QUANTUM COMPUTING: EFFICIENT PRIME FACTORIZATION

AARON GEELON SO

Abstract. Quantum computing has recently been the focus of much theoretical and experimental research. The reason is the belief that, compared to classical computers, quantum computers are able to solve a larger class of problems efficiently. One problem that quantum computers can solve is prime factorization, especially important since current online security is based on the conjecture that factoring large numbers is infeasible. This paper builds up to the factoring algorithm, only assuming knowledge of basic functional analysis and linear algebra.

Contents

Introduction

Near the end of the 1800s, physicists discovered phenomena unexplained by classical mechanics. This eventually led to the creation of quantum mechanics as a new paradigm to describe the physical world. Because computers are physical systems (indeed, the information that a machine manipulates is encoded in a physical entity), a new paradigm of physics brings a new way to think about computing as well.

Since computation is ultimately a physical process, it requires space, time, and energy. Given a problem, we can describe its *computational complexity* by how much resource it needs. If the resources required scales at most polynomially with respect to the input, we say that the algorithm solving it is *efficient*. Otherwise, we say that the problem is *infeasible*. For example, multiplication is efficient, but factoring is not; that is, it is exponentially harder to factor large numbers than it is to multiply them. This forms basis of RSA encryption, the method we use to securely send information online.

The interest in quantum computing stems partly from the belief that quantum algorithms can efficiently solve a larger set of problems than classical algorithms. It is easily proved that every classical algorithm has a quantum counterpart. And of course, there already exist quantum algorithms solving problems beyond the reach of current classical computers, Shor's algorithm for prime factorization being one of these.

The goal of this paper is to understand the motivation and method of Shor's algorithm. This paper will be an excursion through quantum mechanics and quantum computing. We will understand the quantum Fourier transform, which lies at the heart of the algorithm. Further, we'll cover discuss RSA encryption briefly to see where factorization comes in. Finally, we go over the algorithm itself.

Date: December 17, 2015.

1. Quantum Mechanics

In classical mechanics, every bit of information about a particle is encoded in its position and momentum; if we know what outside forces has acted on the particle, then we can determine where the particle was, where it will be, its acceleration, and so on.

We can therefore describe the state of a particle by these two quantities $(\mathbf{p}, \mathbf{q})$ representing position and momentum. We call the set of all possible states the *phase space* associated to this one-particle system.

To describe a mult-particle system, we can just look at the direct product of the phase spaces of its constituent particles. Intuitively, this means that individual particles are independent of each other. Multiple particles put together do not inherently limit the phase space.

Quantum mechanics, on the other hand, treats position and momentum as *observables*—quantities that arise out of physical measurements on the system. However, the process of measuring the system changes the system. Furthermore, multi-particle systems, or *joint systems*, are not simply a direct product of an underlying *state space*. Unlike in classical mechanics, two quantum systems can become *entangled*.

The differences between classical mechanics and quantum mechanics lead to fundamentally different ways we can think about computation using a physical machine.[1] Before we see how quantum mechanics lead to new ways of computing, first, we will build up quantum mechanics from four postulates. These postulates deal with (1) what structures the space has, (2) how measurements are performed, and (3) how multiple systems interact to form a joint system.

1.1. State space.

In classical mechanics, we have the phase space. Quantum has a state space.

Postulate 1. *The* state space *is the set S of all possible states of a physical system. It is assumed to be a subset of a complex, separable Hilbert space $\mathcal{H}$. The possible states themselves are called* state vectors *or* wavefunctions*, and these must be unit vectors.*

We denote unit vectors by $|\psi\rangle$. Remember that $|\psi\rangle$ does not represent a particle's position or momentum, but we can perform measurements $|\psi\rangle$ that give rise to quantities such as position, momentum, energy, spin, and so on.

A Hilbert space is a complete inner product space. As a bit of notation, the inner product on $\mathcal{H}$ is denoted by $\langle \phi \,|\, \psi \rangle$, where $|\phi\rangle, |\psi\rangle \in \mathcal{H}$. This is a useful notation because Riesz representation theorem then tells us that we can naturally represent the dual vectors as $\langle \phi | \in \mathcal{H}^*$, with

$$\langle \phi | \, |\psi\rangle := \langle \phi \,|\, \psi \rangle.$$

Furthermore, this gives us an easy way to construct projection operators. We can project onto the vector $|\phi\rangle$ using the operator $|\phi\rangle\langle\phi|$, where the operator is defined as

$$|\phi\rangle\langle\phi| \, |\psi\rangle := |\phi\rangle \, \langle \phi | \, \psi \rangle.$$

Here, we scale $|\phi\rangle$ by the inner product of $|\phi\rangle$ and $|\psi\rangle$. Remember that $|\phi\rangle$ is unit, so this is in fact a projection onto $|\phi\rangle$. And, if the set of vectors $|\phi_n\rangle$ are orthogonal, where $0 \leq n \leq N \leq \infty$, then we can project onto the subspace spanned by those vectors by the operator

$$\sum_{n=0}^{N} |\phi_n\rangle\langle\phi_n| \, .$$

It is also easy to prove that if the set of vectors $|\phi_n\rangle$ is an orthonormal basis, then the above operator is just the identity, which is the projection onto the whole space. Now that we have the notation, we can discuss a bit about dynamics of the space.

In classical mechanics, the state of a particle $(\mathbf{p}(t), \mathbf{q}(t))$ evolves according to Newton's laws. To describe how quantum states change $|\psi(t)\rangle$, we have Schrödinger's equation. This results in

$$|\psi(t)\rangle = U(t) \, |\psi_0\rangle \, ,$$

[1] A natural question here is *if classical mechanics does not accurately describe the physical world, why do current computers work?* That's because classical mechanics is a good approximation up until we get to very small scales, when quantum effects take over.

where $U(t)$ turns out to be a unitary operator. This makes sense because $|\psi_0\rangle$ and $|\psi(t)\rangle$ are both unit vectors. And indeed, quantum computers manipulate information through unitary operators.

Example 1.1. As an example of a quantum system, consider the one-dimensional particle in a box. Physically, this is a particle confined to a one-dimensional space of unit length, with no outside forces acting on it. The associated Hilbert space is $L^2([0,1])$, the space of complex-valued square-integrable functions over the domain $[0,1]$. The state space is the set of unit vectors in $L^2([0,1])$, which are measurable functions such that

$$\int_0^1 |f|^2 \, \mathrm{d}x = 1.$$

An example of a state vector or a wavefunction is the function $f_n : [0,1] \longrightarrow \mathbb{C}$ defined by

$$|n\rangle \equiv f_n(x) = \sqrt{2} \sin(n\pi x).$$

At this point, we should not try to ascribe a physical meaning to $f_n(x)$; it will be more productive for us to think of $|n\rangle$ as a vector in L^2 than explicitly as a function.

1.2. **Observables.** An observable of a system is a property of the system derived from a physical measurement on the system. Examples of observables are position, momentum, energy, or spin.

Take the spin of an electron, for example. Upon being measured, an electron will always either be spin up or spin down. Notice something tricky here: we *measure* spin up or spin down, but also, after measuring, the *state* of the electron is also either spin up or spin down.

