \in \mathbb{C}^4$ so that if Alice has access to the first qubit of ψ, can manipulate and measure it and output $a \in \{0,1\}$ and Bob has access to the second qubit of ψ and can manipulate and measure it and output $b \in \{0,1\}$ then $\Pr[a \oplus b = x \wedge y] \geq 0.8$.

Proof. Alice and Bob will start by preparing a 2-qubit quantum system in the state

$$\psi = \tfrac{1}{\sqrt{2}}|00\rangle + \tfrac{1}{\sqrt{2}}|11\rangle$$

(this state is known as an EPR pair). Alice takes the first qubit of the system to her room, and Bob takes the qubit to his room. Now, when Alice receives x if $x = 0$ she does nothing and if $x = 1$ she applies the unitary map $R_{-\pi/8}$ to her qubit where $R_\theta = \begin{pmatrix} cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$ is the unitary operation corresponding to rotation in the plane with angle θ. When Bob receives y, if $y = 0$ he does nothing and if $y = 1$ he applies the unitary map $R_{\pi/8}$ to his qubit. Then each one of them measures their qubit and sends this as their response.

Recall that to win the game Bob and Alice want their outputs to be more likely to differ if $x = y = 1$ and to be more likely to agree otherwise. We will split the analysis in one case for each of the four possible values of x and y.

Case 1: $x = 0$ and $y = 0$. If $x = y = 0$ then the state does not change. * Because the state ψ is proportional to $|00\rangle + |11\rangle$, the measurements of Bob and Alice will always agree (if Alice measures 0 then the state collapses to $|00\rangle$ and so Bob measures 0 as well, and similarly for 1). Hence in the case $x = y = 1$, Alice and Bob always win.

Case 2: $x = 0$ and $y = 1$. If $x = 0$ and $y = 1$ then after Alice measures her bit, if she gets 0 then the system collapses to the state $|00\rangle$, in which case after Bob performs his rotation, his qubit is in the state $\cos(\pi/8)|0\rangle + \sin(\pi/8)|1\rangle$. Thus, when Bob measures his qubit, he will get 0 (and hence agree with Alice) with probability $\cos^2(\pi/8) \geq 0.85$. Similarly, if Alice gets 1 then the system collapses to $|11\rangle$, in which case after rotation Bob's qubit will be in the state $-\sin(\pi/8)|0\rangle + \cos(\pi/8)|1\rangle$ and so once again he will agree with Alice with probability $\cos^2(\pi/8)$.

The analysis for **Case 3**, where $x = 1$ and $y = 0$, is completely analogous to Case 2. Hence Alice and Bob will agree with probability $\cos^2(\pi/8)$ in this case as well.[14]

[14] We are using the (not too hard) observation that the result of this experiment is the same regardless of the order in which Alice and Bob apply their rotations and measurements.

Case 4: $x = 1$ and $y = 1$. For the case that $x = 1$ and $y = 1$, after both Alice and Bob perform their rotations, the state will be proportional to

$$R_{-\pi/8}|0\rangle R_{\pi/8}|0\rangle + R_{-\pi/8}|1\rangle R_{\pi/8}|1\rangle \ . \tag{18.1}$$

Intuitively, since we rotate one state by 45 degrees and the other state by -45 degrees, they will become orthogonal to each other, and the measurements will behave like independent coin tosses that agree with probability 1/2. However, for the sake of completeness, we now show the full calculation.

Opening up the coefficients and using $\cos(-x) = \cos(x)$ and $\sin(-x) = -\sin(x)$, we can see that (18.1) is proportional to

$$\begin{aligned} &\cos^2(\pi/8)|00\rangle + \cos(\pi/8)\sin(\pi/8)|01\rangle \\ -&\sin(\pi/8)\cos(\pi/8)|10\rangle + \sin^2(\pi/8)|11\rangle \\ &-\sin^2(\pi/8)|00\rangle + \sin(\pi/8)\cos(\pi/8)|01\rangle \\ -&\cos(\pi/8)\sin(\pi/8)|10\rangle + \cos^2(\pi/8)|11\rangle \ . \end{aligned}$$

Using the trigonometric identities $2\sin(\alpha)\cos(\alpha) = \sin(2\alpha)$ and $\cos^(\alpha) - \sin^2(\alpha) = \cos(2\alpha)$, we see that the probability of getting any one of $|00\rangle, |10\rangle, |01\rangle, |11\rangle$ is proportional to $\cos(\pi/4) = \sin(\pi/4) = \frac{1}{\sqrt{2}}$. Hence all four options for (a, b) are equally likely, which mean that in this case $a = b$ with probability 0.5.

Taking all the four cases together, the overall probability of winning the game is at least $\frac{1}{4} \cdot 1 + \frac{1}{2} \cdot 0.85 + \frac{1}{4} \cdot 0.5 = 0.8$.

