Physical Applications of Topological Quantum Field Theory

Topological Quantum Computing

Lukas Hahn

1 Introduction 1

2 Qubits and Universality 1

3 Simulating Quantum Physics 6

4 Abstract Anyonsense? 7

5 Anyons vs. Qubits 13

1 Introduction

Throughout the seminar, we encountered TQFT's with anyonic excitations in various physical manifestations and studied their properties. Many of these properties are shared by all examples which hints at a univeral description of the corresponding models.
The aim of this talk is to explain how such an abstraction allows for the description of a computation scheme whose computational power is comparable to that of a quantum computer. After a lightning review of relevant concepts in classical theoretical computer science, quantum computation and related issues are discussed before introducing the idea behind topological quantum computation, its limitations and advantages.
The talk and these notes are almost entirely based on the lecture notes of John Preskill [1][2][3] and Zhenghan Wang [4].

2 Qubits and Universality

Classical Computation

In theoretical computer science any kind of information is classically encoded in bit strings

$$x \in \mathbb{Z}_2^n \tag{2.1}$$

and the computers‘ task is to calculate boolean maps

$$f : \mathbb{Z}_2^n \longrightarrow \mathbb{Z}_2^m \tag{2.2}$$
$$x \longmapsto f(x). \tag{2.3}$$

Physically, this is realized by assembling a device with some sort of input- and output-values and whose time evolution reproduces the value of the function f.

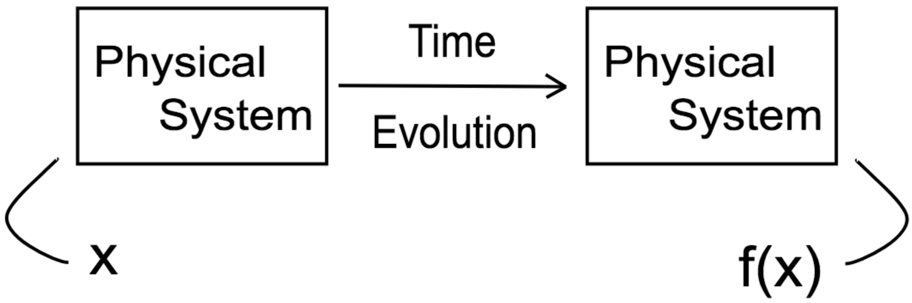


Figure 2.1: Naive sketch of a computer.

Basic examples for the boolean maps are simple logical operations like

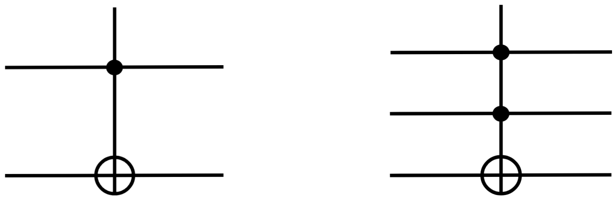
$$f(x, y) = x \cdot y = x \wedge y \qquad (\text{"and"}) \tag{2.4}$$
$$g(x, y) = x + y - x \cdot y = x \vee y \qquad (\text{"or"}) \tag{2.5}$$

but also more elaborate ones 2.2 like the controlled-not (a) and the (controlled)^2-not (b) obeying the rules "If the first bit is one, switch the second" and correspondingly for the second case. The logical operation (b) is known as the *Toffoli-gate*.
These building blocks from which computations are put together are referred to as *logical gates*. Of special interest is the following notion.

Definition 2.1. A *universal set of gates* is a finite set of gates from which all other gates can be constructed by compositions. A single gate is called *universal*, if it alone does the job.

Examples are the NAND-gate

$$h(x, y) = 1 - xy = \neg(x \wedge y) \tag{2.6}$$



(a) (b)

Figure 2.2: The CNOT- and Toffoli-gates.

and the Toffoli gate in the reversible case ($n = m$ in (2.2)). The crucial observation is that due to the existence of universal sets of gates, any function $f : \mathbb{Z}_2^n \to \mathbb{Z}_2^m$ can be evaluated by a sequence of gates for any input value. This motivates the circuit model of classical computation.

Definition 2.2. A *circuit* is a finite sequence of gates. A *computation* is a circuit applied to a string of input bits. A *universal computer* is a device which is capable of performing computations by circuits which consist of universal sets of gates.

In particular, a universal computer is equivalent to a Turing machine.