What does it mean that the state of the electron is spin up/down? Did we not just refuse to give state vectors such as $|n\rangle$ a physical meaning? To illuminate this, we should think about observables in the following way.

There is a fundamental limit to our ability to discern two different state vectors. In particular, we can only distinguish orthogonal state vectors. Suppose $|0\rangle$ and $|1\rangle$ are orthogonal in a two-dimensional Hilbert space. Then, we can always tell the two vectors apart by some observable characteristic. We can call $|0\rangle$ having spin up, while $|1\rangle$ having spin down.

But what about some state vector $|\psi\rangle$ in between (more precisely, what if $|\psi\rangle$ is a linear combination of $|0\rangle$ and $|1\rangle$)? What spin does it have? We can only measure spin up or spin down; there is no spin sideways. The measurement is probabilistic. In particular, we measure spin up with probability $|\langle 0|\psi\rangle|^2$, and spin down with probability $|\langle 1|\psi\rangle|^2$.

Notice that $|0\rangle$ and $|1\rangle$ form an orthonormal basis, and since the magnitude of $|\psi\rangle$ is 1, the probabilities sum to 1 (Pythagorean's theorem). In fact, from above, we know that $|0\rangle\langle 0| + |1\rangle\langle 1|$ is the identity operator. This is consistent, implying that upon measuring the spin of $|\psi\rangle$, we always get either up or down.

But we also mentioned that the state is transformed upon measurement. If we measure spin up, then the state of the electron following the measurement is $|0\rangle$. If we think about the assumption that there is only two distinguishable spin states, we see that measurement of the system cannot preserve it. Otherwise, we could repeatedly measure it, and with arbitrary precision determine $\langle 0|\psi\rangle$ and $\langle 1|\psi\rangle$, distinguishing more states.

So, we can think of $|\psi\rangle$ as a superposition of the up and down states, and measuring the state *collapses* it into one of the 'canonical' states. This in short is Schrödinger's cat: a machine puts a cat into the superposition of its live and dead state. Until we measure the cat, it is neither dead nor alive. However, when we do look at the cat, it will necessarily be dead or alive. We can formalize this into the following postulate:

Postulate 2. *Observables are associated with Hermitian operators on $\mathcal{H}$. The eigenvalues of the operator comprise the possible values obtained from measuring the system. Let the set of eigenvectors be denoted by $|1\rangle, |2\rangle, \ldots$, and let the associated eigenvalues be $a_1, a_2, \ldots$. Let $|\psi\rangle$ denote the state of the system. The probability of measuring a_k is $|\langle k|\psi\rangle|^2$. Upon being measured, the state of the system collapses onto $|k\rangle$.*

Note a few points here. First, A is Hermitian, so its eigenvalues are real; they can correspond to physical quantities. Second, since we are working in a separable Hilbert space, there is a countable orthonormal basis (justifying our use of the naturals as indices). Third, we can expand $|\psi\rangle$ out in the eigenbasis, obtaining

$$|\psi\rangle = \sum_k c_k |k\rangle .$$

Then, the probability of measuring a_k is $|c_k|^2$. Finally, in our postulate, we made a simplying assumption that the a_j's are unique. However, this is not necessarily the case. If $a_j = a_k$, then the probability of measuring a_j is $|c_j|^2 + |c_k|^2$, and the state vector $|\psi\rangle$ is projected onto the subspace spanned by $|j\rangle$ and $|k\rangle$. In fact, if we consider the extreme case of the identity operator (all of its eigenvalues are 1), then we will measure 1 with probability 1, and the measurement projects the vector $|\psi\rangle$ back onto the whole space (i.e. nothing happens to the system, but we also gained no information about the system).

Example 1.2. Let's return to the particle-in-a-box example: we'll measure the energy of the system. Energy is associated to the Hamiltonian operator. Here, it takes the form

$$H |\psi\rangle = -\frac{\hbar^2}{2m}\frac{d^2}{dx}\psi(x).$$

where $\hbar$ and m are constants. It's easy to see that the set of vectors $|n\rangle$ for $n \in \mathbb{N}$ are all in fact eigenvectors of H, and that the associated eigenvalue to $|n\rangle$ is

$$E_n = \frac{n^2\hbar^2\pi^2}{2m}.$$

Now, let $|\psi\rangle$ be a general state of the particle in a box, described as

$$|\psi\rangle = \sum_k c_k |k\rangle .$$

Suppose we have some device that can measure the energy level of the particle. With probability $|c_n|^2$ we measure E_n.[2] Furthermore, when we measure the system, the system 'collapses' onto the subspace defined by the eigenvalue E_n. That is, after the measurement, the state of the system is $|n\rangle$, the only eigenvector with eigenvalue E_n.

The last example dealt with an infinite-dimensional Hilbert space. We should formalize finite-dimensional spaces. We already saw a finite-dimension example, with the electron spin (a.k.a. Schrödinger's cat). In fact, this is the building block of quantum computing.

Example 1.3. We call a quantum system whose associated Hilbert space is 2-dimensional a *two-level quantum system*. Recall this means that any measurement we perform on the system can take on at most two values (since we have at most two orthogonal vectors).

Label the two basis states $|0\rangle$ and $|1\rangle$. In general, the state of the electron is a linear combination

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle ,$$

where $|c_0|^2 + |c_1|^2 = 1$ to ensure $|\psi\rangle$ is unit.

For concreteness, let's assume we are dealing with electron spin. Let's try to determine the form of the observable that tells us whether the quantum system has spin up, associated to its eigenvector $|0\rangle$. This means that our observable has eigenvectors $|0\rangle$ and $|1\rangle$. Let the value 1 denote *yes*, and 0 for *no*. In a purely abstract sense, we are looking for a linear operator whose eigenvectors are $|0\rangle$ and $|1\rangle$ and their respective eigenvalues are 1 and 0.

We easily see that the projection onto $|0\rangle$ satisfies these conditions

$$A \equiv |0\rangle\langle 0| .$$

[2]We implicitly suggest that the vectors $|n\rangle$ from the particle-in-a-box form an orthonormal eigenbasis on $L^2([0,1])$. This is not strictly true, since these wavefunctions consist only of sines. However, Schrödinger's equation impose other conditions on the state space, and it turns out that these vectors span that space.

So, if we let $c_0, c_1 = 1/\sqrt{2}$, we see that the probability of measuring spin up is $|c_0|^2 = 1/2$. Thus, half of the time, we measure that the particle is in state $|0\rangle$, and other half it is in state $|1\rangle$. But suppose that we measure that the particle is in state $|0\rangle$. Then, the state of the system becomes $|0\rangle$, and any further measurements will yield $|0\rangle$ with probability 1.

Definition 1.4. In quantum computing, a two-level quantum system is called a *quantum bit* or *qubit*, analogous to the classical computing *bit*.

The bit describes a system with two states, 0 and 1. The qubit on the other hand can be in an infinite number of states, formed from the superposition of $|0\rangle$ and $|1\rangle$. If we could easily produce and access every state, then we could theoretically encode an infinite amount of information into a single qubit. However, by measuring it once, the *wavefunction collapses* onto a subspace, meaning we would need an arbitrarily large number of identical quantum systems for high precision.

