Unraveling Topological Quantum Computing

Undergraduate Thesis

Submitted in partial fulfillment for the degree of
Bachelor of Science in Physics

By

Irtaza Tanveer Ahmad

Under the supervision of

Dr. Rizwan Khalid
Dr. Adam Zaman Chaudhary

Lahore University of Management Sciences
Department of Physics

Dedicated to Nanou.

Acknowledgements

"Kelsey, in this terrifying world, all we have are the connections that we make."

I have too many people to thank for making the four years I spent at LUMS a period of my life I'm likely to never forget.

The first person I need to thank is my supervisor, Dr. Rizwan. With the unorthodox way I went about with my thesis, coupled with the fact that the subject matter is something he is not all too familiar with, he has been instrumental in keeping me sane throughout my senior year. His infinite patience for my antics, as well as his willingness to trust me with my process, is something I cannot thank him for enough.

Coming to LUMS, I initially hesitated on choosing Physics as my major, thinking I might not be cut out for it. I do not regret my decision in the slightest. The encouragement I've received from the Physics Department knows no bounds. If there is one person who embodies the supportive nature of the Department, it is Dr. Adam. From reassuring me at the start of Quantum Mechanics 2 when I felt like I flunked Quantum Mechanics 1 to helping me with my thesis as my co-supervisor, Dr. Adam has been a constant inspiration throughout my time in the Physics Department. Both he and Dr. Rizwan taught me my first physics courses when I was a freshman, and I find it poetic that they are also at the end of my undergraduate journey.

I also cannot thank my family enough. They have been my backbone for as long as I remember and I cannot imagine finding the strength to work as hard as I did at university without their support. Thank you for tolerating my procrastinating tendencies and bringing me food during the weekends when I'm speed-running through assignments. I hope that it goes without saying that what I am today is primarily because of all of you.

It would be a crime if I did not thank my seniors on account of how many times I have annoyed them with questions related to courses, my SPROJ, graduate school applications and everything in between. From wishing my batchmates and I good luck on our finals to attending our course presentations whenever such an opportunity presented itself, they made the Physics Department feel like a home away from home. I sincerely pray that all of you succeed in whatever you hope to accomplish.

Finally, I have to thank my friends, without whom I cannot imagine my life being the same. Within the Department, my batchmates have been the glue holding me together in the face of assignments and deadlines, and I am so incredibly proud of how far they've all come since first encountering them in E&M class. I will sincerely miss playing Spyfall, Slapjack and whatever other games we would end up playing on Malaik's PC with you all in the senior room. Manahil and Abeera, I cannot thank you two enough for providing me with a listening ear whenever I needed one. Hashir, Qasim and Abdullah, thank you for being some of the most inspiring people I've come across so far in my life; your dedication to LSMS, physics and mathematics is

what I hope to achieve someday. Malaik, Daniel, Rehan and Hania, thank you for teaching me that there's more to life aside from being cooped up in the senior room 24/7; you guys were the reason the senior room felt like a home to me, and I will miss all of your collective antics dearly. Hamza, you've been there since freshman year, have witnessed my toomfoolery firsthand when when we worked on LSS together (and on multiple other occasions), but still chose to be one of the best friends I've made at LUMS. Thank you, for everything. Zainab, I owe too much to you; the friendship you've given me since freshman year knows no equal. Thank you for being the equivalent of a sister to me and for seeing the best in me when I couldn't. I'll be sure to meet you when I visit Europe someday. Ibrahim, you stumbled into my life so unexpectedly but ended up being one of the best people I've come across. With you, talking about things never felt forced, and for that I thank you. I will treasure our semester-annual trips to JJ and Waabi dearly. Nyle, Ubaid and Danish, thank you for providing me with a brotherhood that transcends schools. I hope that me not meeting you guys that frequently doesn't diminish the fact that you mean so much to me, and will miss you all dearly. Nyle, only with you can I share Cowboy Bebop and Spider-Man posters, send reels about Paul Atreides or talk about Invincible, and not feel awkward. Danish, I hope you know that regardless of what the future holds in store for you, I am incredibly proud of how far you've come. Ubaid, thank you for the chai sessions at Juice Zone this semester; you are one of the most genuine souls I've met at LUMS. Abdullah, I never imagined cultivating a friendship like the one I have with you, where we can go days without talking to one another and still pick up where we left off. If anything good came out of Physics Camp, it was meeting you. Thank you for always being there to remind me that someday, it will all be worth it; that has saved me on more than one occasion.

My time at LUMS is a period of my life I will treasure because of how the Physics Department and my friends managed to make me feel safe during a part of my life that was turbulent and overcast with confusion and uncertainty about a lot of things. I am terrible at writing poems, so I dedicate the following song lyrics to you all:

Wherever you may go, a halo
And all that is lit, surrounds you
As all the lights go down
One by one, they open
Forever and ever

Abstract

"Quantum phenomena do not occur in a Hilbert space. They occur in a laboratory." This quote by Asher Peres aptly summarises our primary challenge in harnessing the full potential of quantum computers. Realistic quantum systems are unlike the particle-in-a-box problem we encounter in the classroom; they are plagued by environmental interactions that cause a loss in information, as well as errors in our own abilities to precisely manipulate such systems. It is no wonder then, that the computing revolution promised by quantum computers is more than a hop and a skip away. Luckily, there is hope. Topological quantum computing, which leverages topological aspects of quantum mechanics, presents a promising strategy to cure the illness of noise within quantum computers. Investigating the building blocks of this model of quantum computing, seeing how computations can be performed, and examining a concrete quantum system which can be used for this model are the key goals of this thesis.

Contents

Chapter 1

Noise & Errors in Quantum Computing

In this short chapter, the aim is to motivate the key issue which topological quantum computing attempts to solve: the issue of noise and errors in quantum computers.

Despite the recent boom in research as well as investments by companies such as Google and Microsoft, the road towards building actually useful quantum computers is long and filled with hurdles. Noise and errors within quantum computers arise due to the fact that most of the quantum systems that are engineered are far from the ideal particle in the box that is studied in a first course on quantum mechanics. Interactions between the qubit and the external environment are inevitable, paving the way for errors and noise to manifest and preventing quantum computers from showcasing their full potential [8]. We can also encounter errors in quantum computing due to the intricate nature of such quantum systems; even the slightest error in calibrating certain devices used in quantum computing can lead to errors in our computations. In this section, we look at a simple example of what these errors look like in a simple algorithm run on a quantum computer; we hope that it will allow the reader to see the seriousness of the problem outlined above.
The problem which we will consider is determining the parity of an integer n. This is defined to be the integer's attribute of being even or odd. Given n, finding its parity is a pretty simple problem which can be solved with a few lines of code of any programming language of your choice. The algorithm below is one recipe for using a quantum computer to solve this problem:

1. Initialise a qubit in the $|0\rangle$ state.
2. Act the Pauli $\hat{\sigma}_x$ operator on the qubit n times. Recall that $\hat{\sigma}_x |0\rangle = |1\rangle$ and $\hat{\sigma}_x |1\rangle = |0\rangle$ i.e. $\hat{\sigma}_x$ just flips the qubit.

3. Determine the expectation value of the Pauli $\hat{\sigma}_z$ operator using the final qubit we obtain. If the final qubit is $|0\rangle$ then $\langle\hat{\sigma}_z\rangle = \langle 0|\hat{\sigma}_z|0\rangle = 1$ and if the final qubit is $|1\rangle$ then $\langle\hat{\sigma}_z\rangle = \langle 1|\hat{\sigma}_z|1\rangle = -1$. This follows from the fact that $\hat{\sigma}_z|0\rangle = |0\rangle$ and $\hat{\sigma}_z|1\rangle = -|1\rangle$

Using the algorithm above, n being odd will correspond to $\langle\hat{\sigma}_z\rangle = -1$ because there will have been an odd number of flips resulting in the final qubit state being $|1\rangle$ which will be used to determine $\langle\hat{\sigma}_z\rangle$. Similarly, n being even will correspond to $\langle\hat{\sigma}_z\rangle = 1$. More concretely, with the algorithm above, $\langle\hat{\sigma}_z\rangle = (-1)^n$ i.e. the expectation value will alternate between $+1$ and -1 depending on the parity of n. However, when we try to run this program on an actual quantum computer, the result doesn't match with what we would expect as shown below:

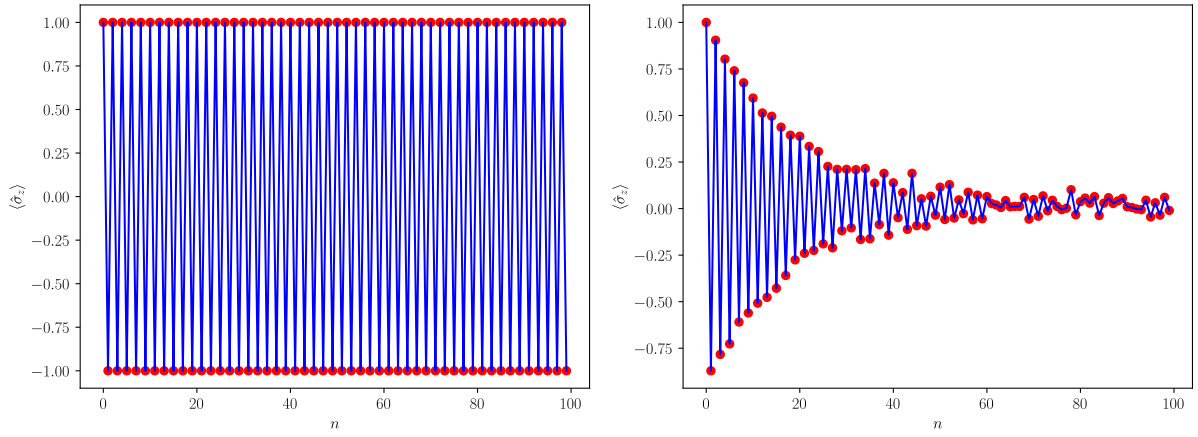


FIGURE 1.1: Expectations (left) vs Reality (right)

Hence, even simple programs such as the one described above, when run on quantum computers are not immune to the effects of noise, and much of the research within quantum computing is dedicated to reducing the effects of the different ways noise and errors manifest themselves within quantum computing architectures. Analogous to how we try to improve the performance of modern-day computers, this is tackled on two fronts. On the software front, we have attempts to develop quantum error correction techniques, where we try to encode our qubits in a way to protect them against noise [8]. On the hardware front, we try to engineer quantum systems which are essentially more robust to errors. Topological quantum computing is an attempt at the latter. The basic crux of this model is to make use of exotic particles known as *anyons* to perform computations; due to certain special properties exhibited by these particles, quantum computing using anyons ends up being significantly more robust against noise and errors as compared to more conventional quantum computing architectures [9].

Chapter 2

An Introduction to Anyons

In this chapter, the aim is to revisit the theory of identical particles in quantum mechanics and see how anyons are a theoretical possibility. Then we will review the Aharonov-Bohm Effect which can be used to construct a toy model for an anyon.

2.1 Identical Particles Revisited

In a standard undergraduate course on quantum mechanics, what is commonly taught is there are two types of particles: bosons and fermions. We can introduce this idea by considering two identical particles which we can label A and B. Let $\mathcal{H}$ denote the Hilbert space of one particle which has some orthonormal basis $\{|v_i\rangle\}$. Let the state of particle A be $|v_i\rangle$ and the state of particle B be $|v_j\rangle$ where $v_i \neq v_j$. The joint state of particles A and B can then be written as $|v_i\rangle_A \otimes |v_j\rangle_B \in \mathcal{H} \otimes \mathcal{H}$. We note that this state is essentially different from $|v_j\rangle_A \otimes |v_i\rangle_B$ which says that particle A is in the state $|v_j\rangle$ and particle B is in the state $|v_i\rangle$ i.e. A and B have been exchanged. Let us now define an operator $\hat{P}_{AB}$ in the following way

$$\hat{P}_{AB}(|v_i\rangle_A \otimes |v_j\rangle_B) = |v_j\rangle_A \otimes |v_i\rangle_B \tag{2.1}$$

As defined above, $\hat{P}_{AB}$ exchanges particles A and B and is usually called the exchange operator. Immediately one can see that $\hat{P}_{AB}^2 = \hat{\mathbb{1}}$ since two exchanges lead back to the original states of the particles. We now show that $\hat{P}_{AB}$ is a Hermitian operator. Recall that an operator $\hat{M}$ is Hermitian if $\left\langle v|\hat{M}w\middle|v|\hat{M}w\right\rangle = \left\langle v\hat{M}|w\middle|v\hat{M}|w\right\rangle \forall v, w$. We will temporarily adopt the shorthand $|v_i\rangle_A \otimes |v_j\rangle_B = v_i \otimes v_j$. We then have that

$$\langle v_k \otimes v_l|v_i \otimes v_j|v_k \otimes v_l|v_i \otimes v_j\rangle = \delta_{ik}\delta_{lj} \tag{2.2}$$

Now we compute $\left\langle v_k \otimes v_l|\hat{P}_{AB}(v_i \otimes v_j)\right|v_k \otimes v_l|\hat{P}_{AB}(v_i \otimes v_j)\Big\rangle$.

$$\left\langle v_k \otimes v_l|\hat{P}_{AB}(v_i \otimes v_j)\right|v_k \otimes v_l|\hat{P}_{AB}(v_i \otimes v_j)\Big\rangle = \langle v_k \otimes v_l|v_j \otimes v_i|v_k \otimes v_l|v_j \otimes v_i\rangle = \delta_{kj}\delta_{il} \quad (2.3)$$

Next we compute $\left\langle (v_k \otimes v_l)\hat{P}_{AB}|v_i \otimes v_j\right|(v_k \otimes v_l)\hat{P}_{AB}|v_i \otimes v_j\Big\rangle$

$$\left\langle (v_k \otimes v_l)\hat{P}_{AB}|v_i \otimes v_j\right|(v_k \otimes v_l)\hat{P}_{AB}|v_i \otimes v_j\Big\rangle = \langle v_l \otimes v_k|v_i \otimes v_j|v_l \otimes v_k|v_i \otimes v_j\rangle = \delta_{il}\delta_{kj} \quad (2.4)$$

Since both expressions of Equations 2.3 and 2.4 are equal, we conclude that $\hat{P}_{AB}$ is Hermitian. We can now consider its eigenvalues and eigenvectors.

$$\begin{aligned} &\hat{P}_{AB}\,|\psi\rangle = \lambda\,|\psi\rangle \\ &\Rightarrow \hat{P}^2_{AB}\,|\psi\rangle = \lambda^2\,|\psi\rangle \\ &\Rightarrow |\psi\rangle = \lambda^2\,|\psi\rangle \end{aligned} \quad (2.5)$$

From the above manipulation, the only allowed solutions for λ are $\lambda = \pm 1$, and each solution corresponds to some sort of exchange symmetry. We thus have that if $\hat{P}_{AB}\,|\psi\rangle = |\psi\rangle$, then $|\psi\rangle$ is a symmetric state and corresponds to bosons, whereas if $\hat{P}_{AB}\,|\psi\rangle = -\,|\psi\rangle$, then $|\psi\rangle$ is an anti-symmetric state and corresponds to fermions. The above analysis can be extended to consider N identical particles, but we will not do this here since that is not the main goal. However, what is important to realise is that the argument for the two types of exchange symmetry as described above is actually only valid when we are considering the particles to be located in three dimensional space.

In order to see how this distinction comes about, we need to physically think about the action of the exchange operator. Exchanging two particles, from a physical viewpoint, must involve moving them in some way or another. So we can think about what the configuration of the particles was before the exchange (which will be described by a certain wavefunction), realise the exchange by moving the particles along some trajectory and then looking at how the final wavefunction (i.e. after the exchange) is related to the original one.

Suppose we have two particles located at positions $\mathbf{r_1}$ and $\mathbf{r_2}$ described by some joint wavefunction $\psi(\mathbf{r_1}, \mathbf{r_2})$. Exchanging the particles by some arbitrary trajectory leads to the joint wavefunction $\psi(\mathbf{r_2}, \mathbf{r_1})$. Since the particles are identical, we can argue that the probability density function upon performing this exchange should remain unchanged i.e. $|\psi(\mathbf{r_1}, \mathbf{r_2})|^2 = |\psi(\mathbf{r_2}, \mathbf{r_1})|^2$. Thus we arrive at the following expression

$$\psi(\mathbf{r_1}, \mathbf{r_2}) = e^{i\theta}\psi(\mathbf{r_2}, \mathbf{r_1}) \quad (2.6)$$

Hence the wavefunctions differ only by some phase factor. One key physical constraint on this phase is that it must be *topological*: if one trajectory can be continuously deformed to another

without changing the initial and final positions of the particles, then both trajectories must correspond to the same phase.

Again, for simplicity, we will consider two particles. We can think of rotating the particles by π radians about a midpoint C which lies on the line joining them as shown below. A second rotation by π radians (a rotation of 2π radians in total) leads the particles back to where they originally were.

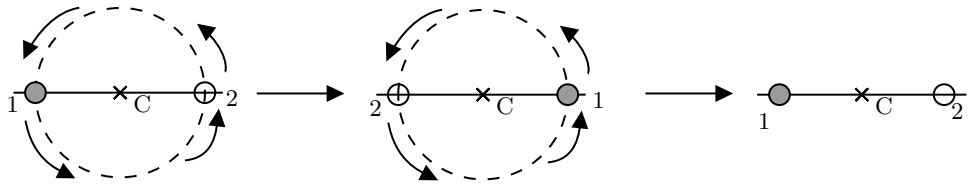


FIGURE 2.1: Identical particles 1 and 2 are exchanged by an angle π rotation about the midpoint C. A second exchange by a further angle π rotation brings them back to their original position.

Now, the idea is that the two-step exchange shown above is topologically equivalent to keeping particle 1 fixed, and moving particle 2 in a closed loop that wraps around particle 1. This can be seen by moving the point C closer and closer to particle 1. Clearly, as C approaches particle 1, particle 2 moves in a bigger circle whereas particle 1 moves in a smaller circle. When C and the position of particle 1 coincide, particle 1 does not move at all whereas particle 2 wraps around particle 1. This is shown below.