Quantum Computing

The idea behind a quantum computer is to realize the computation of the boolean map in terms of a quantum system as a device in the real world. In the quantum system input and output cannot be encoded in usual bit strings, but rather quantum states which we will think of as generated by a basis that is given by the classical bits

$$(\mathbb{C}^2)^{\otimes n} \simeq \mathbb{C}[\mathbb{Z}_2^n]. \tag{2.7}$$

Instead of a vector in the original vector space, we will get a quantum state as carrier of information

$$x \in \mathbb{Z}_2^n \rightsquigarrow |x\rangle \in \mathbb{C}[\mathbb{Z}_2^n]. \tag{2.8}$$

Ideally, the time evolution and therefore the computation itself is determined by

$$U_x|x\rangle = |f(x)\rangle, \tag{2.9}$$

where U_x is a unitary matrix.

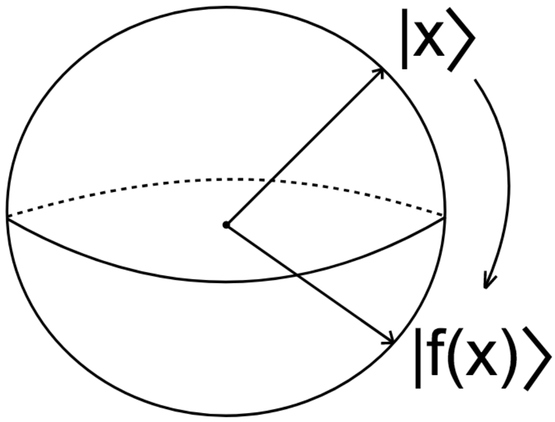


Figure 2.3: The action of the unitary matrix corresponds to a rotation along the Bloch sphere.

While a bit can be thought of as an on-off switch in a classical system, a "qubit" will be a two-level quantum system with properties that capture this intuition.

Definition 2.3. A *qubit* is a

- Hilbert space of states $\mathcal{H} = \mathbb{C}[\mathbb{Z}_2]$
- with time evolution given by U(2)-matrices
- and measurements given by 2×2 Hermitian operators together with a probabilistic interpretation.

A *qubit-state* is an element

$$|x\rangle \in \mathbb{C}[\mathbb{Z}_2]. \tag{2.10}$$

An n-qubit is a vector space

$$\mathbb{C}^{2n} \tag{2.11}$$

together with a generalization of the notions above, it is the analog of a string of bits in a classical computer. A n-qubit state is usually written as, e.g.

$$|0100010110\rangle \in \mathbb{C}^{20}, \tag{2.12}$$

in terms of a basis constructed out of ordinary bits. Importantly, the n-qubit allows for a natural decomposition into subspaces

$$\underbrace{\mathbb{C}^2 \otimes \cdots \otimes \mathbb{C}^2}_{n\text{-times}} \tag{2.13}$$

that correspond to single qubits. This is related to the physical requirement of spatial locality of the quantum system. In analogy to the classical computer, the unitary transformations are called *quantum gates* and a *quantum circuit* corresponds to a sequence of quantum gates, applied to an initial input qubit

$$\underbrace{\cdots U_{x''} U_{x'} U_x}_{\text{quantum circuit}} |x\rangle. \tag{2.14}$$

Not quite analogous, but adjusted to the fact that there is a continuum of quantum gates is the notion of universality:

Definition 2.4. A set of quantum gates $\{U_i\}_{i\in I}$ is called *universal*, if it is able to achieve for any given $\widetilde{U} \in U(2^n)$ and error threshold $\delta > 0$

$$|U_{i_{k_1}} \cdots U_{i_{k_m}} - \widetilde{U}| < \delta, \tag{2.15}$$

for some elements $U_{i_{k_1}}, ..., U_{i_{k_m}}$ of the set.
A *universal quantum computer* is a device which is capable of performing computations by circuits which consist of universal sets of quantum gates.
A universal quantum computer is sometimes referred to as *quantum Turing machine*.

For physical applications not only the existence of the relevant subset of quatum gates is sufficient, but also the possibility to implement these in the physical system in the real world. To this end, a usual demand is the possibility for a classical Turing machine to find the sequence $U_{i_{k_1}}, ..., U_{i_{k_m}}$ in polynomial time.
The basic example for a universal quantum gate is the "generic" k-qubit gate: It is given by a $2^k \times 2^k$ unitary matrix U with eigenvalues

$$e^{i\theta_1}, ..., e^{i\theta_{2^k}} \tag{2.16}$$

with the θ_i's being irrational multiples of π as well as θ_i/θ_j being irrational for $i \neq j$. The need for the irrationality is due to the fact that for $n \in \mathbb{N}$ the set of eigenvalues of U^n

$$e^{in\theta_1}, ..., e^{in\theta_{2^k}} \tag{2.17}$$

densely fills the torus T^{2^k}, which is the important ingredient in the proof of the statement (2.15) for the 2-qubit gate [2].
A variant of the above is the *Deutsch-gate*[1], which was the first proposal for a universal quantum gate. It is given by a (controlled)^2-R gate