Instead of trying to encode a large amount of information into a single qubit, we can also combine qubits together. A last postulate in quantum mechanics tells how quantum systems combine.

1.3. **Entanglement.** In classical mechanics, the phase space that describes a multi-particle system is the Cartesian product of the individual phase spaces; the degrees of freedom are summed. Quantum mechanics is different, where the degrees of freedom are multiplied. We will see that a consequence of this in quantum computing is the increased computing power.

Postulate 3. *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be the state space corresponding to two quantum systems. Their joint quantum system, that is, the space that describes how these two quantum systems interact, is the tensor product of the two subspaces $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$.*

We can characterize $\mathcal{H}_1 \otimes \mathcal{H}_2$ by the following proposition:

Proposition 1.5. *Let $\mathcal{H}_1$ and $\mathcal{H}_2$ be separable Hilbert spaces. Let $\{\phi_k\}$ and $\{\psi_\ell\}$ be bases on $\mathcal{H}_1$ and $\mathcal{H}_2$, respectively. Then $\{\phi_k \otimes \psi_\ell\}$ is a basis $\mathcal{H}_1 \otimes \mathcal{H}_2$.*

Proof. Clearly, the set $\{\phi_k \otimes \psi_\ell\}$ is orthonormal. We just need to show that $\phi \otimes \psi \in \mathcal{H}$ is contained in the span of $\{\phi_k \otimes \psi_\ell\}$. Since $\{\phi_k\}$ and $\{\psi_\ell\}$ are bases, we can write ϕ and ψ as linear combinations

$$\phi = \sum c_k \phi_k \qquad \psi = \sum d_\ell \psi_\ell.$$

This implies that $\sum |c_k|^2 < \infty$ and $\sum |d_\ell|^2 < \infty$. Thus, $\sum |c_k d_\ell|^2 < \infty$. Hence,

$$\mu = \sum c_k d_\ell \phi_k \otimes \psi_\ell$$

is an element of $\mathcal{H}_1 \otimes \mathcal{H}_2$, and

$$\left\| \phi \otimes \psi - \sum c_k d_\ell \phi_k \otimes \psi_\ell \right\| \longrightarrow 0.$$

This proves that $\{\phi_k \otimes \psi_\ell\}$ is a basis of $\mathcal{H}_1 \otimes \mathcal{H}_2$. ♢

We can give some intuition why joint systems are described by tensor products by example.

Example 1.6. Let's describe two particles in $\mathbb{R}^3$. The state space to describe each particle is $L^2(\mathbb{R}^3, \mathrm{d}\mu)$.[3] The state space of the joint system is $L^2(\mathbb{R}^3 \times \mathbb{R}^3, \mathrm{d}\mu_1 \otimes \mathrm{d}\mu_2)$, where $\mathrm{d}\mu_1 \otimes \mathrm{d}\mu_2$ is naturally the usual measure on the product space.[4] We claim that this space is isomorphic to $L^2(\mathbb{R}^3, \mathrm{d}\mu_1) \otimes L^2(\mathbb{R}^3, \mathrm{d}\mu_2)$. First, we find an orthonormal basis.

Let $\{\phi_k(x)\}$ and $\{\psi_\ell(y)\}$ be orthonormal bases on $L^2(\mathbb{R}^3, \mathrm{d}\mu_1)$ and $L^2(\mathbb{R}^3, \mathrm{d}\mu_2)$, respectively. By Fubini's theorem, the set $\{\phi_k(x)\psi_\ell(y)\}$ is orthonormal. To show that this set spans, suppose there is some $f(x, y) \in L^2(\mathbb{R}^3 \times \mathbb{R}^3)$ orthogonal to all elements in this set. That is,

$$\iint_{\mathbb{R}^3 \times \mathbb{R}^3} \overline{f(x,y)} \phi_k(x) \psi_\ell(y) \, \mathrm{d}\mu_1 \otimes \mathrm{d}\mu_2 = 0$$

[3] The measure μ is the usual measure in L^2; we include for clarity later.

[4] More precisely, it is the unique measure on $\mathbb{R}^3 \times \mathbb{R}^3$ such that $(\mu_1 \otimes \mu_2)(S \times T) = \mu_1(S)\mu_2(T)$, where $S, T \subset \mathbb{R}^3$.

for all k, ℓ. By Fubini's theorem, we can rewrite this as a double integral

$$\int_{\mathbb{R}^3} \left[\int_{\mathbb{R}^3} \overline{f(x,y)} \phi_k(x) \, \mathrm{d}\mu_1 \right] \psi_\ell(y) \, \mathrm{d}\mu_2 = 0.$$

As $\{\psi_\ell\}$ is a basis on $L^2(\mathbb{R}^3, \mathrm{d}\mu_2)$, this implies that $\int_{\mathbb{R}^3} \overline{f(x,y)} \phi_k(x) \, \mathrm{d}\mu_1$ is zero almost everywhere, for $y \in \mathbb{R}^3$. Restricted to the points for which the integral is zero, we have that $f(x,y) = 0$ almost everywhere for $x \in \mathbb{R}^3$. And so, $f(x,y) = 0$ almost everywhere for $(x,y) \in \mathbb{R}^3 \times \mathbb{R}^3$, proving that $\{\phi_k \psi_\ell\}$ is a basis.

Next, we can define the isomorphism. Let $U : L^2(\mathbb{R}^3) \otimes L^2(\mathbb{R}^3) \longrightarrow L^2(\mathbb{R}^3 \times \mathbb{R}^3)$ map $\phi_k \otimes \psi_\ell$ to $\phi_k \psi_\ell$; it extends uniquely to the rest of the space. In fact,

$$U(f \otimes g) = U\left(\sum c_k \phi_k \otimes \sum d_\ell \psi_\ell \right) = U\left(\sum c_k d_\ell \phi_k \otimes \psi_\ell \right) = \sum c_k d_\ell \phi_k \psi_\ell = f(x) g(y).$$

Thus, these two spaces are naturally isomorphic.

The physical consequences of the tensor product is quantum entanglement, where the state of a particle cannot be described independently of the whole system. Mathematically, we define:

Definition 1.7. Let $|\psi\rangle \in \mathcal{H}$ be a state in a joint quantum system, where $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$. We say that $|\psi\rangle$ is a *separable* state if it can be written as a tensor product of vectors in $\mathcal{H}_1$ and $\mathcal{H}_2$. That is, $|\psi\rangle$ is separable if there exist $|\alpha\rangle \in \mathcal{H}_1$ and $|\beta\rangle \in \mathcal{H}_2$ such that

$$|\psi\rangle = |\alpha\rangle \otimes |\beta\rangle .$$

Otherwise, $|\psi\rangle$ is an *entangled* state.

Example 1.8. Consider the following canonical example of entanglement. Take a two qubit system. The basis for each of the two qubits is $\{|0\rangle, |1\rangle\}$. For brevity, we'll denote the vector $|j\rangle \otimes |k\rangle$ by $|jk\rangle$. So, Propositon 1.5 tells us that a basis on the joint system is given by $|00\rangle, |01\rangle, |10\rangle, |11\rangle$.

