Hopf Algebras in Quantum Computation

Giovanni de Felice

New College

University of Oxford

A thesis submitted for the degree of

Master of Science

Trinity 2017

Abstract

In this thesis, we use string diagrams to study the theory of Hopf algebras in the context of Categorical Quantum Mechanics. First, we treat the theory of representations of a Hopf algebra diagrammatically. The category of representations of a quasitriangular Hopf algebra $Rep(H)$ is a braided tensor category and can be understood as a process theory of particles in Topological Quantum theory. We provide diagrammatic proofs of equivalences relating the Drinfeld center construction on $Rep(H)$ to the category of Quantum double modules $Rep(DH)$. We then use similar tools to generalize Kitaev's lattice models for Topological Quantum computation and give a categorical perspective on Permutational Quantum computation. Finally, we discuss functorial semantics in the context of quantum computation, by constructing a braided language for stabilizer quantum gates.

Contents

Introduction

Algebraic structures have historically been described in set-theoretic terms. One usually considers a set equipped with functions satisfying certain axioms. For example, a monoid is a set equipped with a multiplication and a unit element satisfying associativity and the unit law. Category theory provides a unifying framework for describing algebraic theories. Many of the driving notions of category theory arise through categorification of set-theoretic notions [4]. Categorification is the process of replacing sets by categories, functions by functors and relaxing equations to natural isomorphisms between the functors. For example, the notion of a monoidal category is the categorification of that of a monoid. Since their preliminary study [25, 20] monoidal categories have been used as building blocks for more complex algebraic theories and have found many applications to quantum physics [1, 7, 39, 34]. They have a very intuitive two-dimensional diagrammatic language where algebraic equations are topological moves [35]. Using this language it is possible to represent processes on physical systems [10], which can often be characterized by their symmetries. These latter are traditionally described by groups, as for instance for the symmetries of crystals [3] or spin-$\frac{1}{2}$ particles [40]. Quasitriangular Hopf Algebras provide a generalization of group theory which allows describing the symmetries of many-body quantum systems as they treat local and global (or topological) symmetries on the same level [37].

The aim of this thesis is to provide a diagrammatic treatment of Hopf algebras and their representations, outlining their relationship to quantum computation.

In the first chapter we introduce monoidal categories to obtain a diagrammatic characterisation of Hopf Algebras. This is done using functorial semantics, in the spirit of Lawvere theories [24]. We then recall some standard results and move on to representation theory. Given a quasitriangular Hopf algebra H, the category of its representations $Rep(H)$ is braided monoidal and every object has a dual. If we additionally require H to be quasitriangular, we obtain braids, making the language of $Rep(H)$ three-dimensional. In these categories, knots and links are scalars, and we naturally obtain topological invariants.

The second chapter is dedicated to the study of special types of particles called anyons, which appear in three-dimensional space-time. Their symmetries and exchange statistics are captured by quasitriangular Hopf algebras, so that categories of representations can be interpreted as process theories of anyons. We also see how the Drinfeld center construction, which takes place at the categorical level, corresponds to a certain quantization of the symmetries: the quantum double construction on a Hopf algebra. Our contribution here is a diagrammatic proof, characterising the Drinfeld center $Z(Rep(H))$ through equivalences. We show $Z(Rep(H))$ is equivalent

to the category of representations of the quantum double of H when H is finite dimensional and prove variations of this result in the cases where H is only a bialgebra and when it is (possibly) infinite-dimensional.

In chapter 3 we study two different models of quantum computation induced by Hopf algebras. Kitaev's double model for topological quantum computation (TQC) [21], which is based on a group, is formulated diagrammatically in terms of Hopf algebras, providing a physical illustration of the results of section 2.3. Jordan's model for permutational quantum computing (PQC) [18] is described from a categorical perspective in view of relating it to TQC. Finally, in section 3.3, we come back to our discussion on functorial semantics using the notion of functorial boxes [28]. These are used to project the pictures of knots from braided fusion categories into Hilbert spaces and obtain a braided syntax for quantum gates.