[1]David Deutsch, 1989

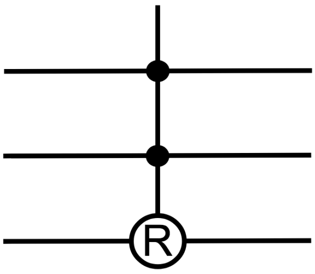


Figure 2.4: Deutsch gate.

with

$$R = -i \exp\left(i\frac{\theta}{2}\sigma_x\right), \tag{2.18}$$

and similarly for σ_y, σ_z, corresponding to a rotation on the Bloch-sphere.

3 Simulating Quantum Physics

One of the problems that physicists have with ordinary universal computers is the inability to similate a quantum physical process in reasonable running time [5]: A classical Turing machine experiences exponential slowdown when simulating quantum phenomena. A typical example is a (k-)local quantum system with Hamiltonian

$$H = \sum_{i=1}^{L} H_i \tag{3.1}$$

on an n-qubit system with initial state $|\psi_0\rangle$ and error threshold $\delta > 0$. The computers' task is to find a state $|\widetilde{\psi}(t)\rangle$ by means of a computation with a quantum circuit, such that

$$|\langle\widetilde{\psi}(t)|e^{\frac{-i}{\hbar}Ht}|\psi_0\rangle|^2 \geq 1 - \delta. \tag{3.2}$$

It was conjectured by Richard Feynman [5] that there is a quantum circuit of polynomial running time that solves the simulation problem. This conjecture was proven 1996 by Seth Lloyd [6].

Sketch of proof 3.1. The proof is based on the Trotter- Product formula for Hermitian operators A and B

$$\lim_{n\to\infty}\left(e^{i\frac{At}{n}}e^{i\frac{Bt}{n}}\right)^n = e^{i(A+B)t}. \tag{3.3}$$

If we slice up the relevant time interval into n slices of length Δt we can approximate the time evolution operator above as [7]

$$e^{\frac{i}{\hbar}Ht} = \left(\prod_{i=1}^{L} e^{\frac{i}{\hbar}H_i\Delta t}\right)^n + \sum_{j'>j}[H_{j'}, H_j]\frac{t^2}{2n} + \text{higher order terms}. \tag{3.4}$$

The higher order terms of order k are bounded by $n||Ht/n||^k/k!$, $||\cdot||$ denoting the operator norm, and for a given error $\epsilon \propto t^2/n$ taking n big enough ($n > t^2/\epsilon$) the error can be made arbitrarily small.
By this approximation, a candidate for a universal set of quantum gates is $\{\exp(iH_i\Delta t/\hbar)\}_i$, which indeed has the hoped for scaling properties with respect to system size [8].

In addition to the vast range of physical applications, the concept of simulation provides a tool for comparison of given computer systems of any kind: If a computer can be simulated by another one, the latter has at least not less computational power. E.g. a classical universal computer can be simulated, through construction of the Toffoli-gate via Deutsch-gates, by a universal quantum computer, but not the other way round. Therefore, quantum computation is more powerful than classical computation. We will follow a similar strategy in case of the topological quantum computer.
However, it is obvious that Lloyd‘s strategy does not apply to topological quantum field theories with Hamiltonian $H = 0$, where another approach is needed.

4 Abstract Anyonsense?

As already spoiled in the introduction, we want to construct a computer device out of a topological quantum field theory and anyons. Given the many different systems where these concepts are applicable, it seems to be exhausting to start with an explicit model and test its computational power. As it turns out, it is very convenient to extract the relevant properties that are shared by all these models and formulate an abstract framework that allows for the most general treatment.
Consider a topological quantum system on a connected surface Σ_g with (possibly empty) boundary of genus g. The lowest lying energy states of the theory correspond to a Hilbert space that decomposes into local and global degrees of freedom:

$$L(\Sigma_g) = V(\Sigma_g) \otimes V^{\text{local}}(\Sigma_g). \tag{4.1}$$

The usual features of such a Hilbert space is a degenerate groundstate incorporating all topologically equivalent configurations, an energy gap between the ground state and the first excitation level as well as the ability to normalize the Hamiltonian such that $H = 0$.
Anyons come into the picture by allowing for excitations on fixed points $p_1, ..., p_n$ on the surface Σ_g.

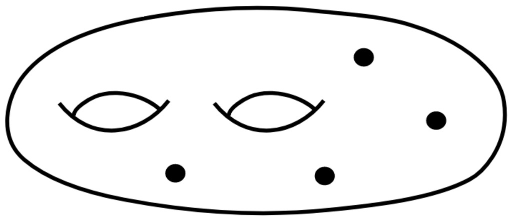


Figure 4.1: Anyonic excitations as punctures on a surface.