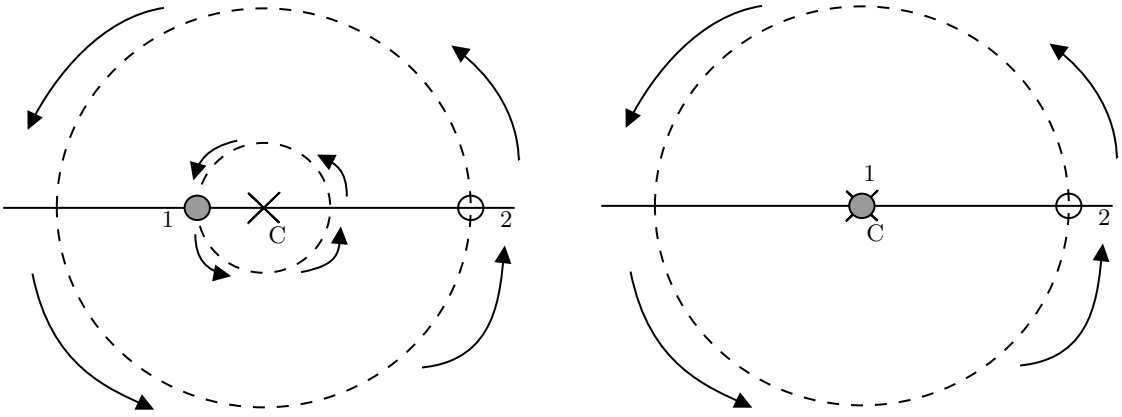


FIGURE 2.2: *Left*: Sliding the rotation point C toward particle 1, particle 1 traces a smaller circle, while particle 2 traces a larger circle. *Right*: The end result, as C reaches particle 1, is a static particle 1, with particle 2 wrapping around it.

From here, we can start to see what the difference is between 2D and 3D space. In 3D space, we can essentially deform the wraparound process even more so that even particle 2 remains fixed, and neither of the two particles undergoes any motion. This is illustrated in Figure 2.3 below.

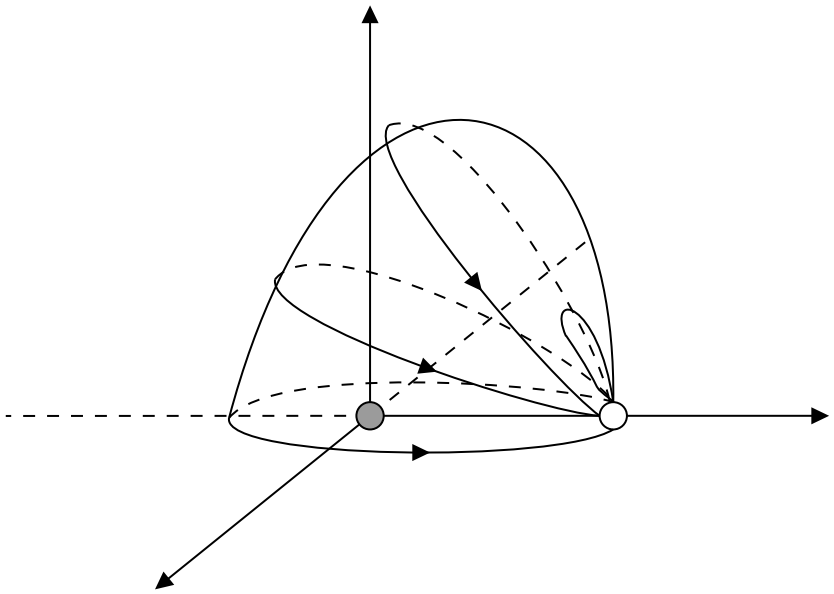


FIGURE 2.3: In three dimensions the wraparound process on the plane can be deformed continuously into a process with no motion by lifting the path into the third dimension and contracting it without hitting particle 1 at the origin.

We now ask ourselves what phase is associated with the two step exchange in 3D space, which we have argued is topologically equivalent to no motion of either particle. From the constraint that this phase has to be topological, and the fact that no phase would be associated to a process in which there is no motion, we can conclude that there is no finite phase associated with the two-step exchange process in 3D space. If that is the case, then a one-step exchange can only lead to a ± 1 factor.

In 2D space however, the wraparound process is not topologically equivalent to no motion. We do not have access to the additional dimension needed to deform the rightmost path in Figure 2.2 continuously into a path with no motion. The only alternative is to have the particles bump into each other, which results in a discontinuous deformation. Thus, in 2D, a two-step exchange allows for the phase factor in Equation 2.6 to be arbitrary. The key role played by two dimensions was realized by John Magne Leinaas and Jan Myrheim in 1977 [6]. Frank Wilczek, who went on to discover the Quantum Hall Effect called particles with such phases *anyons*, for "any" phase (and thus any statistics) allowed.

2.2 A Toy Model for an Anyon

Now that we have some idea about how, theoretically speaking, anyons can arise, it will be a good idea to try and develop a toy model for such a particle in an effort to gain an intuition of some of its properties. For this, we can look at the Aharonov-Bohm Effect in quantum mechanics.

2.2.1 A Review of the Aharonov-Bohm Effect

In classical electrodynamics, there is no disagreement on the idea that the electric and magnetic fields, $\mathbf{E}$ and $\mathbf{B}$ are the actual physical quantities that govern how a charged particle responds to electromagnetic influences. These fields can be determined by the electric potential φ and the magnetic vector potential $\mathbf{A}$ using the following equations

$$\mathbf{E} = -\nabla\varphi - \frac{\partial \mathbf{A}}{\partial t} \qquad \mathbf{B} = \nabla \times \mathbf{A} \tag{2.7}$$

However, these potentials are not unique. We can perform a gauge transformation of the potentials and still obtain the same electric and magnetic fields. The gauge transformations for both φ and $\mathbf{A}$ are

$$\tilde{\mathbf{A}} = \mathbf{A} - \nabla\chi \qquad \tilde{\varphi} = \varphi - \frac{\partial \chi}{\partial t} \tag{2.8}$$

where χ is some arbitrary function of position and time. In quantum mechanics, the potentials play a much more important role. We can see this by considering the classical Hamiltonian of a charged particle in an electromagnetic field

$$H = \frac{(\mathbf{p} - q\mathbf{A})^2}{2m} + q\varphi \tag{2.9}$$

The potentials (not the fields) are present in the Hamiltonian. We can obtain the corresponding Hamiltonian operator by substituting $\mathbf{p} \to -i\hbar\nabla$ which gives us

$$\hat{H} = \frac{(-i\hbar\nabla - q\mathbf{A})^2}{2m} + q\varphi \tag{2.10}$$

The Aharonov-Bohm Effect arises when we consider the quantum-mechanical behaviour of a charged particle in a region where the electric and magnetic fields are zero. Classically, we expect nothing to happen since the Lorentz force is trivially zero when $\mathbf{E}$ and $\mathbf{B}$ are both zero. However, in quantum mechanics, this need not be the case because we can construct non-zero potentials φ and $\mathbf{A}$ which result in $\mathbf{E} = \mathbf{B} = \mathbf{0}$. An example of such a situation is where we have a charged particle on a circular ring, through which a long solenoid passes as shown in Figure 2.4. For this configuration, the magnetic field outside the solenoid is zero but the vector potential $\mathbf{A}$ is not. The quantised energy levels E_n of the particle take the form

$$E_n = \frac{\hbar^2}{2mb^2}\left(n - \frac{q\Phi}{2\pi\hbar}\right)^2 \tag{2.11}$$

where Φ is the flux of the solenoid and b is the radius of the ring [2]. Thus, the allowed energies depend on field inside the solenoid, even though the field at the location of the particle is zero.

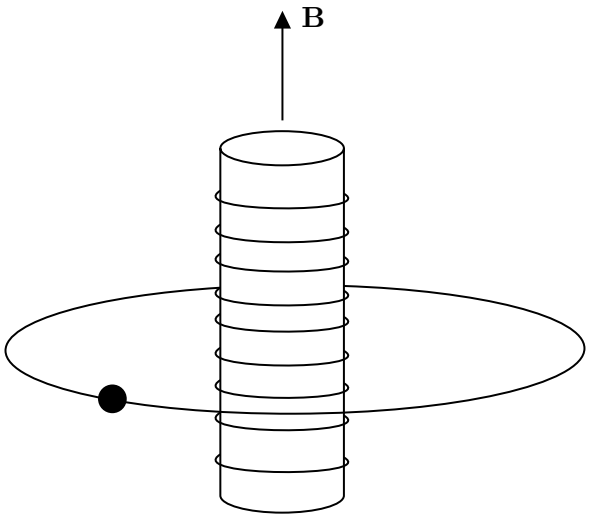


FIGURE 2.4: A charged particle on a circular ring with a solenoid passing through the center of the ring

Let us consider the more general case where a charged particle is moving in a region where $\mathbf{E} = \mathbf{B} = \mathbf{0}$, $\varphi = 0$ but $\mathbf{A} \neq \mathbf{0}$. From the Hamiltonian operator in Equation 2.10, the time-dependent Schrödinger Equation becomes

$$\frac{(-i\hbar\nabla - q\mathbf{A})^2}{2m}\psi = i\hbar\frac{\partial\psi}{\partial t} \tag{2.12}$$

The idea is to simplify the equation above by making some suitable gauge transformation. Before we do that, we need to see what happens to the wavefunction ψ upon performing a gauge transformation. The claim is that if ψ is a solution to the time-dependent Schrödinger Equation corresponding to the Hamiltonian in Equation 2.10. Then $\tilde{\psi} = e^{iq\chi/\hbar}\psi$ satisfies the time-dependent Schrödinger Equation corresponding to the Hamiltonian with the gauge transformed potentials $\tilde{\varphi}$ and $\tilde{\mathbf{A}}$ given by Equation 2.8. In order to prove this, the idea is to start with the TDSE for $\tilde{\psi}$ and arrive at the TDSE for ψ. The TDSE for $\tilde{\psi}$ is

$$\left[\frac{(-i\hbar\nabla - q\tilde{\mathbf{A}})^2}{2m} + q\tilde{\varphi}\right]\tilde{\psi} = i\hbar\frac{\partial\tilde{\psi}}{\partial t} \tag{2.13}$$

We begin by evaluating the term on the right-hand side.

$$\begin{aligned} i\hbar\frac{\partial\tilde{\psi}}{\partial t} &= i\hbar\frac{\partial}{\partial t}(e^{iq\chi/\hbar}\psi) \\ &= i\hbar\left(e^{iq\chi/\hbar}\frac{\partial\psi}{\partial t} + \frac{iq}{\hbar}\frac{\partial\chi}{\partial t}\psi\right) \\ &= i\hbar e^{iq\chi/\hbar}\frac{\partial\psi}{\partial t} - qe^{iq\chi/\hbar}\frac{\partial\chi}{\partial t}\psi \end{aligned} \tag{2.14}$$

We now evaluate the $q\tilde{\varphi}\tilde{\psi}$ term.

$$\begin{aligned} q\tilde{\varphi}\tilde{\psi} &= q\left(\varphi - \frac{\partial\chi}{\partial t}\right)(e^{iq\chi/\hbar}\psi) \\ &= q\varphi e^{iq\chi/\hbar}\psi - qe^{iq\chi/\hbar}\frac{\partial\chi}{\partial t}\psi \end{aligned} \tag{2.15}$$

All that's left to evaluate now is the $(-i\hbar\nabla - q\tilde{\mathbf{A}})^2\tilde{\psi}$ term (the $2m$ in the denominator in Equation 2.13 is a constant which can be tacked on later). To evaluate this, as a preliminary, we show that

$$(-i\hbar\nabla - q\tilde{\mathbf{A}})\tilde{\psi} = e^{iq\chi/\hbar}(-i\hbar\nabla - q\mathbf{A})\psi \tag{2.16}$$

We directly proceed with the derivation.

$$\begin{aligned} (-i\hbar\nabla - q\tilde{\mathbf{A}})\tilde{\psi} &= [-i\hbar\nabla - q(\mathbf{A} + \nabla\chi)]\,(e^{iq\chi/\hbar})\psi \\ &= -i\hbar\left[e^{iq\chi/\hbar}(\nabla\psi) + \frac{iq}{\hbar}(\nabla\chi)e^{iq\chi/\hbar}\psi\right] - q\left[\mathbf{A}e^{iq\chi/\hbar}\psi + (\nabla\chi)e^{iq\chi/\hbar}\psi\right] \\ &= e^{iq\chi/\hbar}(-i\hbar\nabla\psi) + \cancel{q(\nabla\chi)e^{iq\chi/\hbar}\psi} - e^{iq\chi/\hbar}(q\mathbf{A}\psi) - \cancel{q(\nabla\chi)e^{iq\chi/\hbar}\psi} \\ &= e^{iq\chi/\hbar}\,(-i\hbar\nabla\psi - q\mathbf{A}\psi) = e^{iq\chi/\hbar}(-i\hbar\nabla - q\mathbf{A})\psi \end{aligned} \tag{2.17}$$

With Equation 2.16 established, we can now proceed to evaluate the $(-i\hbar\nabla - q\tilde{\mathbf{A}})^2\tilde{\psi}$ term.

$$\begin{aligned} (-i\hbar\nabla - q\tilde{\mathbf{A}})^2\tilde{\psi} &= (-i\hbar\nabla - q\tilde{\mathbf{A}})(-i\hbar\nabla - q\tilde{\mathbf{A}})\tilde{\psi} \\ &= (-i\hbar\nabla - q\tilde{\mathbf{A}})e^{iq\chi/\hbar}(-i\hbar\nabla - q\mathbf{A})\psi \\ &= (-i\hbar\nabla - q\tilde{\mathbf{A}})e^{iq\chi/\hbar}\psi' \\ &= e^{iq\chi/\hbar}(-i\hbar\nabla - q\mathbf{A})\psi' = e^{iq\chi/\hbar}(-i\hbar\nabla - q\mathbf{A})^2\psi \end{aligned} \tag{2.18}$$

In going from the second line to the third line in the above manipulation, we defined $\psi' \equiv (-i\hbar\nabla - q\mathbf{A})\psi$. We now substitute all our results from Equations 2.18, 2.15 and 2.14 into Equation 2.13.

$$\begin{aligned} & e^{iq\chi/\hbar}\frac{(-i\hbar\nabla - q\mathbf{A})^2}{2m}\psi + q\varphi e^{iq\chi/\hbar}\psi - \cancel{qe^{iq\chi/\hbar}\frac{\partial\chi}{\partial t}\psi} = i\hbar e^{iq\chi/\hbar}\frac{\partial\psi}{\partial t} - \cancel{qe^{iq\chi/\hbar}\frac{\partial\chi}{\partial t}\psi} \\ \Rightarrow\ & \cancel{e^{iq\chi/\hbar}}\left[\frac{(-i\hbar\nabla - q\mathbf{A})^2}{2m} + q\varphi\right]\psi = \cancel{e^{iq\chi/\hbar}}\left[i\hbar\frac{\partial\psi}{\partial t}\right] \\ \Rightarrow\ & \left[\frac{(-i\hbar\nabla - q\mathbf{A})^2}{2m} + q\varphi\right]\psi = i\hbar\frac{\partial\psi}{\partial t} \end{aligned} \tag{2.19}$$

Thus, we've arrived at the TDSE for ψ.

Now we know that upon performing a gauge transformation, our wavefunction only picks up a phase factor. Hence, a solution to Equation 2.12 can be constructed by performing a gauge transformation to simplify the problem, solving the simplified problem and just include whatever phase factor we obtain from the gauge transformation.

The question now is what gauge transformation we should perform. Looking at Equation 2.12, it seems like a good idea to perform a transformation such that there is no vector potential term in the new TDSE i.e. $\tilde{\mathbf{A}} = \mathbf{0}$. From Equation 2.8 we then have

$$\begin{aligned} \tilde{\mathbf{A}} = \mathbf{0} &\Rightarrow \mathbf{A} + \nabla\chi = \mathbf{0} \\ &\Rightarrow \nabla\chi = -\mathbf{A} \\ &\Rightarrow \chi = -\int_{\mathcal{O}}^{\mathbf{r}} \mathbf{A} \cdot d\mathbf{r}' \end{aligned} \tag{2.20}$$

where $\mathcal{O}$ is an arbitrary reference point. Hence, with the choice of χ shown above, Equation 2.12 transforms to

$$\frac{\hbar^2}{2m}\nabla^2\tilde{\psi} = i\hbar\frac{\partial\tilde{\psi}}{\partial t} \tag{2.21}$$

where

$$\psi = \exp\left[\frac{iq}{\hbar}\int_{\mathcal{O}}^{\mathbf{r}} \mathbf{A} \cdot d\mathbf{r}'\right]\tilde{\psi} \tag{2.22}$$

Now, in the Aharonov-Bohm Effect, we essentially split an electron beam into two, and have each beam travel through the front and back of a solenoid, after which they recombine. This is shown in Figure 2.5 below. As previously pointed out, the magnetic field $\mathbf{B}$ is zero outside the solenoid, but the vector potential $\mathbf{A}$ is not. More specifically, if the radius of the solenoid is a and Φ is the magnetic flux inside the solenoid, then in cylindrical coordinates, we have

$$\mathbf{A} = \frac{\Phi}{2\pi r}\hat{\phi} \quad (r > a) \tag{2.23}$$

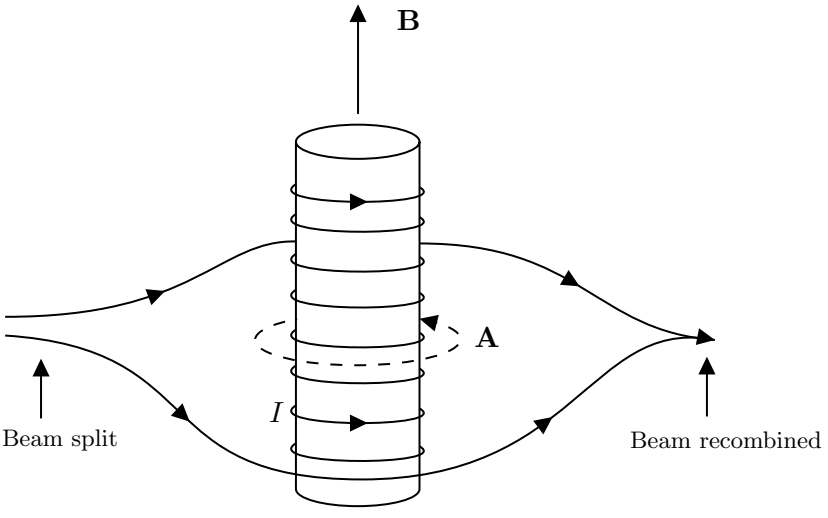


FIGURE 2.5: The Aharonov–Bohm Effect

When the two beams recombine, they are actually arriving out of phase of one another. In order to see this, we can calculate the phase of both beams using cylindrical coordinates. We place the origin on the axis of the solenoid, the reference point $\mathcal{O}$ on the incoming beam and let ϕ run

from 0 to π for one beam and from $-\pi$ to 0 for the other beam. The phase then comes out to be

$$\frac{q}{\hbar}\int\left(\frac{\Phi}{2\pi r}\hat{\phi}\right)\cdot\left(\pm r\hat{\phi}d\phi\right)=\pm\frac{q\Phi}{2\pi\hbar}\int d\phi=\pm\frac{q\Phi}{2\hbar} \tag{2.24}$$

where the plus sign corresponds to the beam travelling in the same direction as $\mathbf{A}$ and the minus sign corresponds to the beam travelling in the opposite direction. Thus, the phase difference between the two beams when they recombine is $q\Phi/\hbar$, which can be measured by means of interference experiments [2].