Consider the 'Bell state' $|\psi\rangle$ defined by:

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

Suppose we want to measure the state of the first particle, so we produce a detector that tells us whether it is in state $|0\rangle$ or $|1\rangle$. Let the observable A give us the value 1 when the first particle is in $|0\rangle$ and 0 otherwise. Then, in matrix form, we can represent A by:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$

where we order the basis vectors as above. That way,the eigenvalue 1 is associated with the eigenvectors $|00\rangle$ and $|01\rangle$. These are precisely the vectors whose first particles are in the state $|0\rangle$. From the representation of $|\psi\rangle$ above, we see that we will see that the first particle is in state $|0\rangle$ with probability $1/2 = \left|1/\sqrt{2}\right|^2$. Suppose we find that the first particle is in state $|0\rangle$. Then, $|\psi\rangle$ is projected onto the subspace spanned by $|00\rangle$ and $|01\rangle$, so after this measurement, the state $|\psi\rangle$ becomes $|00\rangle$. If we then want to measure the state of the second particle, we will obtain, with probability 1, that it is in state $|0\rangle$ as well. However, before we measured the state of the first particle, we would have found that the second particle is in state $|0\rangle$ half of the time.

This example shows physical effects of entanglement: the act of measuring the first particle changes the nature of the second particle as well. This clearly differs from classical intuition, where measuring the first particle should have no effect on the second. But it is entanglement, along with superposition, that adds possibilities to quantum computing that have no analogies in classical computing.

2. Quantum Computing

Let's return to the qubit, a two-level quantum system $\mathcal{H}$, with basis $|0\rangle$ and $|1\rangle$. In practice, such a system might be realized based on nuclear spin or light polarization, for example. But for a two-level quantum system to be able to be used in a quantum computer, it must have at least the following properties: we can measure the basis states, prepare the system in a well-defined initial state, and perform any unitary operation on the system.

2.1. **Bloch sphere.**

Recall that a general quantum state can be written as:

$$|\psi\rangle = c_0 |0\rangle + c_1 |1\rangle ,$$

where $|c_0|^2 + |c_1|^2 = 1$. Furthermore, if $|\psi_2\rangle = e^{i\gamma} |\psi_1\rangle$, no measurement we can perform will differentiate the two states. So, the space of quantum states we can distinguish is a quotient space of the unit sphere in $\mathbb{C}^2$ with respect to the equivalence relation $|\psi_1\rangle \sim |\psi_2\rangle$ if the above relation holds. Another way we can visualize this space is by first writing out the general state:

$$|\psi\rangle = \cos\frac{\theta}{2} |0\rangle + e^{i\phi} \sin\frac{\theta}{2} |1\rangle ,$$

where $\theta \in [0, \pi]$ and $\phi \in [0, 2\pi]$. Notice that we can assume that the first coefficient is always nonnegative since the overall phase cannot be physically determined. The associated vector:

$$(\cos\phi \sin\theta, \sin\phi \sin\theta, \cos\theta)$$

is called the *Bloch vector*. Therefore, we can associate to every quantum state a point on the 2-sphere in $\mathbb{R}^3$. Then, $|0\rangle$ corresponds to the point $(0, 0, 1)$ and $|1\rangle$ to $(0, 0, -1)$. This is a particularly useful representation because we will soon see that unitary operations correspond to rotations of the sphere.

2.2. **Unitary operators.**

For concreteness, we will use matrix representations of operators. So, we associate to the column vectors the states:

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix} .$$

As an example, the *Hadamard* gate is the unitary operator represented by the matrix:

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

Other important operators are the *Pauli matrices*:

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \qquad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix} .$$

From the Pauli matrices, we can define a class of rotation matrices by:

$$R_{\hat{x}}(\xi) = e^{-i\xi X/2} = \begin{bmatrix} \cos\frac{\xi}{2} & -i\sin\frac{\xi}{2} \\ -i\sin\frac{\xi}{2} & \cos\frac{\xi}{2} \end{bmatrix}$$

$$R_{\hat{y}}(\xi) = e^{-i\xi Y/2} = \begin{bmatrix} \cos\frac{\xi}{2} & -\sin\frac{\xi}{2} \\ \sin\frac{\xi}{2} & \cos\frac{\xi}{2} \end{bmatrix}$$

$$R_{\hat{z}}(\xi) = e^{-i\xi Z/2} = \begin{bmatrix} e^{-i\xi/2} & 0 \\ 0 & e^{i\xi/2} \end{bmatrix}$$

We call these rotation matrices because they rotate the Bloch sphere about the x, y, or z-axes. To see this, we can first calculate the eigenvalues and eigenvectors of X, Y, and Z. For example, the eigenvalues of Z are ± 1, with eigenvectors:

$$\mathbf{v}_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \qquad \mathbf{v}_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} .$$

As above, these vectors correspond to the Bloch vectors $(0, 0, \pm 1)$. Notice that if A is an operator such that $A^2 = I$, then:

$$e^{-i\xi A} = \left[1 - \frac{1}{2!}\xi^2 + \cdots\right] I - i\left[\xi - \frac{1}{3!}\xi^3 + \cdots\right] A = \cos(\xi) I - i\sin(\xi) A,$$

where ξ is a real number. Since $X^2 = Y^2 = Z^2 = I$, we can expand $e^{i\xi A/2}$ as such. This means that vectors $\mathbf{v}_{0,1}$ are eigenvectors of $R_{\hat{z}}(\xi)$, with eigenvalue $\cos\frac{\xi}{2} \mp i\sin\frac{\xi}{2} = e^{\mp i\xi/2}$. So, for a general state vector, we get that:

$$R_{\hat{z}}(\xi)\begin{bmatrix} c_0 \\ c_1 \end{bmatrix} = e^{-i\xi/2} c_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + e^{i\xi/2} c_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix} \sim c_0 \begin{bmatrix} 1 \\ 0 \end{bmatrix} + e^{i\xi} c_1 \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where the two vectors are equivalent modulo an overall phase of $e^{-i\xi/2}$. In fact, it is easy to see that the Bloch vector transformation is:

$$(\cos\phi\sin\theta, \sin\phi\sin\theta, \cos\theta) \overset{R_z(\xi)}{\longmapsto} (\cos\mu\sin\theta, \sin\mu\sin\theta, \cos\theta),$$

where $\mu = \phi + \xi$; indeed $R_{\hat{z}}(\xi)$, a rotation by ξ about the z-axis.