With respect to a single puncture, the decomposition now reads

$$L(\Sigma_g, p_i) = V(\Sigma_g, p_i) \otimes V^{\text{local}}(\Sigma_g, p_i), \tag{4.2}$$

where the local degrees of freedom are determined by a small neighbourhood around the puncture, which of course differs from puncture to puncture. Not so for the global Hilbert space: It is determined by the topological information of the whole surface, together with the data of the punctures. We therefore have an assignment

$$(\Sigma_g, p_1, ..., p_n) \longmapsto V(\Sigma_g, p_1, ..., p_n). \tag{4.3}$$

Due to $H = 0$, this global theory has no continuous dynamics. The only nontrivial transformations of the Hilbert space must correspond to diffeomorphisms that act on Σ_g in a way which cannot be homotopically deformed to the identity. These transformations are given by the *mapping class group* which acts on the surface, together with its punctures, by these diffeomorphisms

$$\Gamma(\Sigma_g, p_1, ..., p_n) := {}^{\text{Diff}(\Sigma_g, p_1, ..., p_n)}\!\big/_{\text{Diff}_0(\Sigma_g, p_1, ..., p_n)} \curvearrowright (\Sigma_g, p_1, ..., p_n), \tag{4.4}$$

where the subscript denotes "homotopic to the identity".
Examples of mapping class groups and their generators are in case of genus g surfaces the *Dehn-Twists*

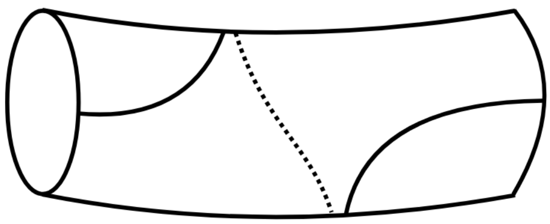


Figure 4.2: For any genus g surface, the Dehn twists are obtained by cutting out an annulus, twisting it along one of the generators of $\pi_1(\Sigma_g)$ and glueing it back in.

and in case of the punctured disc the braid group $\mathcal{B}_n$.

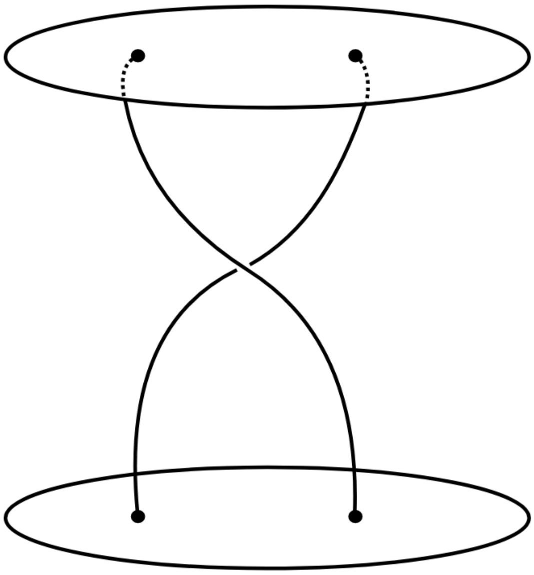


Figure 4.3: Movie of a diffeomorphism of the punctured disc and the emerging braid.

The punctured disc and the braid group is the familiar example that arose in the context of anyons in previous talks, where it came along with a projective representation

$$\rho : \mathcal{B}_n \longrightarrow \mathrm{U}(V(\mathrm{Disc})). \tag{4.5}$$

But in the more general setting we would in analogy conclude that there should be a projective representation

$$\rho : \Gamma(\Sigma_g, p_1, ..., p_n) \longrightarrow \mathrm{U}(V(\Sigma_g, p_1, ..., p_n)). \tag{4.6}$$

The assignment (4.3) and the representation (4.6) are the essential ingredients of a functor from the extended[1] Teichmüller space of genus g surfaces with n punctures to finite dimensional complex vector spaces

$$V : \mathcal{T}^{\mathrm{ext}}_{g,n} \longrightarrow \mathrm{VS}/\mathbb{C} \tag{4.7}$$

that qualifies as a *unitary topological modular functor* (UTMF). The data of a UTMF is equivalent to a *unitary modular tensor category* (UMTC) which one might also call "abstract anyonic system", as it generalizes the notion of a physical system with anyon excitations. Instead of giving the full definition [9], a table of the relations to physics will aid as a starting point.

[1]The punctures and boundaries are labeled and all diffeomorphisms respect the labels.