2.2.2 The Charge-Flux Composite Model

Now, we can think of constructing a "charge-flux composite" which comprises of a particle with some charge q bound to an infinitely thing solenoid with magnetic flux Φ [12]. We will notate such an object as a (q,Φ) particle for simplicity.

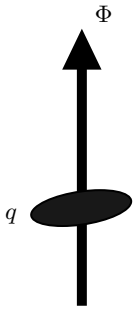


FIGURE 2.6: A charge-flux composite

Now, we can consider a (q_1,Φ_1) particle and a (q_2,Φ_2) particle, and have the latter particle encircle the former.

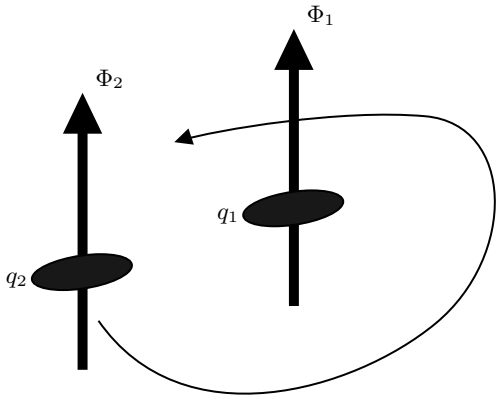


FIGURE 2.7: A (q_2,Φ_2) particle encircling a (q_1,Φ_1) particle

From our discussion of the Aharonov-Bohm Effect in the previous section, we know that this will result in a phase. There will be a phase factor of $q_2\Phi_1/\hbar$ since the charge q_2 is traveling around the flux Φ_1. Additionally, there will be a phase factor of $q_1\Phi_2/\hbar$ since the flux Φ_2 is traveling around the charge q_1. Thus the total phase accumulated is $(q_1\Phi_2 + q_2\Phi_1)/\hbar$. From our discussion about particle exchanges in Section 2.1, we can further conclude that the phase accumulated by the exchange of the two charge-flux composites is simply $(q_1\Phi_2 + q_2\Phi_1)/2\hbar$. Consequently, we have that these charge-flux composites exhibit the behaviour of anyons, in the sense that the phase accumulated by their exchange is non-trivial.

If we think of anyons as charge-flux composites, we can also develop an intuition of forming a new anyon by pushing two anyons close together. Suppose we again have a (q_1, Φ_1) anyon and a (q_2, Φ_2) anyon. If we bring them close together, the charges and fluxes should simply add up from the requirement that the total charge and flux in a region should be conserved. We can depict this diagrammatically as shown below.

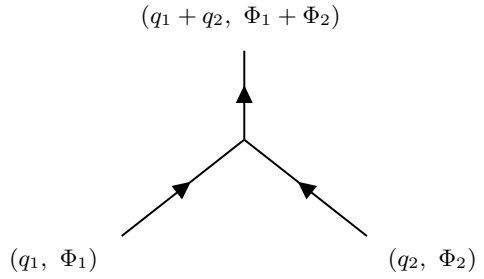


FIGURE 2.8: A (q_1, Φ_1) anyon combining with a (q_2, Φ_2) anyon to produce a $(q_1 + q_2, \Phi_1 + \Phi_2)$ anyon

Another idea which we can introduce using the charge-flux composite model is that of anti-anyons. If we have a (q, Φ) anyon, we can also think of a $(-q, -\Phi)$ anyon and upon combining the two, we obtain zero charge and flux i.e. we obtain the vacuum (which can be further thought of as a sort of identity anyon that doesn't really combine with any anyon to form something new). This is diagrammatically shown below where we've denoted the vacuum with I to emphasise its role as the identity anyon.

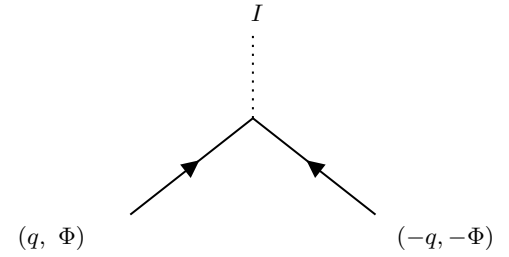


FIGURE 2.9: A (q, Φ) anyon combining with a $(-q, -\Phi)$ anyon to produce the identity anyon I i.e. the vacuum

Chapter 3

The General Theory of Anyons

The charge-flux composite model of anyons introduced in the previous chapter, being a toy model, can only take one so far in trying to understand anyons and their properties. This chapter is dedicated to discussing anyons at the most general level. This will involve providing the necessary ingredients needed to construct a consistent theory of anyons, including objects such as the F and R matrices which encode how the different anyons can combine and be exchanged with one another. Alongside this, we will focus on two concrete anyon theories: the Fibonacci and Ising Anyons.

3.1 Anyon Fusion and the Hilbert Space of Anyons

The first thing that we need to do is to specify what types of particles of present in the model of anyons we are considering. Conventionally this is done by constructing a set of particle types of the form $\mathcal{A} = \{1, a, b, c, \ldots\}$ where 1 denotes the identity and the labels a, b and c are a way for us to distinguish between the different types of anyons present in the model. When we have defined $\mathcal{A}$, the next important property we need to consider is that of *anyon fusion*. In the previous chapter, we saw an illustration of this when we discussed bringing a (q_1, Φ_1) charge-flux composite close to a (q_2, Φ_2) charge-flux composite to obtain a $(q_1 + q_2, \Phi_1 + \Phi_2)$ composite which is a completely different particle. There are two ways we can think of anyon fusion:

1. Anyon a and anyon b combine to form anyon c
2. If we have anyons a and b in some region, we can essentially "zoom out" and and think of the region as containing a single anyon c.

The first way to think about anyon fusion can be thought of as analogous to particle-antiparticle annihilation, and is the viewpoint that we implicitly adopted when building the charge-flux

composite model in the previous chapter. The second way is also not unfamiliar to many physicists. The hydrogen atom, which contains a proton and an electron (both of which are fermions), behaves like a boson. For the purposes of this chapter, it will be helpful to adopt the second way of thinking for anyon fusion.

Now, the subtlety with anyon fusion is that we need not be left with a unique type of anyon when we perform the process i.e. anyons a and b need not fuse to give anyon c. They may also fuse to give anyon d or even e. Furthermore, there can be multiple ways in which anyons a and b can combine to form anyon c [9]. Hence, we can write the fusion of two anyons a and b in the most general fashion in the following way:

$$a \times b = N_{ab}^{c} c + N_{ab}^{d} d + N_{ab}^{e} e + \ldots = \sum_{c \in \mathcal{A}} N_{ab}^{c}\, c \tag{3.1}$$

where N_{ab}^{c} is an integer known as the *fusion multiplicity* that tells us how many different ways there are to get the anyon of type c from a and b. The N_{ab}^{c} can in general be positive integers but this happens in more complicated anyon models which aren't exactly relevant for the purposes of topological quantum computing. Hence, for the purposes of this thesis, we will restrict $N_{ab}^{c} \in \{0, 1\}$ [10]. We say that a theory of anyons is *Abelian* if for each pair of anyons a and b, only one fusion outcome c is possible i.e. $N_{ab}^{c} = 1\, \forall c, a, b \in \mathcal{A}$. Otherwise, we say that the anyon theory is *non-Abelian*. For each anyon $a \in \mathcal{A}$, we also have its corresponding anti-anyon $\bar{a} \in \mathcal{A}$ such that

$$a \times \bar{a} = 1 + \sum_{c \neq 1} N_{a\bar{a}}^{c}\, c \tag{3.2}$$

which essentially tells us that when we fuse an anyon with its anti-anyon, one possible outcome is the identity but we can allow other possibilities in the most general case. Additionally, for any $a \in \mathcal{A}$, we have that

$$1 \times a = a \tag{3.3}$$

The idea of two anyons combining to give multiple possible outcomes is also an idea that physicists are not that unfamiliar with. A good analogy would be to consider the addition of angular momentum in quantum mechanics. We know that if we combine two spin-$\frac{1}{2}$ particles, we can obtain a spin singlet (a particle with a total spin of zero) or a spin triplet (a particle with a total spin of 1).

With the way we have defined the fusion of two anyons in Equation 3.1, we can also see how to deal with the fusion of more than two anyons. Suppose we first fuse a and b, and the resulting anyon is fused with c. We thus obtain the following expression

$$(a \times b) \times c = \left(\sum_{d} N_{ab}^{d}\, d \right) \times c = \sum_{d} N_{ab}^{d} (d \times c) = \sum_{d,e} N_{ab}^{c} N_{dc}^{e}\, e \tag{3.4}$$

A nice way to represent anyon fusion is by means of a *fusion diagram*. In such a diagram, we

start with the initial anyons at the top, and work our way down to the final fusion outcome. Using fusion diagrams allows us to look at fusing more than two anyons in a convenient way, as shown below.

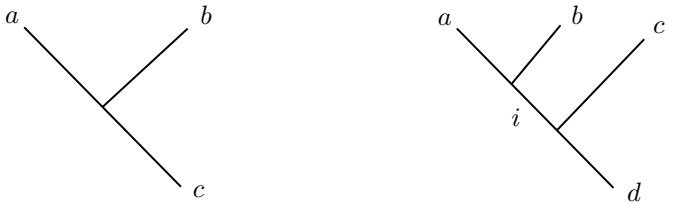


FIGURE 3.1: *Left:* A fusion diagram of anyons a and b combining to produce anyon c. *Right:* A fusion diagram of anyons a, b and c combining to produce anyon d; first a and b fuse to produce anyon i, which fuses with c to produce d.

Physically speaking, the order in which we fuse anyons together shouldn't matter, which means that anyon fusion is commutative i.e. $a \times b = b \times a$. This further means that $N_{ab}^{c} = N_{ba}^{c}$. If we have more than two anyons which are being fused, we again physically have that the order in which they are fused together shouldn't change the final outcome of the overall fusion process. Thus anyon fusion is associative i.e. $a \times (b \times c) = (a \times b) \times c$. If we consider Equation 3.3 and expand it using the definition of anyon fusion as written in Equation 3.1, we have

$$1 \times a = \sum_{c} N_{a1}^{c}\ c = a \tag{3.5}$$

which will hold only when $N_{a1}^{c} = \delta_{ac}$.

When talking about the Hilbert space of anyons, we need to be a bit careful. Although we can draw an analogy between anyon fusion and addition of angular momentum, we do not really think of anyons having any internal degrees of freedom like how a spin-$\frac{1}{2}$ particle has a 2-dimensional Hilbert space due to its spin degree of freedom. However, from Equation 3.1, we can think of a pair of anyons (which collectively carry information) as having a Hilbert space $\mathcal{H}_{ab}$ spanned by the number of possible fusion diagrams which we can construct (so each unique fusion diagram represents a quantum state), meaning that $\dim(\mathcal{H}_{ab}) = \sum_{c} N_{ab}^{c}$.

As an example, let's consider two anyons x and y which can fuse to either w or z. We thus have two fusion diagrams that can be constructed, and to each diagram we associate a ket vector. These two ket vectors are defined to be a basis for all possible states that can be constructed within the shared Hilbert space of the anyons x and y, denoted by $\mathcal{H}_{xy}$. This results in a 2-dimensional Hilbert space. For bra vectors, we essentially "flip" the fusion diagram upside down to form a splitting diagram (if a and b fuse to c, the "dual" of this would be c splitting up into a and b).

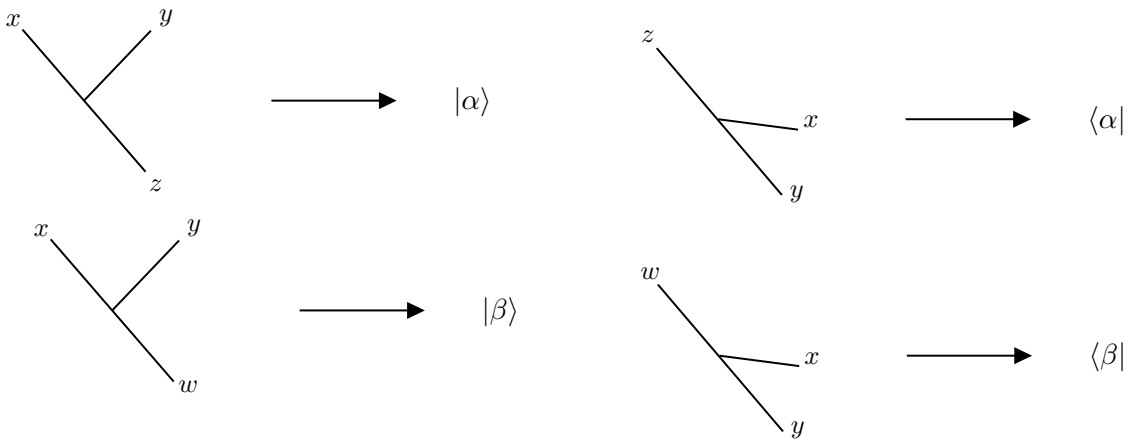


FIGURE 3.2: *Left:* The fusion diagram where anyons x and y fuse to z is denoted by the ket vector $|\alpha\rangle$, and the diagram where they fuse to w is denoted by β. $\{|\alpha\rangle, |\beta\rangle\}$ forms a basis for $\mathcal{H}_{xy}$ which is a 2-dimensional Hilbert Space. *Right:* The bra vectors $\langle\alpha|$ and $\langle\beta|$ depicted by splitting diagrams.

With anyons, we are also interested in a quantity known as the *quantum dimension* of a specific anyon. For this, we consider the Hilbert space formed when we fuse n anyons of the same type (let's say type a) together. We will denote this with $\mathcal{H}_a^{(n)}$ and wish to determine the dimension of this Hilbert space. Let a^n denote the fusion outcome from fusing n anyons of the same type a. From repeated application of Equation 3.4, we obtain the following result

$$\begin{aligned} a^n &= \sum_{b_1,b_2,\ldots b_{n-2}} N_{aa}^{b_1} N_{ab_1}^{b_2} \ldots N_{ab_{n-2}}^{b_{n-1}}\, b_{n-1} \\ \Rightarrow \dim(\mathcal{H}_a^{(n)}) &= \sum_{b_1,b_2,\ldots b_{n-2}} N_{aa}^{b_1} N_{ab_1}^{b_2} \ldots N_{ab_{n-2}}^{b_{n-1}} = \sum_b ([N_a]_{ab})^n \end{aligned} \tag{3.6}$$

where we have defined $[N_a]$ to be a matrix with components N_{ab}^c, which is being raised to the n^{th} power in the expression above. In the limit where $n \to +\infty$, this product is dominated by the largest positive eigenvalue of $[N_a]$, which is what we define to be the quantum dimension d_a of the anyon a. It essentially controls how much $\mathcal{H}_a^{(n)}$ grows as we add more anyons, and can be interpreted as the number of degrees of freedom carried by in a single anyon. However, these numbers are typically non-integer reflecting the fact that one can't really think of the information as being stored on an individual anyon [12]. One question that could be asked is whether the existence of the largest possible eigenvalue is guaranteed for the matrix $[N_a]$. The answer is yes, because of the Perron-Frobenius Theorem [3]. This theorem states that if A is a square matrix with nonnegative real entries, then A has a nonnegative real eigenvalue $\lambda_{\max}$ such that for all other eigenvalues λ of A, we have $|\lambda| \leq \lambda_{\max}$. Since N_{ab}^c are all nonnegative integers, the matrix $[N_a]$ has nonnegative real entries, and the theorem guarantees the existence of the quantum dimension d_a. One observation that we can make almost immediately is that for any arbitrary set of anyons $\mathcal{A}$, the matrix $[N_1]$ will simply be the identity matrix. This is because of

the fact that $N_{a1}^c = \delta_{ac} = N_{1a}^c$ but N_{1a}^c are just the matrix entries of $[N_1]$. This further means that $d_1 = 1$ for any $\mathcal{A}$.