Of course, we didn't need to go through all this work to find the eigenvalues and eigenvectors of $R_{\hat{z}}(\xi)$; just look at its matrix form. But, we can apply the same technique for $R_{\hat{x}}$ and $R_{\hat{y}}$. In fact, this technique allows us to look at the general rotation $R_{\hat{n}}(\xi)$, where $\hat{n} = (n_x, n_y, n_z)$ is any vector on the Bloch sphere, and:

$$R_{\hat{n}}(\xi) = \left(\cos\frac{\xi}{2} I - i\sin\frac{\xi}{2}\right)(n_x X + n_y Y + n_z Z).$$

In short, we have the following proposition:

Proposition 2.1. *Let $R_{\hat{n}}(\xi)$ as above, and $\hat{\psi}$ be the state of a qubit as represented on the Bloch sphere. Then, $R_{\hat{n}}(\xi)\hat{\psi}$ is obtained by rotating $\hat{\psi}$ about $\hat{n}$ by an angle ξ.*

Theorem 2.2. *Let U be an arbitrary unitary operation on a qubit. Then, it can be written as:*

$$U = e^{i\alpha} R_{\hat{n}}(\xi),$$

where $\alpha, \xi \in \mathbb{R}$ and $\hat{n}$ a vector on the Bloch sphere.

We won't present the proof here because it is not very enlightening, but it is based on the fact that the set $\{I, X, Y, Z\}$ forms an orthogonal basis on the space of 2×2 complex matrices. [GL, p.119] Ultimately, the unitary condition along with some algebra leads to the conclusion that U can be expanded as:

$$U = e^{i\alpha}\left[\cos\frac{\xi}{2} I - i\sin\frac{\xi}{2}(n_x X + n_y Y + n_z Z)\right] = e^{i\alpha} R_{\hat{n}}(\xi).$$

We can represent the operations using a circuit diagram. Below are the representation of certain gates:

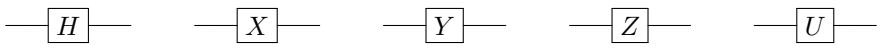

FIGURE 1. The Hadamard gate, the Pauli gates, and an arbitrary gate.

The qubit is input on the left, the gate acts on the qubit, and the output is on the right. Of course, we can string together operations. For example, if we wanted to apply the Hadamard gate then the rotation about $\hat{n}$ by ξ, we could have:

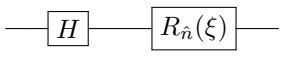

FIGURE 2. The Hadamard gate followed by a rotation.

2.3. **Control gates.** In the previous section, we looked at operations we can do to single qubits. We'll now look at controlled operations: *if A then B*. In such a system, there is a *control qubit* and a *target qubit*. The state of the control qubit determines whether a unitary operation is performed on the target qubit. For example, we'll consider the *controlled*-NOT (CNOT) gate. Here, if the control qubit is a 1, then the target qubit is flipped. To write this out in matrix form, recall the computational basis: $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, as above. Then, we have:

$$\text{CNOT} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}.$$

In a circuit diagram, we represent the control qubit by a solid dot. The NOT gate itself is an open circle. The CNOT gate is:

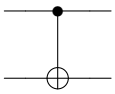

FIGURE 3. A CNOT gate. The top qubit controls the lower qubit.

If U is a unitary operator that acts on a qubit, we can turn it into a controlled-U gate, where it acts on the second qubit depending on the state of the first qubit. We represent a controlled-U gate by:

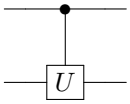

FIGURE 4. A CONTROLLED-U gate. The top qubit controls the lower qubit.

Controlled gates are important because they can generate entangled states. For example, take CNOT and the state

$$|\psi\rangle = (a|0\rangle + b|1\rangle) \otimes |0\rangle.$$

Applying CNOT to the gate produces the entangled state

$$\text{CNOT}|\psi\rangle = a|00\rangle + b|11\rangle.$$

Entangling qubits is a fundamental component of quantum computing; perhaps when it is too difficult to work with the information directly, we can couple it to other bits of information that are easier to work with. But instead of trying to convey the spirit in which entanglement is used, let's look at the quantum Fourier transform, which relies on entanglement.

2.4. **Discrete Fourier transform.** The quantum Fourier transform (QFT) is an important component to many quantum algorithms, including Shor's algorithm. In this section, we will go over the classical discrete Fourier transform (DFT), which we can think of as a unitary operator on the complex space $\mathbb{C}^n$. Because of this, we can implement the Fourier transform in a quantum computer. Those familiar with this result can move on to the next section.

The discrete Fourier transform is concerned with n-periodic functions $f : \mathbb{Z} \longrightarrow \mathbb{C}$. We can think of n-periodic functions as functions $f : [n] \longrightarrow \mathbb{C}$, where $[n] = \{1, \ldots, n\}$. In other words, $f \in \mathbb{C}^n$. Of course, $\mathbb{C}^n$ is a complete inner product space, with the usual inner product

$$\langle f, g\rangle := \sum_{k=0}^{n-1} \overline{f^k} \cdot g^k.$$

So, this space admits an orthonormal basis. This is unsurprising, since we have the standard basis on $\mathbb{C}^n$. The basis important for the DFT is the set of vectors $\{e_k\}$ defined component-wise

$$e_k^j := \frac{1}{\sqrt{n}} e^{-(2\pi i/n)jk}$$

A straightforward computation proves that this set of vectors form an orthonormal basis. The discrete Fourier transform is then just the change from the standard basis to this Fourier basis. That is, the function f is mapped to $\hat{f}$, where

$$\hat{f}(k) = \langle e_k, f\rangle = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} e^{(2\pi i/n)jk} \cdot f(j) = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} \omega_n^{jk} f(j),$$

where $\omega_n = \exp(2\pi i/n)$ is the primitive nth root of unity. A standard result from linear algebra is that such change of basis transformations are unitary. In fact, we can represent this transformation with the matrix

$$F = \frac{1}{\sqrt{n}} \begin{bmatrix} \omega_n^{0\cdot 0} & \cdots & \omega_n^{0\cdot(n-1)} \\ \vdots & \ddots & \vdots \\ \omega_n^{(n-1)\cdot 0} & \cdots & \omega_n^{n\cdot n} \end{bmatrix}$$

And, the inverse transform is given by the matrix $F^{-1} = F^*$.

2.5. **Quantum Fourier transform.** The quantum version of DFT applies precisely the same transformation onto the state vector $|\psi\rangle = \sum_{k=0}^{n-1} c_k |k\rangle$ to get

$$F|\psi\rangle = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} \omega_n^{jk} c_k |j\rangle,$$

where ω_n is the primitive nth root of unity, as above.

As an example, consider a N-qubit system; the dimension of the state space is therefore 2^N, with the basis vectors $|j\rangle$ where $j = 0, \ldots, 2^N - 1$. It will be useful to be able to represent $|j\rangle$ with its binary representation, $|j_1, \ldots, j_N\rangle$, where $j_k = 0, 1$, so we will use these two notations interchangeably.

First, for concreteness, let's calculate $F|0, \ldots, 0\rangle$. Since $|0, \ldots, 0\rangle = |0\rangle$ is the 0th basis vector, F applied to $|0\rangle$ corresponds to the 0th column of F. But $\omega_n^{k\cdot 0} = 1$, so $F|0\rangle$ is just

$$F|0, \ldots, 0\rangle = \frac{1}{\sqrt{n}} \sum_{j=0}^{n-1} |j\rangle,$$

where $n = 2^N$.