UMTC	Physical system
(simple) object	anyon
morphism	unitary operator
label	anyon type
tensor product with rules	"fusion" with fusion-rules
representations of Γ	anyon statistics
"tangles"	anyon trajectories

Most importantly, any anyonic system is specified by:

Particle type

Abstractly, the particle type of the physical system is given by a finite set of labels of objects, together with conjugate labels and the "trivial label"

$$a, b, c \qquad \bar{a}, \bar{b}, \bar{c} \qquad 1. \tag{4.8}$$

In the corresponding physical system the labels usually are determined by some kind of conserved charge. The notion of conjugate and trivial label is due to them being subject to the

Fusion rules

These are the abstract data of how the anyons in the physical system interact to form new anyons with new charge. In case of FQH fluids they describe how quasi-holes join together and in Chern-Simons how interactions result in bound states. They are denoted as

$$a \times b = \sum_c N_{ab}^c c \tag{4.9}$$

such that

$$a \times b = b \times a \quad \text{and} \quad a \times 1 = a. \tag{4.10}$$

The role of a neutral element with respect to fusion explains the term "trivial label". The fusion processes can be packaged into vector spaces denoted as V_{ab}^c that are called *topological Hilbert spaces*, because they carry global information of the anyonic system: The data of how the labels join via fusion is no specific property of any of the partners, only of the pair itself. They will play a crucial role in the encoding of information in the topological quantum computer. A complete, orthogonal basis [3] is given by

$$\{|a, b; c, \mu\rangle; \mu = 1, ..., N_{ab}^c\} \tag{4.11}$$

and a useful notation is

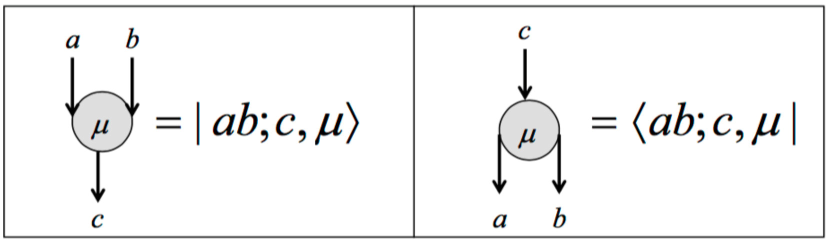


Figure 4.4: Diagram notation of the basis elements. Source: [3].

The role of conjugate labels in the physical picture is that of antiparticles: They are obtained by reversing the arrows in all pictures. By this rule, we have isomorphisms

$$V_{ab}^{c} \simeq V_{a\bar{c}}^{\bar{b}} \simeq V_{ab\bar{c}}^{1} \simeq \dots \tag{4.12}$$

In particular

$$V_{a0}^{a} \simeq V_{0}^{a\bar{a}}, \tag{4.13}$$

i.e. anyon pairs that are created from the vacuum have conjugate labels.
Lastly, the dimension of the topological Hilbert spaces are related to the notion of abelian and non-abelian in the abstract case. It is

$$\dim\left(\bigoplus_{c} V_{ab}^{c}\right) = \sum_{c} N_{ab}^{c} \tag{4.14}$$

and if this value is 1 for *any* a and b, meaning that there is only one fusion channel for two labels of any kind, the anyonic system is called *abelian*, otherwise *non-abelian*.

Braiding

Another part of the definition of the UMTC are the braiding rules. These are inspired by the braiding anyons thought of as punctures on the disc and should abstractly correspond to a morphism

$$R : V_{ab}^{c} \longrightarrow V_{ba}^{c}. \tag{4.15}$$

This can be visualized in the diagramatic language of the basis by

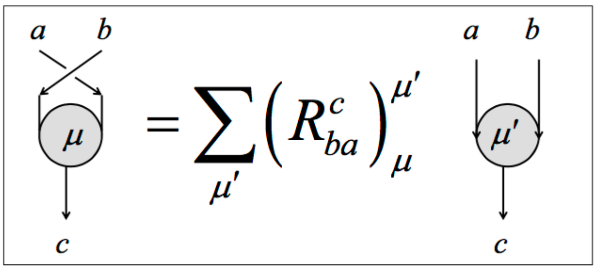


Figure 4.5: Definition of the R-matrix. Source: [3].