3.1.1 Example 1: The Fibonacci Anyons

As an example, we can consider a class of anyons known as the Fibonacci Anyons. For this model, $\mathcal{A} = \{1, \tau\}$ and the only nontrivial fusion rule is

$$\tau \times \tau = 1 + \tau \tag{3.7}$$

because of which this anyon model is non-Abelian. We now consider the problem determining $\dim\left(\mathcal{H}_\tau^{(n)}\right)$. For $n = 2$, there are only two possible fusion diagrams which we can construt (since two τ anyons can fuse to give either 1 or τ).

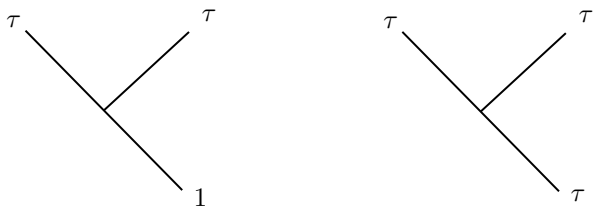


FIGURE 3.3: The possible fusion diagrams for 2 τ anyons

For $n = 3$, there are 3 fusion diagrams which we can construct.

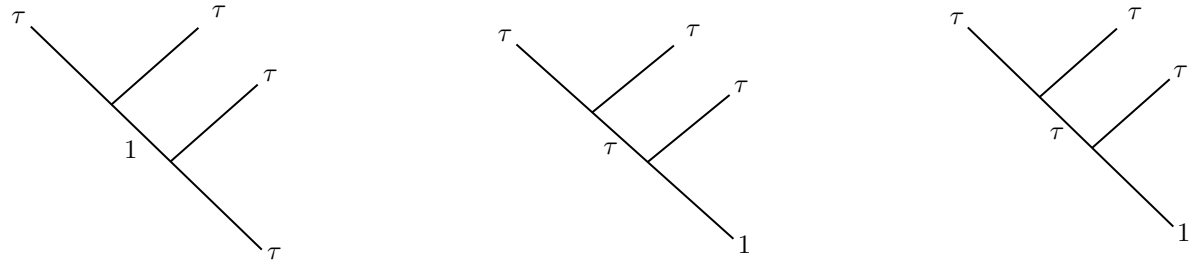


FIGURE 3.4: The possible fusion diagrams for 3 τ anyons

For $n = 4$, we have 5 possible fusion diagrams which can be constructed, as shown in Figure 3.5. More generally, one can show that for the Fibonacci anyon model the following relation holds:

$$\dim\left(\mathcal{H}_\tau^{(n+1)}\right) = \dim\left(\mathcal{H}_\tau^{(n)}\right) + \dim\left(\mathcal{H}_\tau^{(n-1)}\right) \tag{3.8}$$

which is essentially just the Fibonacci sequence (hence the name Fibonacci Anyons) [14].

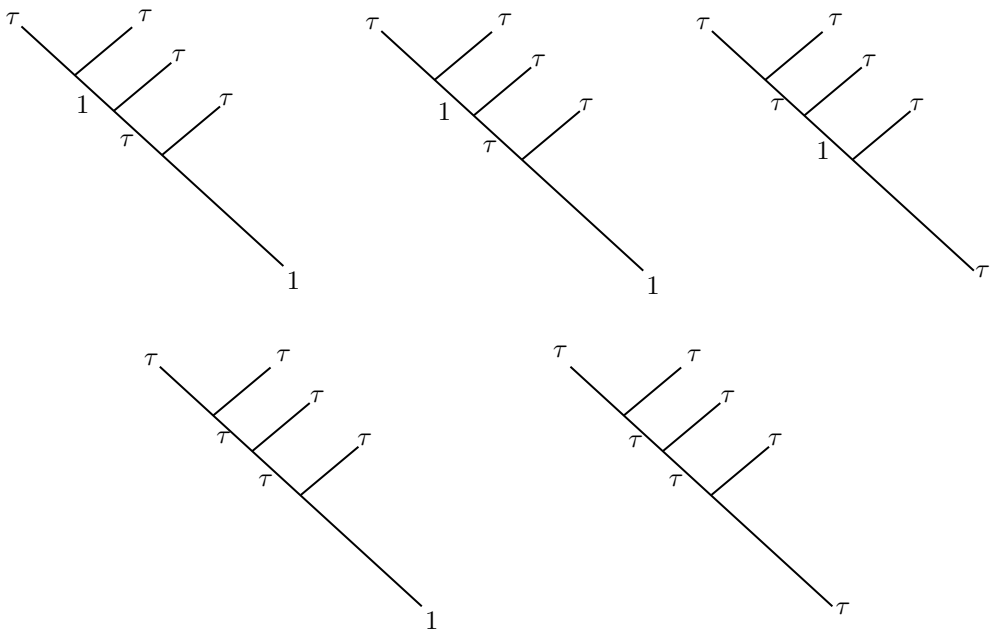


FIGURE 3.5: The possible fusion diagrams for 4 τ anyons

We can also compute the matrix $[N_\tau]$ which has the following form

$$[N_\tau] = \begin{pmatrix} N_{\tau 1}^1 & N_{\tau 1}^\tau \\ N_{\tau\tau}^1 & N_{\tau\tau}^\tau \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \tag{3.9}$$

The largest eigenvalue of the matrix above is the golden ratio φ, which means that $d_\tau = \varphi$.

3.1.2 Example 2: The Ising Anyons

As another example of a relatively simple theory of anyons, we can consider the Ising Anyons. For this model, $\mathcal{A} = \{1, \psi, \sigma\}$ and the nontrivial fusion rules are

$$\begin{aligned} \sigma \times \sigma &= 1 + \psi \\ \sigma \times \psi &= \sigma \\ \psi \times \psi &= 1 \end{aligned} \tag{3.10}$$

Let's first consider the ψ particle. We wish to determine $\dim\left(\mathcal{H}_\psi^{(n)}\right)$. From the third fusion rule, we see that it's basically its own antiparticle in some sense. For $n = 2$, there is only one fusion diagram that we can construct, shown in Figure 3.6.

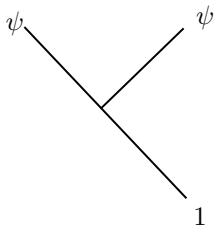


FIGURE 3.6: The only possible fusion diagram for 2 ψ anyons

For $n = 3$, we also have only one possible diagram, shown below:

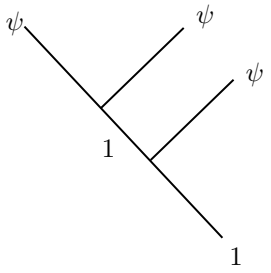


FIGURE 3.7: The only possible fusion diagram for 3 ψ anyons

For $n = 4$, we again have only one possible diagram, shown below:

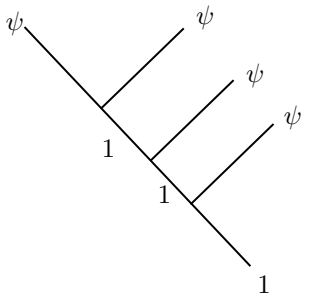


FIGURE 3.8: The only possible fusion diagram for 4 ψ anyons

More generally, if we have n ψ particles, then we will still have only one fusion diagram that can be constructed because of the last fusion rule in Equation 3.10 which tells us that fusing any two ψ particles uniquely gives us the identity. Thus we have that $\dim\left(\mathcal{H}_{\psi}^{(n)}\right) = 1\ \forall n$. Computing the matrix $[N_\psi]$, we obtain

$$[N_\psi] = \begin{pmatrix} N_{\psi 1}^{1} & N_{\psi 1}^{\sigma} & N_{\psi 1}^{\psi} \\ N_{\psi\sigma}^{1} & N_{\psi\sigma}^{\sigma} & N_{\psi\sigma}^{\psi} \\ N_{\psi\psi}^{1} & N_{\psi\psi}^{\sigma} & N_{\psi\psi}^{\psi} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \tag{3.11}$$

The largest positive eigenvalue of the matrix above is 1, which means that $d_\psi = 1$.
Now, let's consider the σ particle. For $n = 2$, we have two possible fusion diagrams which we can construct, since two σ anyons can fuse to give either 1 or ψ. This means that $\dim\left(\mathcal{H}_{\sigma}^{(2)}\right) = 2$.

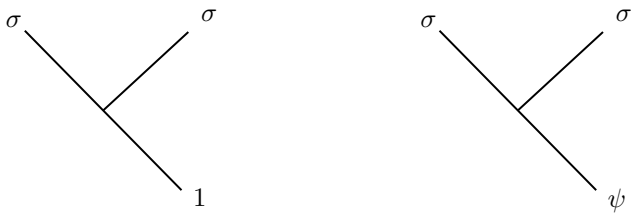


FIGURE 3.9: The possible fusion diagrams for 2 σ anyons

For $n = 3$, we find that there are also 2 possible fusion diagrams we an construct. This is because the first two σ anyons can fuse to give either 1 or ψ, but both of these fuse uniquely with σ to give σ back. This means that $\dim\left(\mathcal{H}_\sigma^{(3)}\right) = 2$.

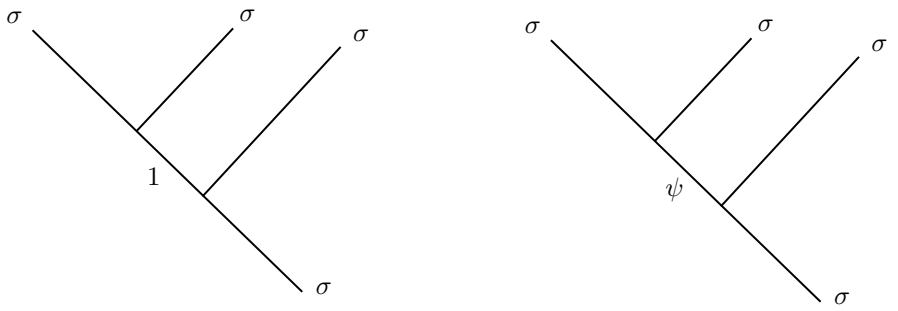


FIGURE 3.10: The possible fusion diagrams for 3 σ anyons

For $n = 4$, we find that there are 4 possible diagrams as shown in Figure 3.11. The first two σ anyons can either give 1 or ψ. Either of these would fuse with the third σ anyon to give σ, which can again fuse with the fourth σ anyon to give either 1 or ψ. This means that $\dim\left(\mathcal{H}_\sigma^{(4)}\right) = 4$. More generally, we find that $\dim\left(\mathcal{H}_\sigma^{(n)}\right) = 2^{\lfloor n/2 \rfloor}$ where $\lfloor x \rfloor$ is the floor function. We can think about this by essentially making pairs of σ anyons. If we have n σ anyons where n is even, then there are $n/2$ pairs. Each pair can fuse to either 1 or ψ, which means that each pair has 2 possibilities. This gives us $2^{n/2}$ possible fusion diagrams. If n is odd, then we can construct pairs of $(n-1)$ σ anyons. The outcome of each pair is either ψ or 1, but from the fusion rules given in Equation 3.10, both of these uniquely fuse with the remaining σ anyon to give us σ back. So there are no additional possibilities aside from the $2^{(n-1)/2}$ possibilities corresponding to the $(n-1)/2$ pairs of σ anyons. Hence $\dim\left(\mathcal{H}_\sigma^{(n)}\right) = 2^{n/2}$ if n is even and $2^{(n-1)/2}$ if n is odd; we can incorporate both cases for any integer n by introducing the floor function.
Computing the matrix $[N_\sigma]$, we obtain

$$[N_\sigma] = \begin{pmatrix} N_{\sigma 1}^{1} & N_{\sigma 1}^{\sigma} & N_{\sigma 1}^{\psi} \\ N_{\sigma\sigma}^{1} & N_{\sigma\sigma}^{\sigma} & N_{\sigma\sigma}^{\psi} \\ N_{\sigma\psi}^{1} & N_{\sigma\psi}^{\sigma} & N_{\sigma\psi}^{\psi} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \tag{3.12}$$

The largest positive eigenvalue of the matrix above is $\sqrt{2}$, which means that $d_\sigma = \sqrt{2}$.

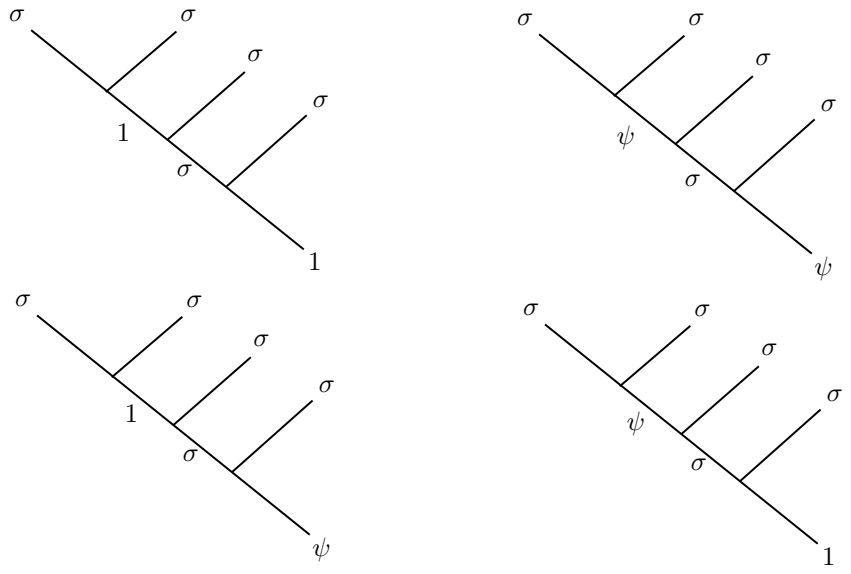


FIGURE 3.11: The possible fusion diagrams for 4 σ anyons

3.2 The F Matrix

In order to specify a particular theory of anyons, defining a set of anyons $\mathcal{A}$ alongside the fusion rules is not enough. One important ingredient that is required comes from considering the order in which multiple anyons fuse together to give the same final result. Suppose we have three anyons a, b and c which fuse amongst each other to give a final anyon d. There are two ways in which we can achieve this. One way is the fuse a and b to some anyon i, which then fuses with c to give d. The second way would be to fuse b and c together to give some anyon j, which then fuses with a to give us d. Both possibilities result in different fusion diagrams but both correspond to the same overall fusion process (a, b and c fusing to give d).

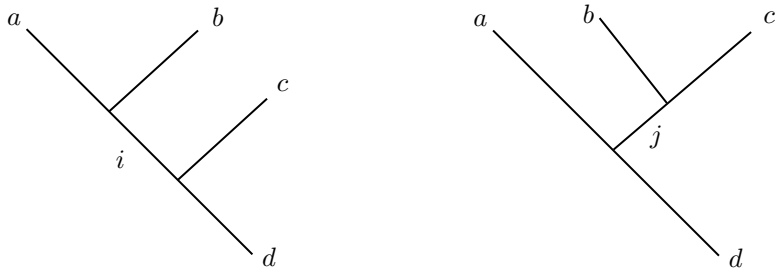


FIGURE 3.12: *Left:* Anyons a and b first fuse to i, which then fuses with c to j. *Right:* Anyons b and c fuse to j, which fuses with a to d. The fusion diagrams are different, but show the same overall fusion process of a, b and c fusing to d.

The question then is: If two different fusion diagrams indicate the same overall fusion process, how exactly will they be related? The answer turns out to be that they are related by a unitary matrix known as the *fusion (F) matrix*. One can think of two different fusion diagrams that

indicate the same overall fusion process and the F matrix which relates them in a similar way as how we think about a change of basis in linear algebra. For the fusion diagrams shown in Figure 3.12 above, the two diagrams can be related using the F matrix in the following way

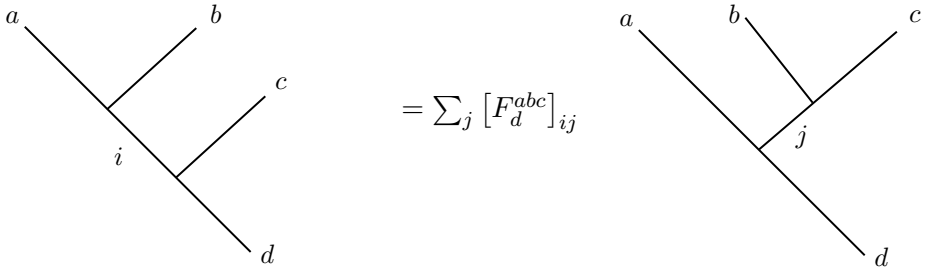
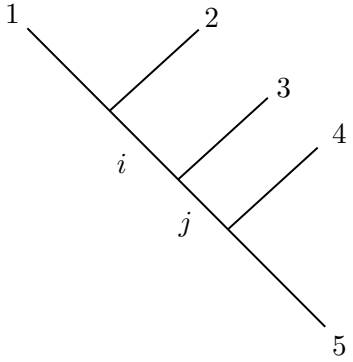


FIGURE 3.13: the F matrix for two fusion diagrams with the same overall fusion process can be thought of as a change of basis

where $[F_d^{abc}]_{ij}$ are the matrix elements of a unitary matrix F_d^{abc} which is specified by the anyons we are starting out with (a, b and c) and the final fusion result we obtain (d). One comment that should be made is that we have assumed for simplicity that the fusion multiplicities N_{ab}^c are not greater than 1. If we consider the case where $N_{ab}^c > 1$, then each vertex in the fusion diagram gets an additional index which runs from 1 up until the multiplicity of the anyon, which would result in the F matrix having additional indices [12].

3.2.1 The Pentagon Equation

The F matrices corresponding to a particular theory of anyons cannot be chosen arbitrarily. There is a consistency condition which they need to satisfy. The consistency condition comes from considering 4 anyons fusing to an end product. In order to avoid alphabetical notation for the matrix elements, we will denote the initial anyons with the numbers 1,2,3 and 4 and denote the final anyon with the number 5 (here the number 1 does not mean the identity). The starting point is to consider the following fusion diagram:



and reverse the fusion order. This can be done in two different ways, as shown in Figure 3.14 which is known as the *Pentagon Diagram*.