It turns out that we can represent the Fourier transform in a different way, based on the binary representation. Let's expand $F|j\rangle$ out in another way. Recall that $k = \sum_{\ell=1}^{N} k_\ell 2^{N-\ell}$. So

$$\begin{aligned} F|j\rangle &= \frac{1}{\sqrt{n}} \sum_{k=0}^{n-1} e^{2\pi i/n \cdot jk} |k\rangle \\ &= \frac{1}{2^{N/2}} \sum_{k_1=0}^{1} \cdots \sum_{k_N=0}^{1} e^{2\pi i j \cdot \left(\sum_{\ell=1}^{N} k_\ell 2^{N-\ell}/n\right)} |k_1, \ldots, k_N\rangle. \end{aligned}$$

Since $n = 2^N$, we have $2^{N-\ell}/n = 2^{-\ell}$

$$\begin{aligned} F|j\rangle = &= \frac{1}{2^{N/2}} \sum_{k_1=0}^{1} \cdots \sum_{k_N=0}^{1} \bigotimes_{\ell=1}^{N} e^{2\pi i j k_\ell 2^{-\ell}} |k_\ell\rangle \\ &= \frac{1}{2^{N/2}} \bigotimes_{l=1}^{N} \left[\sum_{k_\ell=0}^{1} e^{2\pi i j k_\ell 2^{-\ell}} |k_\ell\rangle \right] \\ &= \frac{1}{2^{N/2}} \bigotimes_{\ell=1}^{N} \left[|0\rangle + e^{2\pi i j 2^{-\ell}} |1\rangle \right] \end{aligned}$$

Finally, notice that $j2^{-\ell}$ can be written as a sum of its integral and fractional parts

$$j2^{\ell} = [j_1 \cdots j_{N-\ell}] + [0.j_{N-\ell+1} \cdots j_N]$$

Therefore, we see that

$$F|j\rangle = \frac{1}{2^{N/2}} \left(|0\rangle + e^{2\pi i [0.j_N]} |1\rangle \right) \left(|0\rangle + e^{2\pi i [0.j_{N-1} j_N]} |1\rangle \right) \cdots \left(|0\rangle + e^{2\pi i [0.j_1 \ldots j_N]} |1\rangle \right) \quad (1)$$

This way of representing the Fourier transform gives us a way to build the corresponding quantum circuit. Consider the first qubit of $F|j\rangle$

$$|j_1\rangle \mapsto \frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i [0.j_N]} |1\rangle \right).$$

Notice that this is just the Hadamard operator acting on the nth qubit, $|j_n\rangle$, since $j_n = 0$ corresponds to $e^{2\pi i [0.0]} = 1$, and $j_n = 1$ corresponds to $e^{2\pi i [0.1]} = -1$, remembering that [0.1] is binary for 1/2.

Now, look at the second qubit of $F|j\rangle$, with

$$|j_2\rangle \mapsto \frac{1}{2} \left(|0\rangle + e^{2\pi i [0.j_{N-1} j_N]} |1\rangle \right).$$

By the same analysis as before, we can create the state vector $\frac{1}{\sqrt{2}} \left(|0\rangle + e^{2\pi i [0.j_{N-1}]} |1\rangle \right)$ by applying the Hadamard operator on $|j_{N-1}\rangle$. But we are off by a phase of $e^{2\pi i/2^2}$ if $j_N = 1$. In order to obtain the correct vector, we need to apply a phase shift of $e^{2\pi i/2^2}$ only when $j_N = 1$. This is just the controlled-phase shift operator R_2 where R_k is defined as

$$R_k = \begin{bmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{bmatrix}.$$

Continuing this pattern, it's not too difficult to see that the circuit for the Fourier transform consists of Hadamard gates followed by controlled-phase shift gates, as diagrammed in Figure 5.

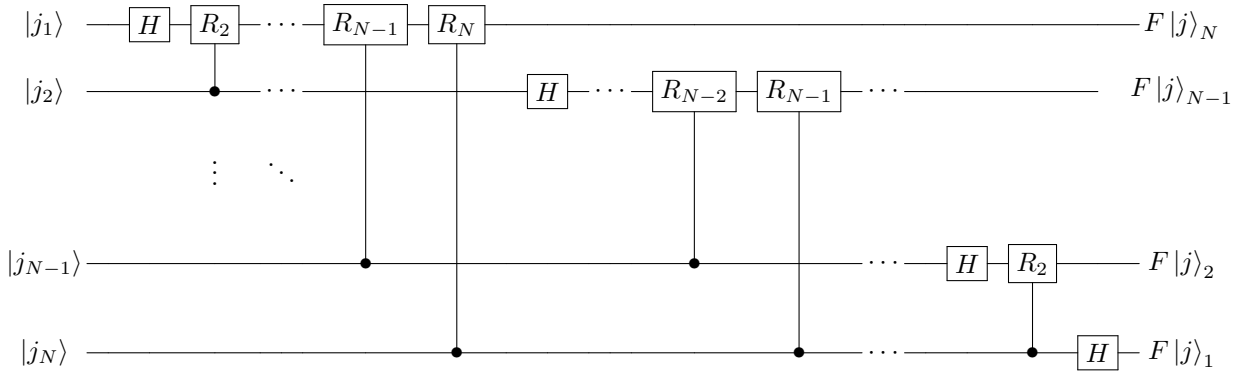

FIGURE 5. Quantum circuit for the Fourier transform.

Notice in the circuit diagram that, as shown, the output qubits are in reverse order. However, this is no problem, since the relabeling of qubits is itself a unitary operator

$$\text{SWAP} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

This circuit computes the Fourier transform of 2^n elements using $O(n^2)$ gates. The fast Fourier transform algorithms we have for classical computers would compute 2^n elements using on the order of $O(n2^n)$ gates; one uses exponentially more resource. On the other hand, quantum mechanics provide limitations of preparing the input state or measuring the output phases. Despite this, QFT still is an important component to many quantum algorithms. In particular, we will look at Shor's algorithm for prime factorization.

But before we discuss Shor's algorithm, we will go over RSA encryption in order to gain the necessary number theory and motivation to solve prime factorization efficiently.

3. RSA Encryption

RSA is a major method used to send encrypted messages between two previously noncommunicating parties, while allowing confidentiality and authentication. It is based on the asymmetry of computational power required to multiply and factor numbers. With classical computers, it is essentially impossible to factor large numbers efficiently, but easy to multiply and exponentiate.