Associativity of Fusion

Fusion should be an associative operation with respect to the labels, i.e.

$$(a \times b) \times c = a \times (b \times c). \tag{4.16}$$

This can be visualized by

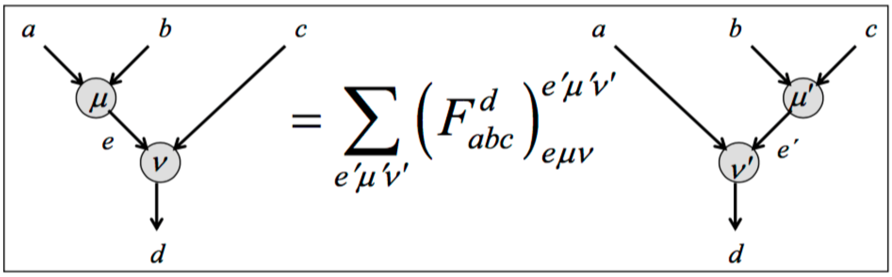


Figure 4.6: Definition of the F-matrix. Source: [3].

Standard Basis

This last point is not part of the definition, but a usual starting point for the treatment of the topological quantum computer. Due to their global nature, the topological Hilbert spaces do not allow for a natural decomposition into subspaces, but another useful *standard basis* is the decomposition

$$V^c_{a_1,\ldots,a_n} \simeq \bigoplus_{b_1,\ldots,b_n} V^{b_1}_{a_1 a_2} \otimes V^{b_2}_{b_1 a_3} \otimes \cdots V^c_{b_{n-2} a_n}. \tag{4.17}$$

It is obtained by pairwise fusing the ingoing states by starting from the left, and continuing with the intermediate labels $b_1, \ldots, b_{n-2}$. The dimension can be simplified by summing over all intermediate states

$$\dim\left(V^c_{a_1,\ldots,a_n}\right) = \sum_{b_1,\ldots,b_{n-2}} N^{b_1}_{a_1 a_2} \ldots N^c_{b_{n-2} a_n}. \tag{4.18}$$

It is not directly obvious how braiding works with respect to the new basis elements, but using the diagramatic expression for a basis element

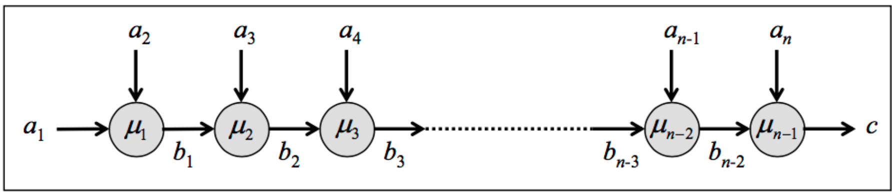


Figure 4.7: Diagram notation for the standard basis. Source: [3].

we can deduce that the correct morphism is given by

$$B := F^{-1}RF : V^d_{abc} \longrightarrow V^d_{acb}, \tag{4.19}$$

due to

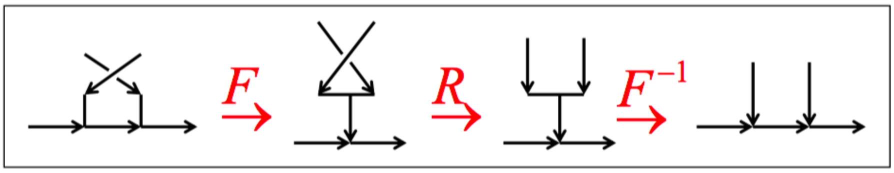


Figure 4.8: Definition of the B-matrix. Source: [3].

5 Anyons vs. Qubits

In order to compare the computational power of any kinds of systems it suffices to analyze their ability to simulate each other. The device we want to test in this way is a computer build out of a physical system with anyons that is formulated in the abstract language.

Definition 5.1. A *topological quantum computer* (TQC) is an abstract anyonic system subject to the following computation scheme:

1) **Initialization.**
   A finite set of anyon pairs is created from the vacuum.

2) **Processing.**
   The system evolves in time. The physical information is purely encoded in the braiding of anyons.

3) **Output.**
   Adjacent pairs are fused and it is checked whether there is a full annihilation, i.e. if the product has a trivial label. There is only one fusion channel for this process.

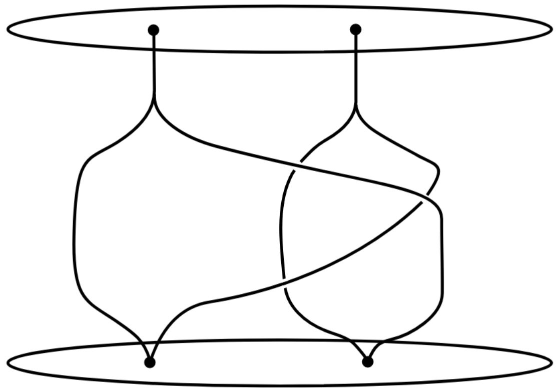


Figure 5.1: Sketch of a topological quantum computation.