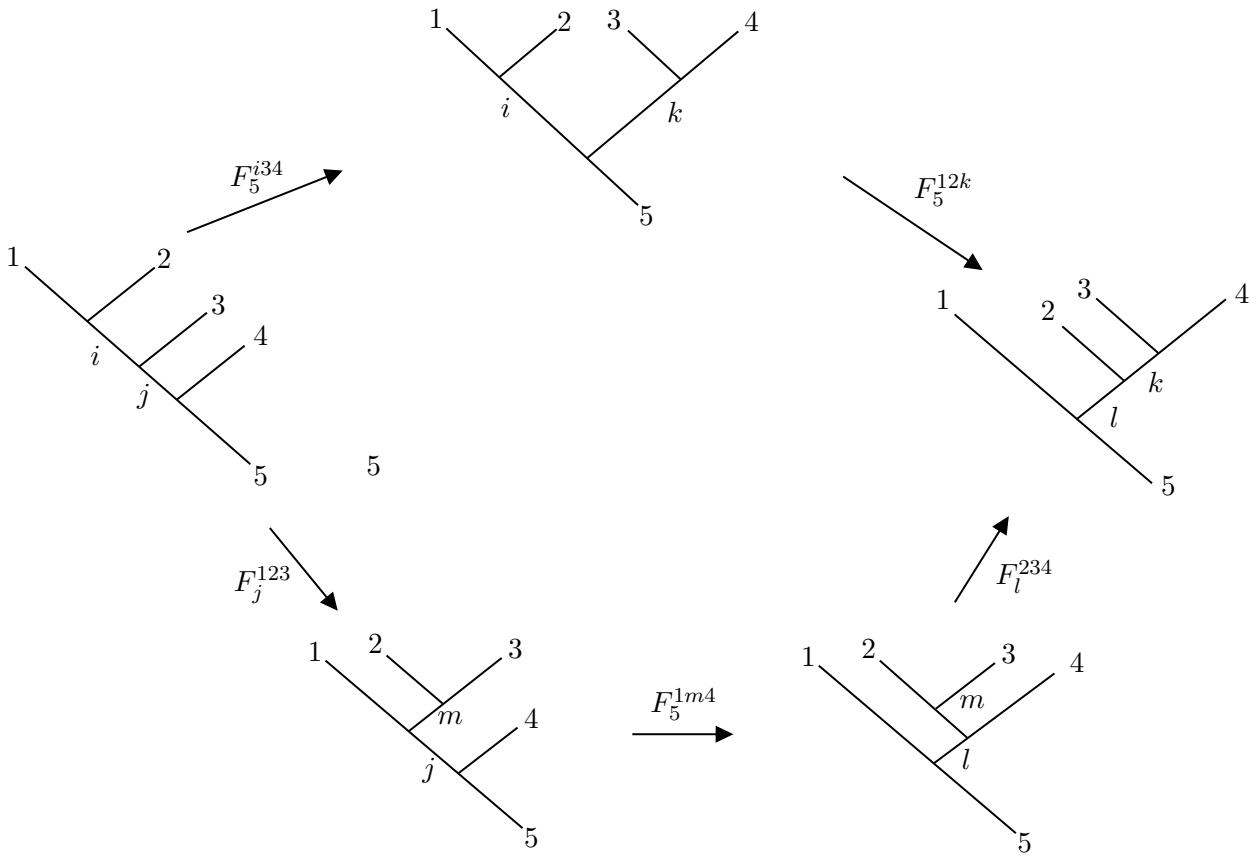


FIGURE 3.14: The Pentagon Diagram

The fact that the upper and lower sequences in the Pentagon Diagram lead to the same result gives us the following equation:

$$[F_5^{12k}]_{il}[F_5^{i34}]_{jk} = \sum_m [F_l^{234}]_{mk}[F_5^{1m4}]_{jl}[F_j^{123}]_{im} \tag{3.13}$$

which is essentially a series of polynomial equations known as the *Pentagon Equations*.
One might ask why we do not have to consider more than 4 anyons in order to arrive at further consistencies for the F matrices. A theorem from category theory known as Mac Lane's Coherence Theorem essentially guarantees that if we consider more complicated sequences, they will reduce to the Pentagon Equation [10]. Generally speaking, the Pentagon Equations are hard equations to solve. For simple anyon theories however, the fusion rules and the Pentagon Equation are sufficient to determine the F matrices. This can be done for the Fibonacci anyons and the Ising anyons [13] [7] [12] [14] [9]. In the references mentioned previously, the derivations have been done quite extensively, so we will not repeat them here. For the Fibonacci anyons, we find that the only non-trivial fusion matrix is $F_\tau^{\tau\tau\tau}$ and this is given by

$$F_\tau^{\tau\tau\tau} = \begin{pmatrix} \frac{1}{\varphi} & \frac{1}{\sqrt{\varphi}} \\ \frac{1}{\sqrt{\varphi}} & -\frac{1}{\varphi} \end{pmatrix} \tag{3.14}$$

For the Ising anyons, we find that F_4^{123} is essentially a one dimensional matrix for all cases except when the anyons 1,2,3 and 4 are all σ. For the former cases, the one-dimensional F matrix elements comes out to be -1, while $F_\sigma^{\sigma\sigma\sigma}$ comes out to be

$$F_\sigma^{\sigma\sigma\sigma} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{3.15}$$

3.3 The R Matrix

The third ingredient that we need to incorporate in order to specify a theory of anyons concerns a process known as *braiding*. When we braid two anyons, we are basically just exchanging their positions. Diagrammatically, we can depict braiding as shown below

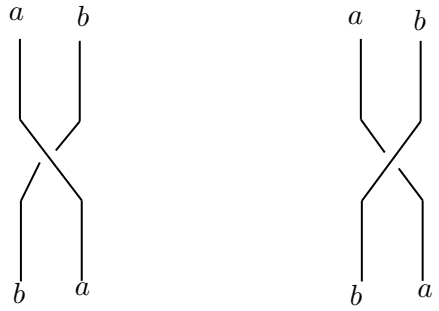


FIGURE 3.15: Two ways in which one can braid two anyons a and b. The leftmost diagram shows a clockwise braid and the rightmost diagram shows an anticlockwise braid

We have already mentioned that anyon fusion is commutative, which means that the braiding process cannot influence the final result of the fusion process. More concretely, if a and b fuse to c, and we braid a and b before the fusion, then the fusion diagram we obtain without any braiding is essentially equivalent to the fusion diagram with braiding. This equivalence is encoded using a matrix known as the R *matrix*, as shown in Figure 3.16.

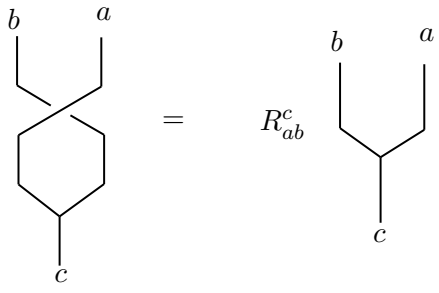
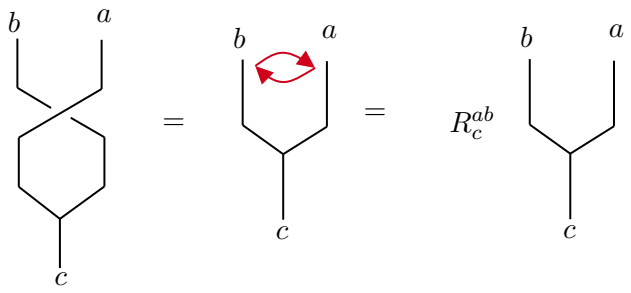


FIGURE 3.16: A fusion diagram in which anyons a and b are braided before they fuse to c is related to the fusion diagram in which they don't braid by means of the R matrix denoted by R_c^{ab}

If $N_{ab}^{c} = 1$, meaning that there is only a single option for the final anyon, then R_c^{ab} is simply just some complex phase factor. However, if the fusion multiplicities are greater than 1, which would mean that there are several ways to obtain the final anyon c as a product of the fusion process, then the R matrix is a matrix of size $N_{ab}^{c} \times N_{ab}^{c}$, and there would be a sum over c in the right hand side of Figure 3.16. A useful shorthand for indicating braiding of two anyons is given in the figure below:



3.3.1 The Hexagon Equation

Just like the F matrix, we also have a consistency condition for the R matrix. This is because of the commutativity of the diagram shown in Figure 3.17, which is known as a *hexagon diagram*. In this diagram, we have adopted the same notation as the pentagon diagram for the F matrix (naming the anyons with numbers instead of letters).

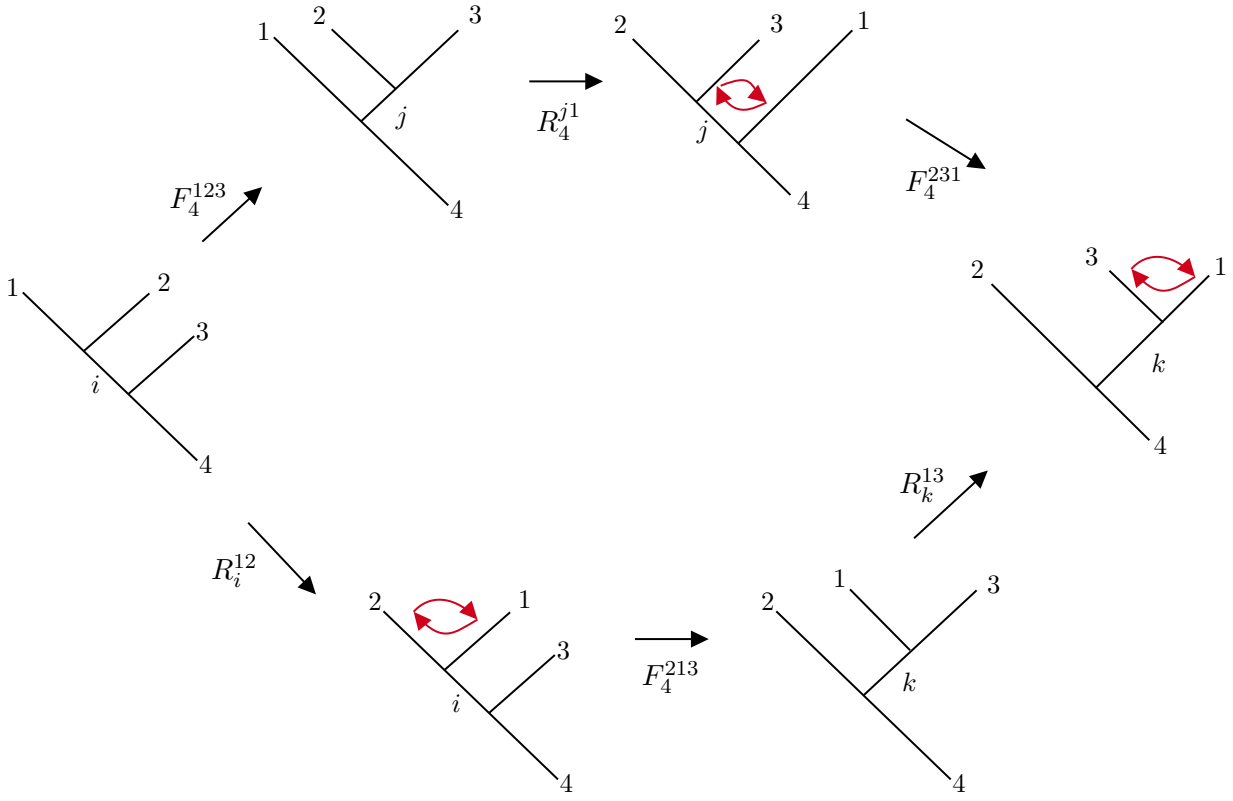


FIGURE 3.17: The Hexagon Diagram

The commutativity of the diagrams in Figure 3.17 leads to the following equation:

$$R_k^{13}[F_4^{213}]_{ki}R_i^{12} = \sum_j [F_4^{123}]_{kj}R_4^{j1}[F_4^{231}]_{ji} \tag{3.16}$$

which is known a set of polynomial equations known as the *Hexagon Equations*. This is the final ingredient we need. If we can specify a set of anyons $\mathcal{A}$ with fusion rules that satisfy the Pentagon (3.13) and Hexagon (3.16) Equations, we have constructed a consistent anyon theory. Like the Pentagon Equations, the Hexagon Equations are hard to solve. For simple anyon theories, the fusion rules and the F matrices (once they have been derived) are enough to determine the R matrices. This can again be done for the Fibonacci and Ising anyon theories [13] [7] [12] [14] [9]. In the same references as for the F matrices, an extensive derivation for the R matrices can be found. For the Fibonacci anyons, we find out that there are two possible solutions to the Hexagon Equations. One set of solutions corresponds to a "right-handed" Fibonacci anyon theory and a "left-handed" Fibonacci anyon theory. For the right-handed theory, we find that

$$R_1^{\tau\tau} = e^{-i4\pi/5} \qquad R_\tau^{\tau\tau} = e^{i3\pi/5} \tag{3.17}$$

and for all other cases, the R matrix is just 1. The values of R for the left-handed Fibonacci theory are just complex conjugates of the values of the right-handed one. For the Ising anyons, we also find two solution sets, corresponding to a left-handed and a right-handed theory. For the right-handed Ising theory we find that

$$R_1^{\sigma\sigma} = e^{-i\pi/8} \quad R_\psi^{\sigma\sigma} = e^{i3\pi/8} \quad R_1^{\sigma 1} = 1 \quad R_\sigma^{\sigma\psi} = i \tag{3.18}$$

where the values of R for the left-handed theory are again simply complex conjugates of the values for the right-handed theory.

3.4 More General Braiding

The problem we wish to consider now is how to braid any number of anyons in arbitrary ways. As an example, we will consider 3 τ anyons from the Fibonacci model. In this case, we have 3 unique fusion diagrams which correspond to 3 ket vectors which span the Hilbert space $\mathcal{H}_\tau^{(3)}$. The 3 diagrams are shown in Figure 3.18.

We will denote with $\hat{\sigma}_1$ the operator that exchanges the two leftmost particles, and with $\hat{\sigma}_2$ the operator that exchanges the two rightmost particles. Computing $\hat{\sigma}_1$ is fairly straightforward because of the fact that the two leftmost τ anyons are being directly fused with one another in each fusion diagram, which means that we simply tack on the relevant R factor to each ket

vector. This yields the following expressions for how $\hat{\sigma}_1$ operates on each ket vector:

$$\hat{\sigma}_1 |0\rangle = R_1^{\tau\tau} |0\rangle \qquad \hat{\sigma}_1 |1\rangle = R_\tau^{\tau\tau} |1\rangle \qquad \hat{\sigma}_1 |N\rangle = R_\tau^{\tau\tau} |N\rangle \tag{3.19}$$

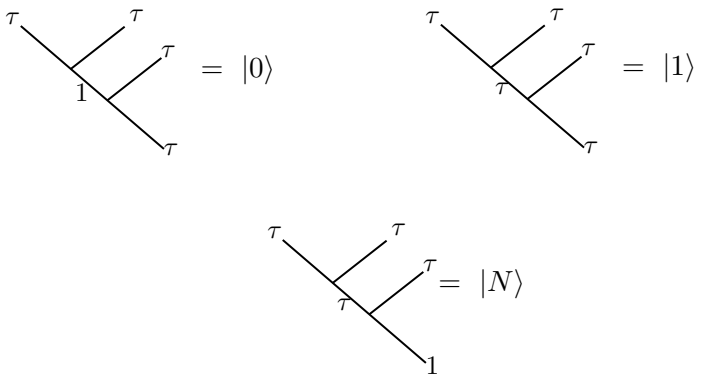


FIGURE 3.18: The state vectors corresponding to each possible fusion diagram for 3 τ anyons

The more interesting problem is determining $\hat{\sigma}_2$ because this involves two anyons which are not directly fusing with one another as shown in the diagrams. The solution is to use the F matrices to obtain fusion diagrams in which the two rightmost anyons are being directly fused together. For the vector $|N\rangle$, this is actually a lot simpler because this is the only state in which the 3 τ anyons are fusing together to the identity. It should not matter if we choose to fuse the leftmost two anyons first, or the rightmost two. In either case, the intermediate anyon has only a single possible label, τ. Thus we have that $\hat{\sigma}_2 |N\rangle = R_\tau^{\tau\tau} |N\rangle$. Hence, we only need to focus on the states $|0\rangle$ and $|1\rangle$. We first define two new ket vectors $|0'\rangle$ and $|1'\rangle$ corresponding to the following fusion diagrams:

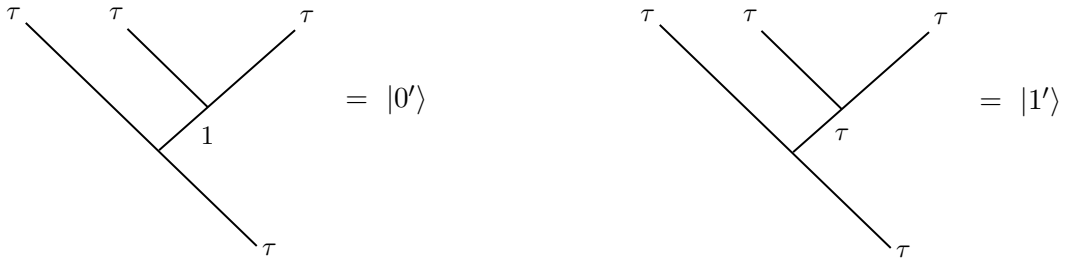


FIGURE 3.19: In the ket vectors $|0'\rangle$ and $|1'\rangle$, the two rightmost τ anyons are directly fusing with each other