3.1. Number theory.

Here, we cover some basic number theory required to understand RSA encryption. We begin with when division is allowed in modular arithmetic:

Lemma 3.1. *Let $a, b, c, n \in \mathbb{Z}$, $n \neq 0$, and $\gcd(a, n) = 1$. If $ab \equiv ac \pmod{n}$, then $b \equiv c \pmod{n}$.*

Proof. Since a and n are relatively prime, there exist integers x, y such that $ax + ny = 1$. Multiplying through by $(b - c)$, we get that

$$(ab - ac)x + n(b - c)y = b - c.$$

Since $(ab - ac) \equiv 0 \pmod{n}$ by assumption and $n(b - c)y \equiv 0 \pmod{n}$, this implies that $b - c$ is congruent to $0 \pmod{n}$. In other words, $b \equiv c \pmod{n}$. ♢

Theorem 3.2 (Euler's Theorem)**.** *Let $\phi(n)$ be the number of integers relatively prime to n between 1 and n. If $\gcd(a, n) = 1$, then $a^{\phi(n)} \equiv 1 \pmod{n}$.*

Proof. Let S be the set of integers between 1 and n that are relatively prime to n, so that S contains $\phi(n)$ elements. We claim that

$$\{ax \pmod{n} : x \in S\} = S.$$

In other words, the function $x \mapsto ax \pmod{n}$ is a permutation of the set S. Let $x, y \in S$ such that $ax \equiv ay \pmod{n}$. Since $\gcd(a, n) = 1$, the above lemma tells us that $x \equiv y \pmod{n}$, implying $x = y$. Second, $ax \pmod{n} \in S$ since a and x are both relatively prime to n, so their product is also relatively prime to n. This proves our claim. Then, consider the product

$$\prod_{x \in S} ax \equiv \prod_{x \in S} x \pmod{n}.$$

Since each x in S is relatively prime to n, we can divide through by x. We find that

$$\prod_{x \in S} a = a^{\phi(n)} \equiv 1 \pmod{n},$$

proving the theorem. ♢

This theorem tells us that when we work with numbers in mod n, we want to work with their exponentials in mod $\phi(n)$. That is, if $x = y + k\phi(n)$, where x, y, k are integers, and $\gcd(a, n) = 1$, then

$$a^x \pmod{n} = a^{y+k\phi(n)} \pmod{n} = a^y \cdot a^{k\phi(n)} \pmod{n} = a^y \cdot \left(a^{\phi(n)}\right)^k \pmod{n} \equiv a^y \pmod{n},$$

since $a^{\phi}(n) \equiv 1 \pmod{n}$.

3.2. **RSA algorithm.** Suppose Alice needs to send a confidential message to David, but they have not established a prior key. So, David picks two large distinct primes p, q. Let $n = pq$. Alice and David will both work in mod n. Thus, David will need to be able to work in mod $\phi(n)$ as well. Since p and q are prime, $\phi(n) = (p-1)(q-1)$. David also chooses an encryption exponent e such that

$$\gcd(e, \phi(n)) = 1.$$

Since e and $\phi(n)$ are relatively prime, David can also find a decryption exponent d such that $de \equiv 1 \pmod{\phi(n)}$. Then, he makes available to the public the pair (n, e).

Alice takes her message m (we assume that $m < n$, otherwise we could break up the message into smaller pieces), and sends back to David c, where

$$c \equiv m^e \pmod{n}.$$

David can raise c by the decryption exponent, and determine m, since

$$c^d \equiv m^{de} \pmod{n} \equiv m^{1 \pmod{\phi(n)}} \pmod{n} = m.$$

This algorithm is computationally efficient but secure because (1) it is easy to compute $m^2 \pmod{n}$, and (2) it is difficult to determine p, q from n (so one cannot easily determine d). We now discuss these points in more detail.

3.3. **Exponentiation and factoring.** Suppose that we want to compute $a^b \pmod{n}$. We consider b in binary. We then need to compute $a^{2^k} \pmod{n}$. Then, $a^b \pmod{n}$ is just the product of at most $\log_2 b$ integers less than n. Furthermore, in calculating $a^{2^k} \pmod{n}$, we never need to work with numbers greater than n^2. In short, calculating exponents can be computed quickly with limited memory.

On the other hand, we assume that it is impossible to factor n efficiently. However, notice that to decrypt the message, we only need d, the decryption exponent. We claim that finding $\phi(n)$ or finding d is equivalent to factoring n in terms of complexity.

First, suppose that we know n and $\phi(n)$. Then, we easily find p and q since $n - \phi(n) + 1 = p + q$. We claim that if we know $p+q$ and pq, then we can find p and q. Indeed, we consider the quadratic

$$(x-p)(x-q) = x^2 - (p+q)x + pq.$$

So, the quadratic equation gives us p and q from $p + q$ and pq.

Second, suppose we know d and e. We will show that if we have a universal exponent $b > 0$ such that $a^b \equiv 1 \pmod{n}$ for all a relatively prime to n, then we can probably factor n. Since $de - 1$ is a multiple of $\phi(n)$, then for any a such that $\gcd(a, n) = 1$, then

$$a^{de-1} \equiv \left(a^{\phi(n)}\right)^k \equiv 1 \pmod{n}.$$

In both cases, since it is easy to factor n after determining $\phi(n)$ or d, and because factoring is a computationally difficult problem, it is difficult to determine either $\phi(n)$ or d. Finally, let's get to Shor's algorithm.

4. Shor's Algorithm

Let $n = pq$ be a product of two primes. Let x be a nontrivial square root of 1 modulo n

$$x^2 \equiv 1 \pmod{n}, \quad x \not\equiv \pm 1 \pmod{n}.$$

These conditions tell us that $1 < x < n - 1$ and $x^2 - 1 = (x+1)(x-1) \equiv 0 \pmod{n}$. Consider the greatest common divisors

$$\gcd(x+1, n) \qquad \gcd(x-1, n).$$

At least one of these must be a nontrivial factor of n since $1 < x < n-1$. Shor's algorithm prime factorizes by finding such an x. The algorithm follows

1. Choose a random integer $a < n$. If $\gcd(a, n) \neq 1$, then we have stumbled upon a nontrivial factor of n. Otherwise,

2. Find the period r of the function $f(k) = a^k \pmod n$. In other words, we want a^r to be the identity, with $a^r \equiv 1 \pmod n$.
3. Notice that if r is even, we let $x = a^{r/2}$. And if x satisfies $x \not\equiv \pm 1 \pmod n$, then we can find nontrivial factors of n.

A few questions are immediately obvious: (i) does r exist for all a, (ii) is r always even, and (iii) when will $x \not\equiv \pm 1 \pmod n$? Once we answer these questions using a bit of group theory, we will discuss how to find the period of a function, and it is this step that will require a quantum computer.

4.1. **Groups and periods.** First, we will consider periods on $\mathbb{Z}/p\mathbb{Z}$, and the Chinese remainder theorem will help us generalize to $\mathbb{Z}/n\mathbb{Z}$. Recall that multiplicative group $(\mathbb{Z}/p\mathbb{Z})^\times$, the group of all congruence classes relatively prime to p, is a cyclic group; that is, there is some $g \in (\mathbb{Z}/p\mathbb{Z})^\times$ that generates the group. Thus, for every $a \in (\mathbb{Z}/p\mathbb{Z})^\times$, we can write

$$a \equiv g^k \pmod p .$$

Euler's theorem tells us that $a^{p-1} \equiv 1 \pmod p$. So, there exists a smallest integer r such that $a^r \equiv 1 \pmod p$. This value is also called the *order* of a.