■

R

Remark 18.4 — Quantum vs probabilistic strategies. It is instructive to understand what is it about quantum mechanics that enabled this gain in Bell's Inequality. For this, consider the following analogous probabilistic strategy for Alice and Bob. They agree that each one of them output 0 if he or she get 0 as input and outputs 1 with probability p if they get 1 as input. In this case one can see that their success probability would be $\frac{1}{4} \cdot 1 + \frac{1}{2}(1-p) + \frac{1}{4}[2p(1-p)] = 0.75 - 0.5p^2 \leq 0.75$. The quantum strategy we described above can be thought of as a variant of the probabilistic strategy for parameter p set to $\sin^2(\pi/8) = 0.15$. But in the case $x = y = 1$, instead of disagreeing only with probability $2p(1-p) = 1/4$, because we can use these negative probabilities in the quantum world and rotate the state in opposite directions, and hence the probability of disagreement ends up being $\sin^2(\pi/4) = 0.5$.

18.6 GROVER'S ALGORITHM

Shor's Algorithm, which we'll see in the next lecture, is an amazing achievement, but it only applies to very particular problems. It does not seem to be relevant to breaking AES, lattice based cryptography, or problems not related to quantum computing at all such as scheduling, constraint satisfaction, traveling salesperson etc.. etc.. Indeed, for the most general form of these search problems, classically we don't how to do anything much better than brute force search, which takes 2^n time over an n-bit domain. Lev Grover showed that quantum computers can obtain a quadratic improvement over this brute force search, solving SAT in $2^{n/2}$ time. The effect of Grover's algorithm on cryptography is fairly mild: one essentially needs to double the key lengths of symmetric primitives. But beyond cryptography, if large scale quantum computers end up being built, Grover search and its variants might end up being some of the most useful computational problems they will tackle. Grover's theorem is the following:

Theorem (Grover search , 1996): There is a quantum $O(2^{n/2}poly(n))$-time algorithm that given a $poly(n)$-sized circuit computing a function $f : \{0,1\}^n \to \{0,1\}$ outputs a string $x^* \in \{0,1\}^n$ such that $f(x^*) = 1$.

Proof sketch: The proof is not hard but we only sketch it here. The general idea can be illustrated in the case that there exists a single x^* satisfying $f(x^*) = 1$. (There is a classical reduction from the general case to this problem.) As in Simon's algorithm, we can efficiently initialize an n-qubit system to the uniform state $u = 2^{-n/2}\sum_{x\in\{0,1\}^n} |x\rangle$ which has $2^{-n/2}$ dot product with $|x^*\rangle$. Of course if we measure u, we only have probability $(2^{-n/2})^2 = 2^{-n}$ of obtaining the value x^*. Our goal would be to use $O(2^{n/2})$ calls to the oracle to transform the system to a state v with dot product at least some constant $\epsilon > 0$ with the state $|x^*\rangle$.

It is an exercise to show that using Had gets we can efficiently compute the unitary operator U such that $Uu = u$ and $Uv = -v$ for every v orthogonal to u. Also, using the circuit for f, we can efficiently compute the unitary operator U^* such that $U^*|x\rangle = |x\rangle$ for all $x \neq x^*$ and $U^*|x^*\rangle = -|x^*\rangle$. It turns out that $O(2^{n/2})$ applications of UU^* to u yield a vector v with $\Omega(1)$ inner product with $|x^*\rangle$. To see why, consider what these operators do in the two dimensional linear subspace spanned by u and $|x^*\rangle$. (Note that the initial state u is in this subspace and all our operators preserve this property.) Let $u_\perp$ be the unit vector orthogonal to u in this subspace and let $x^*_\perp$ be the unit vector orthogonal to $|x^*\rangle$ in this subspace. Restricted to this subspace, U^* is a reflection along the axis $x^*_\perp$ and U is a reflection along the axis u.

Now, let θ be the angle between u and $x_\perp^*$. These vectors are very close to each other and so θ is very small but not zero - it is equal to $\sin^{-1}(2^{-n/2})$ which is roughly $2^{-n/2}$. Now if our state v has angle $\alpha \geq 0$ with u, then as long as α is not too large (say $\alpha < \pi/8$) then this means that v has angle $u + \theta$ with $x_\perp^*$. That means that U^*v will have angle $-\alpha - \theta$ with $x_\perp^*$ or $-\alpha - 2\theta$ with u, and hence UU^*v will have angle $\alpha + 2\theta$ with u. Hence in one application from UU^* we move 2θ radians away from u, and in $O(2^{-n/2})$ steps the angle between u and our state will be at least some constant $\epsilon > 0$. Since we live in the two dimensional space spanned by u and $|x\rangle$, it would mean that the dot product of our state and $|x\rangle$ will be at least some constant as well. QED