Since the transformation from the unprimed vectors to the primed vectors is simply a change of basis (since it involves the F matrix), we can write

$$\begin{aligned} |0\rangle &= F_{00'} |0'\rangle + F_{01'} |1'\rangle \\ |1\rangle &= F_{10'} |0'\rangle + F_{11'} |1'\rangle \end{aligned} \tag{3.20}$$

where F_{ij} is a shorthand for $[F_\tau^{\tau\tau\tau}]_{ij}$. We can also invert this directly to express $|0'\rangle$ and $|1'\rangle$ in terms of $|0\rangle$ and $|1\rangle$ as shown below

$$\begin{aligned}\left|0'\right\rangle &= [F^{-1}]_{0'0}\left|0\right\rangle + [F^{-1}]_{0'1}\left|1\right\rangle \\ \left|1'\right\rangle &= [F^{-1}]_{1'0}\left|0\right\rangle + [F^{-1}]_{1'1}\left|1\right\rangle\end{aligned} \tag{3.21}$$

where $[F^{-1}]_{ij}$ is the inverse of the matrix F_{ij}. Now, we can finally consider $\hat{\sigma}_2\left|0\right\rangle$.

$$\begin{aligned}\hat{\sigma}_2\left|0\right\rangle &= F_{00'}\hat{\sigma}_2\left|0'\right\rangle + F_{01'}\hat{\sigma}_2\left|1'\right\rangle \\ &= F_{00'}R_1^{\tau\tau}\left|0'\right\rangle + F_{01'}R_\tau^{\tau\tau}\left|1'\right\rangle \\ &= F_{00'}R_1^{\tau\tau}\left([F^{-1}]_{0'0}\left|0\right\rangle + [F^{-1}]_{0'1}\left|1\right\rangle\right) + F_{01'}R_\tau^{\tau\tau}\left([F^{-1}]_{1'0}\left|0\right\rangle + [F^{-1}]_{1'1}\left|1\right\rangle\right) \\ &= \left(F_{00'}R_1^{\tau\tau}[F^{-1}]_{0'0} + F_{01'}R_\tau^{\tau\tau}[F^{-1}]_{1'0}\right)\left|0\right\rangle + \\ &\qquad \left(F_{00'}R_1^{\tau\tau}[F^{-1}]_{0'1} + \ F_{01'}R_\tau^{\tau\tau}[F^{-1}]_{1'1}\right)\left|1\right\rangle\end{aligned} \tag{3.22}$$

A similar calculation will essentially follow for determining $\hat{\sigma}_2\left|1\right\rangle$, where all of the terms within the parentheses are numbers we can compute from the F matrix and the relevant R matrices. All in all, we can write down the operators $\hat{\sigma}_1$ and $\hat{\sigma}_2$ as matrices in the basis $\{\left|N\right\rangle, \left|0\right\rangle, \left|1\right\rangle\}$, which yields the following two matrices:

$$\hat{\sigma}_1 = \begin{pmatrix} e^{i3\pi/5} & 0 & 0 \\ 0 & e^{-i4\pi/5} & 0 \\ 0 & 0 & e^{i3\pi/5} \end{pmatrix} \qquad \hat{\sigma}_2 = \begin{pmatrix} e^{i3\pi/5} & 0 & 0 \\ 0 & \varphi^{-1}e^{i4\pi/5} & \varphi^{-1/2}e^{-i3\pi/5} \\ 0 & \varphi^{-1}e^{-i3\pi/5} & -\varphi^{-1} \end{pmatrix} \tag{3.23}$$

The procedure used in this section to derive the matrices $\hat{\sigma}_1$ and $\hat{\sigma}_2$ can be easily generalised to compute more general braiding matrices for more general anyon theories, provided that we have that F and R matrices for that theory already.

Chapter 4

Computing with Anyons

This chapter is the answer to the following question: With our understanding of anyons from the previous chapters, how exactly can we perform quantum computing with them? First, a review of the circuit model for quantum computing will be provided, which essentially involves sequences of unitary operators to manipulate qubits and perform computations. We will then discuss how the braiding of anyons allows us to replicate these unitary operators, allowing for computation.

4.1 A Review of Quantum Circuits

The most common model for quantum computing which we have right now is called the *quantum circuit model*. In this model, we essentially manipulate a qubit by means of unitary operators which act analogously to how the logical gates operate on a bit when we study classical computation. A sequence of unitary operators can then be constructed in order to perform some sort of computation. A sample quantum circuit is shown below:

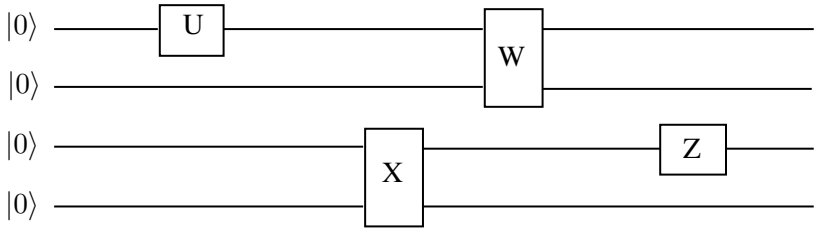


FIGURE 4.1: A sample quantum circuit

In the figure above, all of the qubits have been initialised to $|0\rangle$ and the boxes labelled are the unitary quantum gates which transform the qubits in some way to perform some computational task. The operators U and Z are single qubit gates (they act on only one qubit) whereas the

operators W and X are two quibit operators (they will act on two qubits indicated by the fact that the box intersects two lines).

4.1.1 A Quantum Circuit for Creating Entanglement

For a more concrete example, we can consider a quantum circuit that creates an entangled state from two unentangled qubits. The circuit for this is shown below:

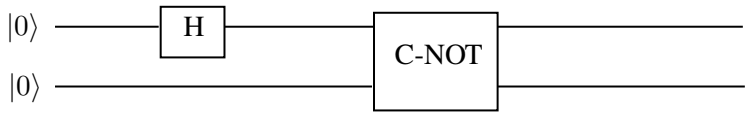


FIGURE 4.2: A quantum circuit for entangling two qubits

In the circuit above, the initial state of the two qubits is $|00\rangle$. The operator H shown in the circuit above is known as the Hadamard gate, which can be written as the following matrix in the $\{|0\rangle, |1\rangle\}$ basis:

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{4.1}$$

This operator acts on the first qubit to produce a superposition state of $|0\rangle$ and $|1\rangle$, which can be verified by explicit matrix multiplication.

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \tag{4.2}$$

So, after applying the Hadamard gate on the first qubit, the state of the two qubits is given by

$$\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) \tag{4.3}$$

which is fed into the second operator known as the controlled-NOT gate. This operator checks if the first qubit is in $|0\rangle$ or $|1\rangle$. If it is in the state $|0\rangle$, then nothing happens. If the first qubit is in the state $|1\rangle$, then we apply $\hat{\sigma}_x$ on the second qubit. Looking at Equation 4.3, it is easy to see that after applying the controlled-NOT gate, the state we obtain is

$$\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) \tag{4.4}$$

which is one of the Bell states.

4.1.2 Approximating Quantum Gates

In the circuit model, we manipulate qubits by means of unitary operators. The way this is conventionally done is by using a discrete set of unitary operators, and constructing a sequence of operators using that set which essentially mimics the action of a specific unitary operator. More concretely, let the specific unitary gate which we want to mimic be denoted by U_{target} and let the discrete set of unitary gates be the set $X = \{U_1, U_2, \ldots, U_k\}$. We construct a sequence such that $U_{\text{target}} = U_{i_m} \ldots U_{i_2} U_{i_1}$ where each $i_x \in \{1, 2, \ldots, k\}$. However, constructing such a sequence which exactly mimics U_{target} from the finite set X is difficult a lot of the time, so we usually try to approximate U_{target} up to some desired accuracy. For this, it is useful to define a distance measure between two unitary matrices that allows us to quantify the accuracy of our approximations. One such distance measure is given below [12]:

$$d(U, V) = \sqrt{1 - \frac{|\text{tr}(U^\dagger V)|}{N}} \tag{4.5}$$

where N is the size of the matrices and tr() denotes the trace operation. There are many other distance measures that may be used [8], but the one defined above will be used in this chapter. With a distance measure defined, we can now discuss the notion of approximating any unitary gate using a discrete set as accurately as we want. Let $U(N)$ denote the set of $N \times N$ unitary matrices, and $X = \{U_i \in U(N)\}_{i=1}^{k}$ be a discrete set of gates being used to approximate any $U_{\text{target}} \in U(N)$. We then say that the gate set X approximates U_{target} with an error tolerance of ϵ if there is a sequence $U_{i_m} \ldots U_{i_2} U_{i_1}$ such that

$$\text{d}(U_{\text{target}}, U_{i_m} \ldots U_{i_2} U_{i_1}) < \epsilon \tag{4.6}$$

If the above equation holds for any ϵ, then X is said to be a *universal gate set* i.e. we can approximate any $U_{\text{target}} \in U(N)$ for any desired accuracy.

4.2 Computing with Anyons

We can now look at how exactly one can use anyons to perform quantum computing. In the quantum circuit model, this involves the following steps:

1. Identify the Hilbert space which contains the qubits.
2. Initialise the physical system to some known configuration of qubits.
3. Operate the qubits with the unitary quantum gates to perform the computation.
4. Perform a measurement of some degree of freedom in the Hilbert space.

4.2.1 Hilbert Space Identification

From the discussion of the Hilbert space of anyons in Chapter 3, we know how to associate to each possible fusion diagram a ket vector. This idea is what we use to create a Hilbert space which contains the qubits for the purposes of performing computations. As an example, in Section 3.4, we considered the Hilbert space that was formed when 3 τ anyons from the Fibonacci anyon model fused together, and identified 3 state vectors: $|0\rangle$, $|1\rangle$ and $|N\rangle$ (these are also shown in Figure 3.18). The naming of these vectors was no coincidence, because aside from the state $|N\rangle$, using three τ anyons essentially allows us to define a Hilbert space containing a qubit. The state $|N\rangle$ is actually not used for any computations at all (the N here stands for non-computational). As another example, we can look at the Ising anyon theory. When we considered the quantum dimension of the σ anyon for this theory, we found out that for the case where 3 σ anyons fuse together, we have two possible fusion diagrams (these are shown in Figure 3.11). We can essentially map one diagram to $|0\rangle$ and the other to $|1\rangle$, which gives us a basis for a qubit as shown in Figure 4.3.

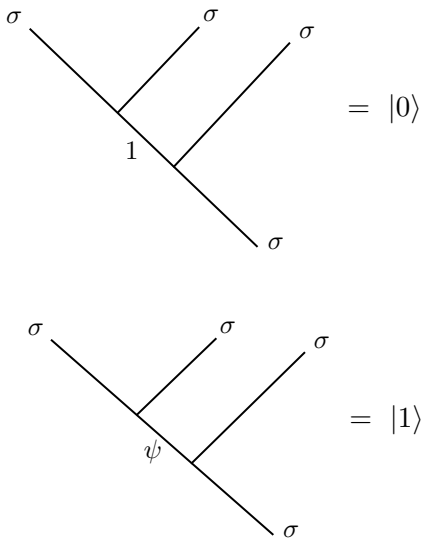


FIGURE 4.3: A possible basis for a qubit using σ anyons from the Ising theory

4.2.2 Qubit Initialisation and Measurement

The reason we are considering the problems of qubit initialisation and qubit measurement together is because if we know how to measure what quantum state the anyons are in, then initialising our qubits is fairly straightforward. Assuming that we know how to manipulate the qubits by means of unitary gates, we simply measure the qubits and check if they are in the required initial state or not. If a qubit is in the state we want, then we do nothing. If it is not, we apply the required unitary gate operations to bring it to the required initial state.
The question of how to measure our qubits when considering computing using anyons essentially

boils down to what physical system we are using. One way in which this can be done is by means of interference experiments such as one done by Willet et al. in fractional quantum Hall systems [15]. In such an experiment, we split a test anyon particle wave two partial waves that travel on opposite sides of the anyons to be measured and then re-interfere with each other. We then measure the phase difference that is accumulated by the test anyon to measure the overall fusion outcome. This is analogous to how in the Aharonov-Bohm Effect, the electron beam that splits up results in a measurable phase difference [12].

4.2.3 Performing Unitary Gate Operations

This part is where topological quantum computing gets its name. The essential idea is that we can perform unitary gate operations on the qubits by braiding the anyons present in the Hilbert space around each other. As an example, we can again consider the case of the 3 τ anyons from the Fibonacci anyon theory. For this case, we have already derived the braiding matrices $\hat{\sigma}_1$ and $\hat{\sigma}_2$ which are shown in Equation 3.23. However, since the state $|N\rangle$ plays no part in computations, we want to consider the action of the braiding operators just on the states $|0\rangle$ and $|1\rangle$. We can achieve this by remove the first diagonal entry from each matrix, since those are the ones which corresponds to how both braiding operators act on $|N\rangle$. Thus we arrive at the following 2×2 braiding matrices:

$$\hat{\sigma}_1 = \begin{pmatrix} e^{-i4\pi/5} & 0 \\ 0 & e^{i3\pi/5} \end{pmatrix} \qquad \hat{\sigma}_2 = \begin{pmatrix} \varphi^{-1}e^{i4\pi/5} & \varphi^{-1/2}e^{-i3\pi/5} \\ \varphi^{-1}e^{-i3\pi/5} & -\varphi^{-1} \end{pmatrix} \tag{4.7}$$

Looking at the $\hat{\sigma}_1$ matrix, it is immediate that we can express $\hat{\sigma}_z$) as $\hat{\sigma}_z = (\hat{\sigma}_1)^5$ with complete accuracy. For the operator $\hat{\sigma}_x$, we can make the following approximation U_{approx}

$$U_{\text{approx}} = (\hat{\sigma}_2)^{-1}(\hat{\sigma}_1)^3(\hat{\sigma}_2)^{-1} \approx e^{-i3\pi/5} \begin{pmatrix} 0.073 - 0.225i & 0.972 \\ 0.972 & -0.073 - 0.225i \end{pmatrix} \tag{4.8}$$

for which we find that $d(\hat{\sigma}_x, U_{\text{approx}}) \approx 0.17$. This makes the U_{approx} above a pretty bad approximation. In this matrix, we only performed 5 braids (since we applied $(\hat{\sigma}_2)^{-1}$ twice and $\hat{\sigma}_1$ thrice). If we include more braiding operations, we can essentially get a better and better approximation for $\hat{\sigma}_x$. If we perform 9 braiding operations and consider $U_{\text{approx}} = (\hat{\sigma}_1^{-1})^2(\hat{\sigma}_2)^3(\hat{\sigma}_1)^2(\hat{\sigma}_2^{-1})^2$, we find that $d(U_{\text{approx}}, \sigma_x) \approx 0.08$, making it a much better approximation.

4.3 The Appeal of Computing with Anyons

The previous section, particularly the section on approximating unitary gate operations using braiding operations on anyons, is the reason why topological quantum computing has appealed

researchers. For a conventional qubit, rotating the qubit may mean that some experimental procedure (such as applying a microwave field to a spin of a certain amplitude for a certain amount of time) that is prone to human error (if one mistakenly applies this field for a longer time than needed). In contrast, with quantum computing using anyons, one either does, or does not, make the right braid. These braids are said to be topological in the sense that even if the anyon is in the wrong position after performing a particular braid, the fact that the specific braiding operation has been performed does not change. The robustness of these braids (which is how computations are performed) that gives topological quantum computing resilience to noise and errors [12].

Chapter 5

Kitaev's Toric Code

Now that we know how, in principle, anyons can be used to perform quantum computing, we need to think of actual quantum systems in which anyons can arise in some way. This chapter will deal with one such system: Kitaev's Toric Code.

5.1 A Review of Quantum Error Correction

Before looking at the toric code, it will be fruitful to review some of the ideas related to quantum error correction; this will be useful when we actually introduce the Toric Code later on.

5.1.1 Classical Error Correction

A common model for errors in classical bits involves bit flipping; we say an error has occurred if the actual bit was 0 but 1 was received, or vice versa. A straightforward solution is a technique known as *repetition coding*. This involves making copies of the "logical bit" to form a "physical bit". The logical bit is what contains the information, and the physical bit is what is used to check if there is an error. As an example, we can consider the three-bit repetition code. Here, we map the logical bit 0 to the physical bit 000 and the logical bit 1 to the physical bit 111. The set $\mathcal{C} = \{000, 111\}$ is known as the *code space*, which is used to detect an error. Let x be an arbitrary bit string of length 3. If $x \in \mathcal{C}$, then no error has occurred. If $x \notin \mathcal{C}$, then we know a bit flip has occurred. After identifying whether an error has occurred or not, we would ideally like to come up with some rule that lets us decide what the value of the logical bit was given the value of the bit string x (assuming that x is neither 000 nor 111 - the decision is quite obvious in those cases). Here, we will assume that the probability of a bit flip is p, and the probability of each bit flipping in the physical bit is independent. Now, consider the random variable X

which is the number of 1s in the physical bit. We then have the following decision rule:

$$\mathrm{P}(0|X=k) \underset{1}{\overset{0}{\gtrless}} \mathrm{P}(1|X=k) \tag{5.1}$$

We can determine the value of k for which we would prefer one bit value over the other by simplifying the above expression using Baye's Rule. Before we do that however, it will be useful to derive the conditional probabilities $\mathrm{P}(X=k|0)$ and $\mathrm{P}(X=k|1)$ beforehand. The probability $\mathrm{P}(X=k|0)$ is essentially just the probability that k out of the 3 bits are flipped. Similarly, the probability $\mathrm{P}(X=k|1)$ is the probability that $3-k$ out of the 3 bits are flipped. Since we are further assuming that each bit flip is independent, we arrive at the following expressions for the conditional probabilities:

$$\mathrm{P}(X=k|0) = \binom{3}{k} p^k (1-p)^{3-k} \qquad \mathrm{P}(X=k|1) = \binom{3}{3-k} p^{3-k} (1-p)^k \tag{5.2}$$

We can now proceed with deriving the decision rule:

$$\begin{aligned}
\mathrm{P}(0|X=k) &\underset{1}{\overset{0}{\gtrless}} \mathrm{P}(1|X=k) \\
\frac{\mathrm{P}(X=k|0)\cancel{\mathrm{P}(0)}}{\cancel{\mathrm{P}(X=k)}} &\underset{1}{\overset{0}{\gtrless}} \frac{\mathrm{P}(X=k|1)\cancel{\mathrm{P}(1)}}{\cancel{\mathrm{P}(X=k)}} \\
\cancel{\binom{3}{k}} p^k (1-p)^{3-k} &\underset{1}{\overset{0}{\gtrless}} \cancel{\binom{3}{3-k}} p^{3-k} (1-p)^k \\
\left(\frac{p}{1-p}\right)^k (1-p)^3 &\underset{1}{\overset{0}{\gtrless}} p^3 \left(\frac{1-p}{p}\right)^k \\
\left(\frac{p}{1-p}\right)^k &\underset{1}{\overset{0}{\gtrless}} \left(\frac{p}{1-p}\right)^{3-k} \\
k &\underset{0}{\overset{1}{\gtrless}} 3-k \\
k &\underset{0}{\overset{1}{\gtrless}} \frac{3}{2}
\end{aligned} \tag{5.3}$$

In the above working, the factors of $\mathrm{P}(0)$ and $\mathrm{P}(1)$ cancel each other out since we are implicitly assuming that each bit value is equally likely. The binomial coefficient factors cancel because of the symmetry of the binomial coefficients. In the second last line, the inequality signs switch since $\frac{p}{1-p} < 1$; here another implicit assumption is that the probability p of a bit flip is smaller than 0.5.