Lemma 4.1. *Let p be an odd prime and $a \in (\mathbb{Z}/p\mathbb{Z})^\times$ be chosen uniformly at random. With probability $1/2$, the order of a is even.*

Proof. Since $a = g^k$ is chosen uniformly at random, where $1 \le k \le p-1$, half of the time k will be odd. If k is odd, notice that:

$$g^{p-1} \equiv 1 \pmod n \equiv g^{kr}.$$

This implies that $(p-1)$ divides kr. But $(p-1)$ is even while k is odd. Thus, r is even. ♢

The generalization to $(\mathbb{Z}/n\mathbb{Z})^\times$ is quick using the Chinese remainder theorem. Here, we assume that n is the power of two odd primes p and q. Let $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. We can write the orders of a as r_p and r_q for $(\mathbb{Z}/p\mathbb{Z})^\times$ and $(\mathbb{Z}/q\mathbb{Z})^\times$ respectively. Let r be the order of a in $(\mathbb{Z}/n\mathbb{Z})^\times$, which exists, once again due to Euler's theorem.

The Chinese remainder theorem tells us there is an isomorphism between the congruence classes of n and the the direct product of the congruence classes of p and q. That is, we have an isomorphism

$$[a]_n \simeq ([a]_p, [a]_q) .$$

In particular, if we were to raise a to the rth power, we get

$$([1]_p, [1]_q) \simeq [1]_n = [a]_n^r \simeq \left([a]_p^r, [a]_q^r\right) .$$

In short, we have found that raising a to the rth power yields the identity element in $(\mathbb{Z}/n\mathbb{Z})^\times$ as well as in both $(\mathbb{Z}/p\mathbb{Z})^\times$ and $(\mathbb{Z}/q\mathbb{Z})^\times$. This implies that both r_p and r_q divide r. Recapitulating, we have shown the following:

Lemma 4.2. *Let n be the product of two odd primes p and q. Let $a \in (\mathbb{Z}/n\mathbb{Z})^\times$. If r is the order of a modulo n, r_p and r_q the orders of a modulo p and q, then r_p and r_q divide r.*

Finally, we can show that the probability that the random integer a will yield nontrivial factors of n is bounded below by a positive value by the following proposition:

Proposition 4.3. *Let n, p, q be as above. Let $a \in (\mathbb{Z}/n\mathbb{Z})^\times$ be chosen uniformly at random. Let r be the order of a. The probability that r is even and that $a^{r/2} \not\equiv \pm 1 \pmod n$ is at least 1/4.*

Proof. By Lemma 4.1, the probability that r_p is even is $1/2$. Lemma 4.2 tells us that r_p divides r, so the probability that r is even is at least $1/2$. Let $x = a^{r/2} \pmod n$. The Chinese remainder theorem gives us four possibilities for x, associated to the following pairs:

$$([\pm 1]_p, [\pm 1]_q) .$$

Two of these correspond to $\pm 1 \pmod n$. Thus, the probability that $x \not\equiv \pm 1 \pmod n$ is $1/2$. The joint probability is given by their products; the probability that r is even and x is a nontrivial square root of 1 modulo n is bounded below by $1/4$. ♢

To summarize, we have shown that Shor's algorithm, with positive probability, will give us nontrivial factors of n, assuming that we can determine the order of a (or the period of the associated function f).

4.2. **Period finding.** So far, the problem of prime factorizing a number has been reduced to finding the period r of the function $f(k) = a^k \pmod n$. Here, we will present a simplified version of the algorithm; the full algorithm is similar, but requires a little bit more careful analysis. For the reader who wants to work through the full algorithm, [NC] is a great textbook to follow.

Let $Q >> n^2$ be sufficiently large. The simplifying assumption is that r divides Q. This algorithm requires two registers. The algorithm follows:

1. Set the two registers to the initial state $|0\rangle \otimes |0\rangle$.
2. Apply the Fourier transform modulo Q to the first register to get:
$$\frac{1}{\sqrt{Q}} \sum_{j=0}^{Q-1} |j\rangle \otimes |0\rangle .$$
3. Apply the function f to the second register, obtaining:
$$\frac{1}{\sqrt{Q}} \sum_{j=0}^{Q-1} |j\rangle \otimes |f(j)\rangle .$$
Note that we have now entangled the two registers; the sequence in the second register is periodic, with period r. This also means that f is one-to-one on $[0, r-1]$.
4. Measure the second register. We obtain a value $f(k)$ where k is uniformly random over $[0, r-1]$. This collapses the system to:
$$\frac{1}{\sqrt{m}} \sum_{j=0}^{m-1} |jr + k\rangle \otimes |f(k)\rangle ,$$
where $m = Q/r$. We can drop the second register now.
5. By Fourier sampling, we can convert the translation by r into a phase change:
$$\frac{1}{\sqrt{r}} \sum_{j=0}^{r-1} \omega^{jk} \left| \frac{Q}{r} j \right\rangle ,$$
where $\omega = e^{2\pi i/Q}$.
6. Measuring the first register, we get Qj/r, where j is uniformly random on $[0, r-1]$. Results from number theory on how Euler's totient function $\phi(Q)$ grows tells us that there is positive probability that $\gcd(j, Q/r) = 1$. So, by computing $\gcd(Qj/r, Q)$, we get Q/r. Thus, we can determine r.

Epilogue

We stated at the beginning that the goal is to understand the motivation and method for Shor's algorithm. As the reader might realize, this was in part an excuse to explore the many branches of mathematics that go into quantum computing.

For a mathematical treatment of quantum mechanics, I suggest [BT]. For an introduction to the C^*-algebraic formulation of quantum mechanics, [FS] is very readable.

The classic textbook for quantum computing is [NC], which is self-contained and clearly written. For a shorter and less rigorous introduction into quantum computing, I suggest [GL], which is the first of a two-volume set.

Acknowledgments. It is my pleasure to thank Peter May, who made the 2015 REU possible. I also want to thank all of the instructors and mentors teaching me math throughout the whole summer. I especially want to thank my mentor, Tori Akin; her invaluable guidance, patience, and questions have helped me produce a much more focused and concise paper.

Thank you to all my teachers, friends, and family for the continual support all these years.

References

[AD] Aerts, D., Daubechies, I. *Physical justification for using the tensor product to describe two quantum systems as one joint system.* Helvetica Physica Acta, 51, 661-675. 1978.

[BT] Ballentine, L. *Quantum Mechanics: A Modern Development.* World Scientific, New Jersey, 1998.

[FS] Strocchi, F. *An Introduction to the Mathematical Structure of Quantum Mechanics: A short course for mathematicians.* World Scientific, Singapore, 2nd edition, 2008.

[GL] Benenti, G., Casati, G., Strini, G. *Principles of Quantum Computation and Information, Volume 1: Basic Concepts.* World Scientific, New Jersey, 2004.

[NC] Nielsen, M., Chuang, I. *Quantum Computation and Quantum Information.* Cambridge University Press, New York, 2000.

[RD] Rudin, W. *Real and Complex Analysis* McGraw-Hill Mathematics Series, New York, 1987.

[TW] Trappe, W., Washington, L. *Introduction to Cryptography with Coding Theory* Pearson Prentice Hall, New Jersey, 2nd edition, 2006.