We will demonstrate that any TQC can be simulated by an ordinary QC, which means that this new computation scheme is not more powerful than the ordinary one. This result does not depend on the specific model, it is true for any anyonic system that can be thought of. For the question of efficient simulation, see [10]. It suffices to explain how a QC simulates the essential steps: Encoding of the topological Hilbert spaces, braiding and fusion.

Encoding

The data of the TQC is encoded in $(n-2)$ variants of the topological Hilbert spaces $V^1_{a_1,\ldots,a_n}$. As commented above, these cannot be naturally described by n-qubits, which admit a decomposition into single qubits, due to the non-local information content. This problem can be solved by allowing for a sufficiently large Hilbert space of states that contains all of the topological ones, namely

$$(\mathcal{H}_d)^{\otimes(n-2)} \tag{5.1}$$

with

$$\mathcal{H}_d := \bigoplus_{a,b,c} V^1_{abc}. \tag{5.2}$$

Here, d corresponds to the summed dimensions of the single fusion spaces

$$d = \sum_{a,b,c} N^1_{a,b,c}. \tag{5.3}$$

These Hilbert spaces are no qubits, but they are good enough, as this quantum system can be efficiently simulated by an ordinary quantum computer as already discussed. The input information of the TQC can be encoded in an ordinary QC by the Hilbert spaces of $(n-2)$ "qudits" for some d. The natural decomposition is given by (5.1) and a basis is

$$\left\{|a,b,c;\mu\rangle; \mu = 1,\ldots,N^1_{abc}\right\}. \tag{5.4}$$

Braiding

How does B act on $(\mathcal{H}_d)^{\otimes(n-2)}$? The corresponding operator acts on a pair $|a,b,\bar{c}\rangle, |d,e,\bar{f}\rangle$ of anyons in the standard basis and due to the relations of the braid group it suffices to look at neighbours, i.e. $c = d$. Then B can be represented by a $d^2 \times d^2$ unitary matrix acting on neighbouring qudits:

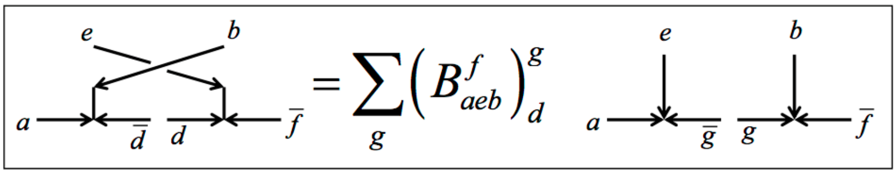


Figure 5.2: Action of the B-matrix on neighbouring qudits. Source: [3].

Output

Given a pair of neighbouring qudits $|a,b,\bar{d}\rangle, |d,e,\bar{f}\rangle$, what is the amplitude for the product $b \times e$ to have trivial total charge? This question can be adressed by using the F-matrix

$$|a,b,\bar{d}\rangle \otimes |d,e,\bar{f}\rangle \overset{F}{\mapsto} \sum_g |a,g,\bar{f}\rangle \otimes |b,\bar{g},e\rangle \left(F^f_{abe}\right)^g_d \tag{5.5}$$

$$= |a,1,\bar{f}\rangle \otimes |b,1,e\rangle \left(F^f_{abc}\right)^1_d + \sum_{g\neq 1} |a,g,\bar{f}\rangle \otimes |b,\bar{g},e\rangle \left(F^f_{abc}\right)^g_d . \tag{5.6}$$

This transformation can be performed by a QC, as F is a unitary two-qudit gate. Reading out the output of the computation can be performed by a projection via

$$|b,\bar{g},e\rangle \otimes \langle b,\bar{g},e| \tag{5.7}$$

and recording whether or not $g = 1$.

As these three steps completely determine the topological quantum computation, we see that a universal quantum circuit can simulate a TQC (efficiently).

The Fibonacci-Model

But is a TQC at least *as good* as a QC? Adressing this question is much harder and requires remarkable theorems from theoretical computer science [3], the answer being that it depends on the specific system: Some anyonic systems are indeed able to efficiently simulate a universal quantum computer, while others are not.