The above manipulation tells us that our decision should be 1 if $k \geq 2$ and 0 otherwise (we can't have 1.5 bits after all). This result matches our intuition as well because it tells us to choose the bit value that occurs in the majority.

5.1.2 The No-Cloning Theorem

Classical error correction schemes, like the three-bit repetition code, rely on making copies of our bits. However, making copies of qubits in a similar fashion is not possible. This idea is known as the *No Cloning Theorem* in quantum mechanics [16]. The statement of the theorem is this: Suppose we have two qubits in the states $|\alpha\rangle$ and $|\beta\rangle$ where $|\alpha\rangle$ is unknown but $|\beta\rangle$ may be known or unknown. Then, there is no unitary operator $\hat{U}$ such that $\hat{U}(|\alpha\rangle\,|\beta\rangle) = e^{i\theta(|\alpha\rangle)}\,|\alpha\rangle\,|\alpha\rangle$. The proof of the theorem proceeds by contradiction. Assume that such an operator $\hat{U}$ exists. We then have that

$$\begin{aligned}\hat{U}(|\psi\rangle\,|\alpha\rangle) &= e^{i\theta(|\psi\rangle)}\,|\psi\rangle\,|\psi\rangle \\ \hat{U}(|\phi\rangle\,|\alpha\rangle) &= e^{i\theta(|\phi\rangle)}\,|\phi\rangle\,|\phi\rangle\end{aligned} \tag{5.4}$$

Now consider applying $\hat{U}$ on the state $a\,|\psi\rangle + b\,|\phi\rangle$

$$\begin{aligned}\hat{U}(\{a\,|\psi\rangle + b\,|\phi\rangle\}\,|\alpha\rangle) &= \hat{U}(a\,|\psi\rangle\,|\alpha\rangle + b\,|\phi\rangle\,|\alpha\rangle) \\ &= a\hat{U}(|\psi\rangle\,|\alpha\rangle) + b\hat{U}(|\phi\rangle\,|\alpha\rangle) \\ &= ae^{i\theta(|\psi\rangle)}\,|\psi\rangle\,|\psi\rangle + be^{i\theta(|\phi\rangle)}\,|\phi\rangle\,|\phi\rangle\end{aligned} \tag{5.5}$$

where the above manipulation follows from the fact that $\hat{U}$ must be linear. However, from the definition of $\hat{U}$ we must have that

$$\hat{U}(\{a\,|\psi\rangle + b\,|\phi\rangle\}\,|\alpha\rangle) = e^{i\theta(a|\psi\rangle + b|\phi\rangle)}[a\,|\psi\rangle + b\,|\phi\rangle][a\,|\psi\rangle + b\,|\phi\rangle] \tag{5.6}$$

which is not the same as the final expression in Equation 5.5. Hence, we cannot construct such an operator $\hat{U}$.

5.1.3 Quantum Error Correction

Despite the No-Cloning Theorem, there are ways in which quantum error correcting codes have been constructed. One example is the three-qubit flip code, which can be thought of as a quantum analogue of the classical three-bit repetition code. Here we will assume that the only possible error is a qubit bit flip (which results from an application of the Pauli matrix $\hat{\sigma}_x$). A logical qubit, denoted by $|\psi\rangle_L$ can be written as

$$|\psi\rangle_L = a\,|0\rangle_L + b\,|1\rangle_L \tag{5.7}$$

In the three-qubit flip code, we map the logical qubit into 3 physical qubits by constructing a state $|\chi\rangle$ given by

$$|\chi\rangle = a\,|000\rangle + b\,|111\rangle \tag{5.8}$$

In this scenario, the code space $\mathcal{C}$ is the Hilbert space spanned by states $|000\rangle$ and $|111\rangle$. Now, we define two operators as shown below

$$\hat{O}_{12} = \hat{\sigma}_z \otimes \hat{\sigma}_z \otimes \hat{\mathbb{1}} \qquad \hat{O}_{23} = \hat{\mathbb{1}} \otimes \hat{\sigma}_z \otimes \hat{\sigma}_z \tag{5.9}$$

Note that the state $|\chi\rangle$ is an eigenstate of both $\hat{O}_{12}$ and $\hat{O}_{23}$ with an eigenvalue of 1. However, if either one of qubit 1 or qubit 2 in the state $|\chi\rangle$ is flipped, then applying $\hat{O}_{12}$ on the state would result in an eigenvalue of -1. A similar result can be concluded for $\hat{O}_{23}$ if either qubit 2 or qubit 3 in the state $|\chi\rangle$ is flipped. This gives us the following recipe for correcting a bit flip error in our logical qubit (assuming that no more than one such error has occurred):

1. If the eigenvalue of $\hat{O}_{12}$ is -1 but the eigenvalue of $\hat{O}_{23}$ is 1 (these two conditions mean that only qubit 1 is flipped), then apply the operator $\hat{\sigma}_x \otimes \mathbb{1} \otimes \mathbb{1}$ on the state $|\chi\rangle$.
2. If the eigenvalue of both $\hat{O}_{12}$ and $\hat{O}_{23}$ is -1 (this would mean that only qubit 2 is flipped), then apply the operator $\mathbb{1} \otimes \hat{\sigma}_x \otimes \mathbb{1}$ on the state $|\chi\rangle$.
3. If the eigenvalue of $\hat{O}_{12}$ is 1 but the eigenvalue of $\hat{O}_{23}$ is -1 (these two conditions mean that only qubit 3 is flipped), then apply the operator $\mathbb{1} \otimes \mathbb{1} \otimes \hat{\sigma}_x$ on the state $|\chi\rangle$.

The three-qubit flip code is a fairly simple error correcting code, but there have been developments (spearheaded by Peter Shor and others) in constructing more sophisticated codes for quantum error correction [8].

5.2 Introducing the Toric Code

With an understanding of the basics of quantum error correction, we can now introduce the Toric Code. This is another example of a quantum error correcting code, but Alexei Kitaev recast it as a phase of matter defined by some Hamiltonian [5]. That is the route we will adopt in this thesis as well.

The system we will consider is an $N \times N$ square lattice with a qubit on each edge and periodic boundary conditions imposed on the lattice (this just means we are embedding the lattice on a torus). If we have an $N \times N$ lattice, there are a total of $2N^2$ qubits, each of which has a two dimensional Hilbert space. Thus, the dimension of the Hilbert space of the toric code is 2^{2N^2}.

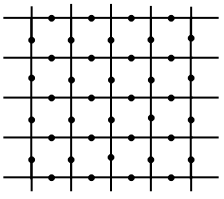


FIGURE 5.1: The toric code for $N = 5$

Instead of drawing the qubits as dots in the way done in Figure 5.1, we will adopt an alternate notation. If there is a qubit in the $|1\rangle$ state present on an edge, that edge will be coloured blue. Otherwise, an edge will remain uncoloured. This is shown in Figure 5.2 below.

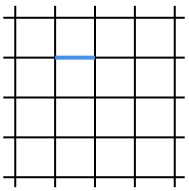


FIGURE 5.2: An example configuration for the Toric Code. There is only one qubit in the $|1\rangle$ state, while all other qubits are in the $|0\rangle$ state

The next thing we will do is to define two operators. One operator, denoted by $\hat{V}_\alpha$, is known as a *vertex operator* defined on the vertex α of the lattice. The second operator, denoted by $\hat{P}_\beta$, is known as a *plaquette operator* and is defined on the plaquette (face) β. These two operators are defined in the following way:

$$\hat{V}_\alpha = \prod_{i \in \text{vertex } \alpha} \hat{\sigma}_z^{(i)} \qquad \hat{P}_\beta = \prod_{j \in \text{plaquette } \beta} \hat{\sigma}_x^{(j)} \tag{5.10}$$

So, the vertex operator at some vertex α is a product of 4 $\hat{\sigma}_z$ operators, where each $\hat{\sigma}_z$ is applied on one of the 4 qubits that are incident on the vertex α. In a similar way, the plaquette operator at some plaquette β is a product of 4 $\hat{\sigma}_x$ operators, where each $\hat{\sigma}_x$ is applied on one of the 4 qubits that are located on the plaquette β. If we recall how $\hat{\sigma}_z$ acts on the states $|0\rangle$ and $|1\rangle$, it can be easily seen that the vertex operator yields an eigenvalue of 1 if there are an even number of blue lines incident on every vertex, and -1 if there are an odd number of blue lines. Similarly, if we recall how $\hat{\sigma}_x$ acts on the states $|0\rangle$ and $|1\rangle$, it can be easily seen that the plaquette operator switches the positions of the coloured and uncoloured lines of a particular plaquette. The action of the two operators is depicted in Figure 5.3 below.

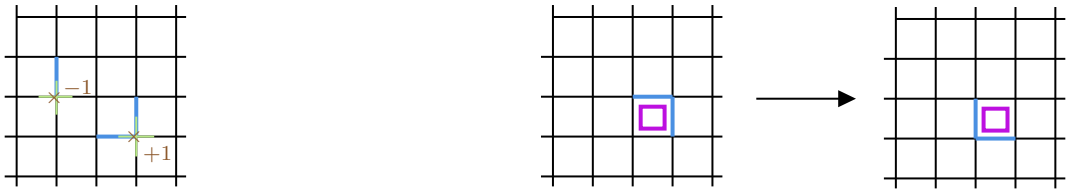


FIGURE 5.3: *Left:* The action of two vertex operators on their respective vertices (the green lines indicate the qubits on which the $\hat{\sigma}_z$ operators will act). *Right:* The action of the plaquette operator on a particular plaquette (coloured in purple).

We now mention some important properties of the vertex and plaquette operators:

1. Both operators square to the identity i.e. $(\hat{V}_\alpha)^2 = (\hat{P}_\beta)^2 = \hat{\mathbb{1}}$

2. The vertex operators at different vertices commute i.e. $[\hat{V}_\alpha, \hat{V}_{\alpha'}] = 0$
3. The plaquette operators at different plaquettes commute i.e. $[\hat{P}_\beta, \hat{P}_{\beta'}] = 0$
4. The vertex and plaquette operators commute i.e. $[\hat{V}_\alpha, \hat{P}_\beta] = 0$
5. The product of all the vertex operators is the identity i.e. $\prod_\alpha \hat{V}_\alpha = \hat{\mathbb{1}}$
6. The product of all the plaquette operators is the identity i.e. $\prod_\beta \hat{P}_\beta = \hat{\mathbb{1}}$

The first property follows from the fact that $(\hat{\sigma}_x)^2 = (\hat{\sigma}_z)^2 = \hat{\mathbb{1}}$. For the second property there are two cases: either the vertices share no edges, or they do. If the vertices don't share any edges, then the $\hat{\sigma}_z$ operators for both vertices are on distinct sites, meaning that $\hat{V}_\alpha$ and $\hat{V}_{\alpha'}$ commute. If the vertices share edge, then geometrically there is only one common edge that two vertices can share. On that common edge, the commutator would involve the same $\hat{\sigma}_z$ operator, and an operator always commutes with itself. A similar reasoning follows to show the third property, along with the observation that two plaquettes can share at most one edge. For the fourth property, we again consider two cases: either a vertex and a plaquette share no edges, or they do. If a vertex and a plaquette share no edges, then $\hat{V}_\alpha$ and $\hat{P}_\beta$ obviously commute. If a vertex and a plaquette edges, then geometrically they can only share at most 2 edges. The operators $\hat{\sigma}_x$ and $\hat{\sigma}_z$ anti-commute on any given edge, but on an even number of edges this effect cancels itself out, resulting in a zero commutator overall. The fifth property also follows from the fact that two adjacent vertices will share at most one edge, and on that common edge, we would be applying $\hat{\sigma}_z$ twice which is just the identity. When performing the product over all the vertices, we are essentially applying $\hat{\sigma}_z$ twice to each edge. A similar reasoning follows for the sixth property, with the modification being that two plaquettes also share at most one edge, and $\hat{\sigma}_x$ squares to the identity.

We now ask the question whether the vertex and plaquette operators are a complete set i.e. whether specifying the eigenstate of each of these operators leads to a unique state in the Toric Code Hilbert space. The answer is no, because of the fifth and sixth properties mentioned previously. Note that in an $N \times N$ lattice, we can have N^2 vertex and plaquette operators each. If we specify the eigenvalues of $N^2 - 1$ vertex operators, then the condition $\prod_\alpha \hat{V}_\alpha = \hat{\mathbb{1}}$ constrains the eigenvalue of the remaining vertex operator. A similar reasoning follows for the plaquette operators. So in total, we will only have specified $2(N^2 - 1)$ eigenvalues but since there are $2N^2$ qubits, we will have 2 degrees of freedom left, which go into the logical qubits that are encoded using the Toric Code.

5.3 Error Correction with the Toric Code

Now we consider how exactly one can do error correction with the Toric Code. The first thing we need to do is construct the code space $\mathcal{C}$, which contains the physical qubits.

5.3.1 Constructing the Code Space

For the Toric Code, the code space is constructed using the following two rules:

1. For each vertex α, specify that the eigenvalue of $\hat{V}_\alpha$ is 1.
2. For each plaquette β, specify that the eigenvalue of $\hat{P}_\beta$ is 1.

A consequence of the first rule is that we cannot have branches of blue lines nor any ends of lines, which means that all configurations must form closed loops. A consequence of the second rule is that we have an equal superposition of configurations where the qubits are flipped and unflipped with a plus sign between them.

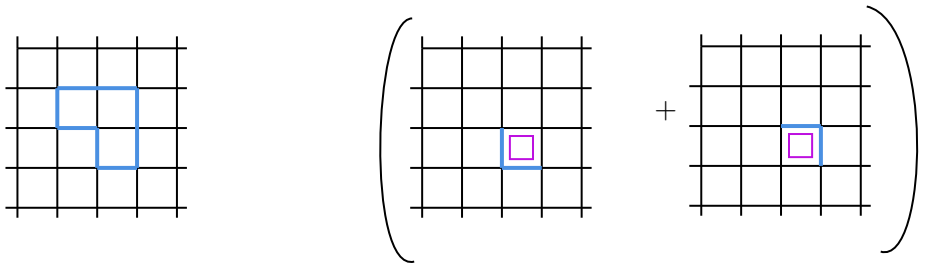


FIGURE 5.4: *Left:* A configuration that satisfies Rule 1. *Right:* A configuration that satisfies Rule 2. Observe that this configuration is an eigenvector of $\hat{P}_\beta$ on the plaquette shown in purple.

With these two rules defined, we have a recipe for constructing a state in the code space. We start with a state that satisfies the first rule. Then a possible state that satisfies both rules is given by

$$|\psi\rangle = \mathcal{N} \sum |\text{loop config}\rangle \tag{5.11}$$

where $\mathcal{N}$ is a normalisation factor and the sum is over all loop configurations that can be obtained by applying $\hat{P}_\beta$ on the plaquettes (i.e. flipping a plaquette). An example is depicted below:

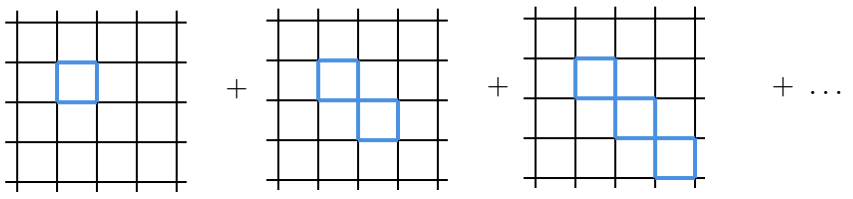


FIGURE 5.5: Constructing a state in the code space. The leftmost configuration is our reference state, and we are adding to it all possible loop configurations we can achieve by flipping plaquettes.

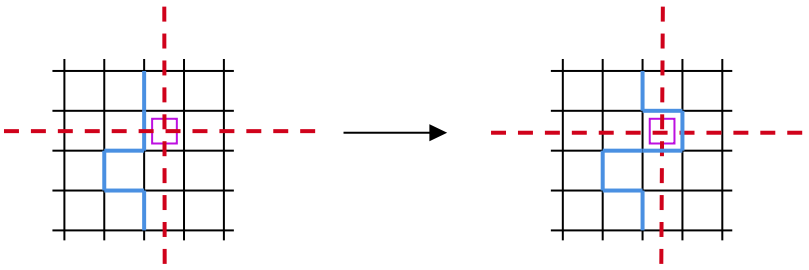


FIGURE 5.6: An initial loop configuration and the configuration obtained after performing a plaquette flip. The number of blue edges intersecting with the vertical red line is zero before the plaquette flip, and 2 after the flip. The number of blue edges intersecting with the horizontal red line is 1 before the plaquette flip, and 1 after the flip.