Chapter 1

Diagrams and Hopf Algebras

1.1 Monoidal categories

In this section, we set in place the basic definitions and the diagrammatic framework which we will use throughout the thesis. This section can be freely skipped given basic knowledge of monoidal categories and string diagrams. The standard reference about basic category theory is [25]. Many of the definitions are taken from [2]. For an introduction to diagrammatic reasoning in monoidal categories consider the first two chapters of [10]. A more technical and up to date survey on monoidal categories can be found in [12]. Many of the results recalled in this section and their relationship to quantum mechanics can be found in [39].

Definition 1.1 (Category)**.** *A category $\mathcal{C}$ consists of the data:*

- *a collection of objects $obj(\mathcal{C})$,*
- *a collection of morphisms (or arrows) $arr(\mathcal{C})$,*
- *domain and codomain assignments: $dom, cod : arr(\mathcal{C}) \to obj(\mathcal{C})$. For any two objects $a, b \in obj(\mathcal{C})$ we define the hom-set*

$$\mathcal{C}(a, b) := \{f \in arr(\mathcal{C}) : a = dom(f), b = cod(f)\}$$

- *for any triple of objects a,b and c, a composition map*

$$\mathcal{C}(a, b) \times \mathcal{C}(b, c) \to \mathcal{C}(a, c)$$

we denote the composition by $g \circ f$, diagrammatically:

$$a \xrightarrow[g \circ f]{f \quad b \quad g} c$$

- *for any object a an identity morphism $id_a : a \to a$,*

satisfying the following axioms:

$$h \circ (g \circ f) = (h \circ g) \circ f \quad ; \quad f \circ id_a = f = id_b \circ f$$

Example 1.1. *Examples of categories are: Sets of sets and functions, FSets of finite sets and functions, Rel of sets and relations, $Vect_k$ of vector spaces over a field k and linear maps and $FVect_k$ of finite dimensional vector spaces and linear maps.*

Category theory expresses equivalences and relationships between structures by means of the following tools.

Definition 1.2 (Functor)**.** *A functor $F : \mathcal{C} \to \mathcal{D}$ is a mapping that*

- *associates an object $F(a)$ of $\mathcal{D}$ to each object a of $\mathcal{C}$,*
- *associates to each morphism $f : a \to b$ in $\mathcal{C}$ a morphism $F(f) : F(a) \to F(b)$ in $\mathcal{D}$ such that $F(id_a) = id_{F(a)}$ and $F(g \circ f) = F(g) \circ F(f)$ for all morphisms $f : a \to b$ and $g : b \to c$.*

For instance, there is a functor $Q : FSets \to FVect_k$ taking a set to the free vector space generated by that set.

Remark Throughout the thesis we will sometimes describe our functors with some adjectives that we have only defined for categories (for example we will sometimes write 'monoidal functor' or 'symmetric functor'). In all cases this means functors which preserve the structure described by the adjective. For the rigorous definitions please consult *ncatlab.org/nlab*.

Given two functors with matching source and target we can have natural transformations between them.

Definition 1.3 (Natural Transformation)**.** *Given categories $\mathcal{C}$ and $\mathcal{D}$ and functors $F, G : \mathcal{C} \to \mathcal{D}$ a natural transformation $\alpha : F \Rightarrow G$ is an assignment to every object a in $\mathcal{C}$ of a morphism $\alpha_a : F(a) \to G(a)$ in $\mathcal{D}$ such that for each morphism $f : a \to b$, the following commutes:*

$$\begin{array}{ccc} G(a) & \xrightarrow{G(f)} & G(b) \\ {\scriptstyle \alpha_a}\uparrow & & \uparrow{\scriptstyle \alpha_b} \\ F(a) & \xrightarrow[F(f)]{} & F(b) \end{array}$$

A natural isomorphism is a natural transformation such that all components are isomorphisms.