Probably the simplest model which is capable of simulating a QC is the *Fibonacci-Model*[1]: It has anyons of two labels 1 and τ, with 1 being trivial and τ subject to $\bar{\tau} = \tau$. There is only a single fusion rule

$$\tau \times \tau = 1 + \tau, \tag{5.8}$$

meaning that two non-trivial labels either annihilate or fuse together to form another copy of themselves. In particular, the Fibonacci-Model is a non-abelian anyonic system, as there are two fusion channels for τ. Of most interest are the fusion spaces $V^1_{\tau\tau\ldots\tau}$ with intermediate states $b_1, ..., b_{n-2}$. If the fusion product of all the τ's is 1, then the fusion product of the first $n-1$ τ's has to be τ. In addition there must not be two successive 1's. Otherwise there are no obstructions to the input data. The dimensions of the topological Hilbert spaces will then obey a recursion relation:
If the first to anyons fuse to the trivial label then the remaining $n-2$ anyons can fuse in $N^1_{\tau^{n-2}}$ distinguishable ways, if they fuse to a nontrivial label, the remaining ones can fuse in $N^1_{\tau^{n-1}}$ distinguishable ways. Therefore, the recursion relation reads

$$N^1_{\tau^n} = N^1_{\tau^{n-1}} + N^1_{\tau^{n-2}}. \tag{5.9}$$

As $N^1_\tau = 0$ and $N^1_{\tau\tau} = 1$ this is exactly the Fibonacci sequence, explaining the name of this model.
In order to test the computational power of Fibonacci anyons by trying to simulate a universal QC, we need to find a braid b, such that for $i \in \{0, 1\}$, a representation ρ of the braid group and a unitary operator U, the diagram

$$\begin{array}{ccc} \mathbb{C}[\mathbb{Z}_2^n] & \longrightarrow & V^i_{\tau^n} \\ \big\downarrow U & & \big\downarrow \rho(b) \\ \mathbb{C}[\mathbb{Z}_2^n] & \longrightarrow & V^i_{\tau^n} \end{array} \tag{5.10}$$

commutes up to arbitrary precision. This is one of the results of [11].

Theorem 5.2 (Freedman, Larsen, Kitaev, Wang)**.** For any quantum circuit $U : \mathbb{C}[\mathbb{Z}_2^n] \to \mathbb{C}[\mathbb{Z}_2^n]$ in $\mathrm{SU}(2^n)$ and error threshold $\delta > 0$, there is a braid $b \in \mathcal{B}_{2n+2}$, such that

$$|\rho(b) - U| < \delta \tag{5.11}$$

and such that b can be found by a classical Turing machine in polynomial running time.

After having this result the immediate question is: How to actually do computations in TQC? Unfortunately, even the simplest logical operations are considerably hard to implement in braids of anyonic trajectories. In particular, they also depend on the specific model. In case of Fibonacci anyons, see [12].

[1] A conformal field theorist would call it *Yang-Lee-Model.*

Error protection in TQC

The major advertisement point of TQC comes from a flaw in ordinary quantum computation: There is a large source of errors throughout the quantum computation process due to local interactions between qubits and the computer device itself [13]. There are strategies which may allow for a solution in the future, using error protection codes, but to date they require an utopian small error to begin with.

This problem is almost immaterial in TQC. While a proper explanation requires the explicit description of the anyonic system [14] or more elaborate abstract machinery [15], the basic idea is already evident from this discussion. The fundamental advantage of TQC is the fact that only the global information of braiding anyon trajectories enters into the computation process. When realized in the physical world, any noise could lead to creation of anyon pairs, but they only enter nontrivially in the computation when at least one of them winds around the trajectory of one of the qubit encoding anyons. If the qubit encoding anyons are properly separated this process is exponentially suppressed with system size.

Bibliography

[1] John Preskill. Quantum Information Theory. 2017.

[2] John Preskill. Quantum Computation. 2017.

[3] John Preskill. Topological Quantum Computation. 2017.

[4] Zhenghan Wang. Topological Quantum Computation. 2010.

[5] Richard P. Feynman. Simulating Physics with Computers. 1981.

[6] Seth Lloyd. Universal Quantum Simulators. 1996.

[7] M. Suzuki. Improved Trotter-like Formula. 1993.

[8] Katherine L. Brown, William J. Munro, and Vivien M. Kendon. Using quantum computers for quantum simulation. 2010.

[9] Bojko Bakalov and Alexander Kirillov. Lectures on Tensor Categories and Modular Functors. 2000.

[10] Michael H. Freedman, Alexei Kitaev, and Zhenghan Wang. Simulation of topological field theories by quantum computers. 2000.

[11] M. Freedman, M. Larsen, and Z. Wang. A modular functor which is universal for quantum computation. 2002.

[12] N. E. Bonesteel, L. Hormozi, G. Zikos, and S. H. Simon. Braid Topologies for Quantum Computation. 2005.

[13] John Preskill. Quantum Error Correction. 2017.

[14] A. Kitaev and C. Laumann. Topological phases and quantum computation. 2009.

[15] M. H. Freedman, A. Kitaev, M. J. Larsen, and Z. Wang. Topological Quantum Computation. 2002.