In hindsight, it seems like there are a lot of states in the code space that can be constructed from the recipe outlined above. However, this is actually not the case. A crucial observation that simplifies our problem greatly is that performing a plaquette flip on a loop configuration does not change the parity of the number of loops running around a handle of the torus. To see this, consider the scenario shown in Figure 5.6. A vertical handle of the torus is depicted by the vertical red dashed line, and a horizontal handle is depicted by the horizontal red dashed line. In both cases, the parity of the number of blue edges that intersect with each red dashed line remains the same. A consequence of this is that there are essentially 4 states that are in the code space:

1. A state that is a superposition of all loop configurations that have an even number of loops running around both the horizontal and vertical handles. This will be denoted by $|\psi\rangle_{ee}$.

2. A state that is a superposition of all loop configurations that have an even number of loops running around the horizontal handle and an odd number of loops running around vertical handle. This will be denoted by $|\psi\rangle_{eo}$.

3. A state that is a superposition of all loop configurations that have an odd number of loops running around the horizontal handle and an even number of loops running around vertical handle. This will be denoted by $|\psi\rangle_{oe}$.

4. A state that is a superposition of all loop configurations that have an odd number of loops running around both the horizontal and vertical handles. This will be denoted by $|\psi\rangle_{oo}$.

This aligns very nicely with the fact that for the Toric Code, we had 2 remaining degrees of freedom for our logical qubits. Now, we can encode the most general state for two logical qubits which are given by the state

$$|\psi\rangle_L = a\,|00\rangle_L + b\,|01\rangle_L + |10\rangle_L + d\,|11\rangle_L \tag{5.12}$$

into the 4 states in our code space to form two physical qubits

$$|\psi\rangle = a\,|\psi\rangle_{ee} + b\,|\psi\rangle_{eo} + c\,|\psi\rangle_{oe} + d\,|\psi\rangle_{oo} \tag{5.13}$$

5.3.2 Identifying and Correcting Errors

Now that we have constructed the code space for the Toric Code, we can look at how to identify errors in our physical qubits.

We will first consider the case where there is a $\hat{\sigma}_x$ error i.e. one of the qubits on the lattice is flipped. Let $\hat{\sigma}_x^{(i)}$ denote the $\hat{\sigma}_x$ operator that acts on the qubit living on site i. For a given site, we can find two vertices α and α' which the site is incident on. From the definition of the vertex operator in Equation 5.10 and the fact that $\hat{\sigma}_x$ and $\hat{\sigma}_z$ anti-commute, it is apparent that $\hat{\sigma}_x^{(i)}$ anti-commutes with both $\hat{V}_\alpha$ and $\hat{V}_{\alpha'}$. If we start off in the code space $\mathcal{C}$ (the eigenvalue of $\hat{V}_\alpha$ is 1 for all vertices), then applying $\hat{\sigma}_x^{(i)}$ on a qubit living on site i results in an eigenvalue of -1 for $\hat{V}_\alpha$. A similar result follows for $\hat{V}_{\alpha'}$. More explicitly, we can perform the following manipulation

$$\begin{aligned} |\psi\rangle \in \mathcal{C} &\Rightarrow \hat{V}_\alpha\,|\psi\rangle = |\psi\rangle \\ &\Rightarrow \hat{\sigma}_x^{(i)}\,|\psi\rangle = \hat{\sigma}_x^{(i)}(\hat{V}_\alpha\,|\psi\rangle) = -\hat{V}_\alpha(\hat{\sigma}_x^{(i)}\,|\psi\rangle) \\ &\Rightarrow \hat{V}_\alpha(\hat{\sigma}_x^{(i)}\,|\psi\rangle) = -\hat{\sigma}_x^{(i)}\,|\psi\rangle \end{aligned} \tag{5.14}$$

which repeats for the vertex α'. We can also depict this graphically as shown below:

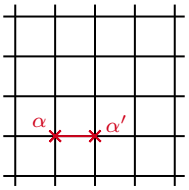


FIGURE 5.7: A $\hat{\sigma}_x$ error applied on a qubit living on the red line results in an error on the vertices α and α'. The eigenvalues of $\hat{V}_\alpha$ and $\hat{V}_{\alpha'}$ are -1 after $\hat{\sigma}_x$ is applied.

In order to correct a $\hat{\sigma}_x$ error then, we essentially just need to check if every adjacent pair of vertices have an eigenvalue of 1. If this is the case, we are in the code space. If we find a pair of adjacent vertices for which the corresponding vertex operators have an eigenvalue of -1, we are not in the code space and have to apply $\hat{\sigma}_x$ on the qubit living on the site corresponding to both vertices.

Now, we can consider a $\hat{\sigma}_z$ error. Let $\hat{\sigma}_z^{(i)}$ denote the $\hat{\sigma}_z$ operator that acts on the qubit living on site i. For a given site, we can find two adjacent plaquettes β and β' which share the site. From the definition of the plaquette operator in Equation 5.10 and the fact that $\hat{\sigma}_x$ and $\hat{\sigma}_z$ anti-commute, it is apparent that $\hat{\sigma}_z^{(i)}$ anti-commutes with both $\hat{P}_\beta$ and $\hat{P}_{\beta'}$. If we start off in the code space $\mathcal{C}$ (the eigenvalue of $\hat{P}_\beta$ is 1 for all plaquettes), then applying $\hat{\sigma}_z^{(i)}$ on a qubit living

on site i results in an eigenvalue of -1 for $\hat{P}_\beta$. A similar result follows for $\hat{P}_{\beta'}$. More explicitly, we can perform the following manipulation:

$$\begin{aligned} |\psi\rangle \in \mathcal{C} &\Rightarrow \hat{P}_\beta |\psi\rangle = |\psi\rangle \\ &\Rightarrow \hat{\sigma}_z^{(i)} |\psi\rangle = \hat{\sigma}_z^{(i)}(\hat{P}_\beta |\psi\rangle) = -\hat{P}_\beta(\hat{\sigma}_z^{(i)} |\psi\rangle) \\ &\Rightarrow \hat{P}_\beta(\hat{\sigma}_z^{(i)} |\psi\rangle) = -\hat{\sigma}_z^{(i)} |\psi\rangle \end{aligned} \tag{5.15}$$

which repeats for the plaquette β'. We can also depict this graphically as shown in Figure 5.8. Similar to the $\hat{\sigma}_x$ error case, the idea is to basically check if for any two adjacent plaquettes, the corresponding plaquette operators have an eigenvalue of -1. If this is the case, then we apply $\hat{\sigma}_z$ to the intervening site to correct the error.

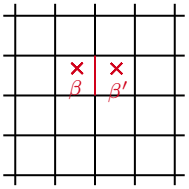


FIGURE 5.8: A $\hat{\sigma}_z$ error applied on a qubit living on the red line results in an error on the plaquettes β and β'. The eigenvalues of $\hat{P}_\beta$ and $\hat{P}_{\beta'}$ are -1 after $\hat{\sigma}_z$ is applied.

We can also consider the case where we have a $\hat{\sigma}_x$ error and a $\hat{\sigma}_z$ error on the same qubit, depicted in Figure 5.9. The solution to this is also pretty straightforward: apply $\hat{\sigma}_x$ to correct the $\hat{\sigma}_x$ error and $\hat{\sigma}_z$ to correct the $\hat{\sigma}_z$ error. The only thing we would need to be careful of is the order in which we apply these two operators because $\hat{\sigma}_x\hat{\sigma}_z = -\hat{\sigma}_z\hat{\sigma}_x$.

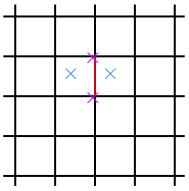


FIGURE 5.9: Applying both a $\hat{\sigma}_x$ error and a $\hat{\sigma}_z$ error on the qubit living on the red line results in an error on the two vertices indicated by purple crosses as well as an error on the two plaquettes indicated by the blue crosses.

One key thing to note about the error correction scheme with the Toric Code is that the vertex and plaquette errors (sometimes also called vertex and plaquette defects) which arise as a result of $\hat{\sigma}_x$ and $\hat{\sigma}_z$ errors occur in pairs. A $\hat{\sigma}_x$ error results in two vertex operators having an eigenvalue of -1, and a $\hat{\sigma}_z$ error results in two plaquette operators having an eigenvalue of -1. Furthermore, a small comment that should be noted is that we can make the $\hat{\sigma}_x$ and $\hat{\sigma}_z$ errors more complicated by applying these operators on more than 1 qubit. The interested reader may consult [12] for how to deal with these more complicated errors. The simpler errors we have considered here are sufficient to get an overall idea of the error correction scheme.

5.4 Recasting the Toric Code

So far, the we have looked at the Toric Code as a quantum error correcting code. We now show how the Toric Code can arise as a quantum phase of matter i.e. the ground state of this Hamiltonian. We define the following Hamiltonian for this purpose:

$$\hat{H} = -\sum_{\text{vertices } \alpha} \hat{V}_\alpha - \sum_{\text{plaquettes } \beta} \hat{P}_\beta \tag{5.16}$$

where $\hat{V}_\alpha$ and $\hat{P}_\beta$ are the same vertex and plaquette operators that we have been working with so far. The objective is to find the ground state of this Hamiltonian. Recall that $(\hat{V}_\alpha)^2 = (\hat{P}_\beta)^2 = \hat{\mathbb{1}}$, from which we concluded that the only possible eigenvalues for both operators are ± 1. Since $[\hat{V}_\alpha, \hat{P}_\beta] = 0$, we can attempt to find simultaneous eigenstates of $\hat{V}_\alpha$ and $\hat{P}_\beta$ that minimise $\hat{H}$ (since we are searching for the ground state). Noting the minus sign in both summations in the expression for $\hat{H}$ in Equation 5.16, it is apparent that the ground state corresponds to the states such that the eigenvalue of $\hat{V}_\alpha$ at each vertex α and the eigenvalue of $\hat{P}_\beta$ at each plaquette β is 1. However, this is just the code space we constructed previously! Thus, the ground state of $\hat{H}$ is 4-fold degenerate. Furthermore, when we constructed the code space states, we concluded that they depend on the parity of the number of loops running around the handle of the torus. Within the context of the ground state of $\hat{H}$, this has the consequence that we cannot distinguish between the states making up the ground state by a local measurement. The only way one can distinguish the state $|\psi\rangle_{ee}$ and $|\psi\rangle_{oo}$ is by traveling all around the torus and counting the number of loops running around the horizontal and vertical handles. This is a manifestation of the phenomenon known as *topological order* that is quite common with many-body systems like the Toric Code.

We can also consider what the excited states of the Hamiltonian defined in Equation 5.16 are. We know that if at a vertex α is such that the eigenvalue of $\hat{V}_\alpha$ is -1, then we are no longer in the ground state of $\hat{H}$. A similar reasoning follows for the eigenvalue of $\hat{P}_\beta$ at a plaquette β. Previously, we interpreted these occurrences as errors but now, they correspond to excited states of $\hat{H}$. We can list the possible excitations we can obtain as well:

1. If we are in the ground state, we can interpret that as no excitation or particle i.e. the identity particle 1.
2. We can have a vertex α such that the eigenvalue of $\hat{V}_\alpha$ is -1. We denote this excitation with the symbol e.
3. We can have a plaquette β such that the eigenvalue of $\hat{P}_\beta$ is -1. We denote this excitation with the symbol m.

4. We can also have a vertex error and a plaquette error which we interpret as the anyon fusion of e and m. We denote this excitation with the symbol $f = e \times m$.

Hence the Toric Code Hamiltonian gives rise to a model of anyons specified by the set $\mathcal{A} = \{1, e, m, f\}$. We can also work towards determining the fusion rules of the anyons present in $\mathcal{A}$. When we considered the $\hat{\sigma}_x$ and $\hat{\sigma}_z$ errors of the Toric Code, we saw that the vertex and plaquette errors came in pairs, and applying the relevant operator corrected the eigenvalues of the pair. This motivates us to think of the e and m particles as being created and annihilated in pairs. This leads to the following two fusion rules:

$$e \times e = 1 \qquad m \times m = 1 \tag{5.17}$$

The fusion rules in Equation 5.17, along with the commutativity and associativity of anyon fusion, further allow us to determine the outcome of the fusion $f \times f$, as shown below:

$$\begin{aligned} f \times f &= (e \times m) \times (e \times m) \\ &= (e \times m) \times (m \times e) \\ &= e \times (m \times m) \times e \\ &= e \times 1 \times e = e \times e = 1 \end{aligned} \tag{5.18}$$

The only two fusion rules that we need to derive now are $f \times e$ and $f \times m$. If we consider $f \times e$ we obtain:

$$\begin{aligned} f \times e &= (e \times m) \times e \\ &= (m \times e) \times e \\ &= m \times (e \times e) \\ &= m \times 1 = m \end{aligned} \tag{5.19}$$

Similarly, if we consider $f \times m$ we obtain:

$$\begin{aligned} f \times m &= (e \times m) \times m \\ &= e \times (m \times m) \\ &= e \times 1 = e \end{aligned} \tag{5.20}$$

Equations 5.17, 5.18, 5.19 and 5.20 are all of the nontrivial fusion rules for the anyonic excitations in the Toric Code. The reason why recasting the Toric Code is significant is because it allows for the possibility of simulating the it in quantum computers and working towards integrating the error correction provided by this system to already existing quantum computing architectures; Google recently simulated the Toric Code on their quantum processors [1].

Chapter 6

Conclusion

The main aim of this thesis was to try and provide an entry-level introduction to some of the key ideas within topological quantum computing. A lot of the focus was dedicated to trying to explain the argument put forward by Leinaas and Myrheim as to why anyons must, theoretically speaking, be a possibility and in trying to explain how to actually work with anyons and how to think about them. Where it was felt that an example would be helpful in allowing the reader to understand some of the ideas within this field that at first glance seem weird and counter-intuitive, an attempt has been made to provide one. However, this thesis is by no means a complete review of the field or a set of lecture notes on the subject. One shortcoming of this work was not looking at an example of a quantum system with non-Abelian anyons as their excitations. Such systems are actually much more relevant for quantum computing purposes, and include the one-dimensional Kitaev Chain [11] as well as the Kitaev Honeycomb Lattice [4] [9]. Another topic for exploration that has not been explored in depth in this thesis is investigating actual condensed matter systems such as Fractional Quantum Hall systems [13] to consider experimental attempts to detect and manipulate anyons. Nonetheless, we hope that this thesis ends up being useful for someone to familiarise themselves with the crucial ideas that dominate the subject.

Bibliography

[1] Benedikt Fauseweh. "Quantum many-body simulations on digital quantum computers: State-of-the-art and future challenges". In: *Nature Communications* 15.1 (2024), p. 2123. DOI: 10.1038/s41467-024-46402-9. URL: https://www.nature.com/articles/s41467-024-46402-9.

[2] David J. Griffiths and Darrell F. Schroeter. *Introduction to quantum mechanics*. Third edition. Cambridge ; New York, NY: Cambridge University Press, 2018. ISBN: 978-1-107-18963-8.

[3] Suyeon Khim. *The Frobenius-Perron Theorem*. 2007. URL: https://www.math.uchicago.edu/~may/VIGRE/VIGRE2007/REUPapers/FINALAPP/Khim.pdf.

[4] Alexei Kitaev. "Anyons in an exactly solved model and beyond". In: *Annals of Physics* 321.1 (Jan. 2006), 2–111. ISSN: 0003-4916. DOI: 10.1016/j.aop.2005.10.005. URL: http://dx.doi.org/10.1016/j.aop.2005.10.005.

[5] A.Yu. Kitaev. "Fault-tolerant quantum computation by anyons". In: *Annals of Physics* 303.1 (Jan. 2003), 2–30. ISSN: 0003-4916. DOI: 10.1016/s0003-4916(02)00018-0. URL: http://dx.doi.org/10.1016/S0003-4916(02)00018-0.

[6] J. M. Leinaas and J. Myrheim. "On the theory of identical particles". In: *Nuovo Cim. B* 37 (1977), pp. 1–23. DOI: 10.1007/BF02727953.

[7] Yingkai Liu. *Introduction to Topological Quantum Computation: Ising Anyons Case Study*. https://yk-liu.github.io/2019/Introduction-to-QC-and-TQC-Ising-Anyons/. Accessed: 2024-05-15. 2019.

[8] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010.

[9] Jiannis K. Pachos. *Introduction to Topological Quantum Computation*. Cambridge University Press, 2012.

[10] John Preskill. *Lecture Notes for Physics 219: Quantum Computation*. URL: http://theory.caltech.edu/~preskill/ph229/.

[11] Sumathi Rao. *Introduction to abelian and non-abelian anyons*. 2016. arXiv: 1610.09260 [cond-mat.mes-hall].

[12] Steven H. Simon. *Topological Quantum*. Oxford University Press, Sept. 2023. ISBN: 978-0-19-888672-3.

[13] David Tong. *Lectures on the Quantum Hall Effect*. 2016. arXiv: 1606.06687 [hep-th].

[14] Simon Trebst et al. "A Short Introduction to Fibonacci Anyon Models". In: *Progress of Theoretical Physics Supplement* 176 (2008), 384–407. ISSN: 0375-9687. DOI: 10.1143/ptps.176.384. URL: http://dx.doi.org/10.1143/PTPS.176.384.

[15] R. L. Willett et al. "Interference Measurements of Non-Abelian $e/4$ amp; Abelian $e/2$ Quasiparticle Braiding". In: *Phys. Rev. X* 13 (1 2023), p. 011028. DOI: 10.1103/PhysRevX.13.011028. URL: https://link.aps.org/doi/10.1103/PhysRevX.13.011028.

[16] W. K. Wootters and W. H. Zurek. "A single quantum cannot be cloned". In: *Nature* 299.5886 (1982), pp. 802–803. ISSN: 1476-4687. DOI: 10.1038/299802a0. URL: https://doi.org/10.1038/299802a0.