Recall that a monoid is a triple $(X, \times, 1)$ where X is a set, $1 \in X$ and $\times$ is an associative and unital multiplication on X. The notion of a monoidal category is the categorification of a monoid. Elements of the set are replaced by objects in a category $\mathcal{C}$, multiplication by a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ and the equalities in the unit and association axioms are replaced by natural isomorphisms. In order for this new structure to be well-behaved we will also need to impose compatibility conditions. We obtain the following definition:

Definition 1.4 (Monoidal category)**.** *A monoidal category is a category $\mathcal{C}$ equipped with a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ called tensor product, an object 1 called unit object, a natural isomorphism*

$$\alpha : (- \otimes -) \otimes - \xRightarrow{\sim} - \otimes (- \otimes -)$$

called associator, a natural isomorphism

$$\lambda : 1 \otimes (-) \xRightarrow{\sim} (-)$$

called left unitor and a natural isomorphism

$$\rho : (-) \otimes 1 \xRightarrow{\sim} (-)$$

called right unitor. Subject to the following coherence conditions holding for all objects a, b, c, d in $\mathcal{C}$:

1. *Pentagon axiom: the following diagram commutes*

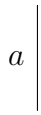
$$\begin{array}{ccc}
 & (a \otimes b) \otimes (c \otimes d) & \\
{\scriptstyle \alpha_{a\otimes b,c,d}}\nearrow & & \searrow{\scriptstyle \alpha_{a,b,c\otimes d}} \\
((a \otimes b) \otimes c) \otimes d & & a \otimes (b \otimes (c \otimes d)) \\
\Big\downarrow{\scriptstyle \alpha_{a,b,c} \otimes id_d} & & \Big\uparrow{\scriptstyle id_a \otimes \alpha_{b,c,d}} \\
(a \otimes (b \otimes c)) \otimes d & \xrightarrow[\alpha_{a,b\otimes c,d}]{} & a \otimes ((b \otimes c) \otimes d)
\end{array}$$

2. *Triangle identity: the following diagram commutes*

$$\begin{array}{ccc}
(a \otimes 1) \otimes b & \xrightarrow{\alpha_{a,1,b}} & a \otimes (1 \otimes b) \\
 & {\scriptstyle \rho_a \otimes id_b}\searrow \quad \swarrow{\scriptstyle id_a \otimes \lambda_b} & \\
 & a \otimes b &
\end{array}$$

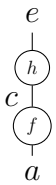
Example 1.2. *The category Sets of sets and functions is monoidal with the cartesian product $\times$ and the singleton set as unit object.*
Sets is also monoidal if equipped with the coproduct $\coprod$ as monoidal product and the empty set as unit object. Note that this shows that being monoidal is a structure and not a property of categories. Many choices of monoidal structure are usually possible.
The category $Vect_k$ of finite dimensional vector spaces over a field k is monoidal with the usual tensor product $\otimes$ and the one dimensional vector space k as unit object.
The category Rel of sets and relations is monoidal with the cartesian product $\times$ and the singleton as unit object.
The category Hilb of Hilbert spaces and linear maps is monoidal when equipped with the usual tensor product $\otimes$.

The pentagon and triangle axioms make sure that any well formed diagram in a monoidal category, made up of associators and unitors, commutes. This is known as the coherence theorem for monoidal categories and can be found in [25]. When the associators and unitors are trivial morphisms (i.e identity morphisms) we say the category is strict monoidal. It is known that every monoidal category is equivalent to a strict one [25], but it is sometimes useful to take associators into account as we will see in our discussion on permutational quantum computation. As shown in the survey [35], strict monoidal categories have a two-dimensional diagrammatic which is rigorous. Objects are represented by their identity morphisms which we draw as labelled wires:

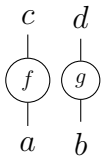


A morphism $f : a \to c$ is drawn as a box with input and output wires going from bottom to top:

The vertical composition $h \circ f$ where $h : c \to e$ is denoted as:



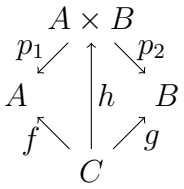
We write the tensor of two morphisms $f \otimes g : a \otimes b \to c \otimes d$ simply putting them side by side:



Note that because the category is strict $id_1 \otimes f = f = f \otimes id_1$ for any f, so we could draw as many copies as we like of id_1 on any diagram to obtain an equivalent one. So really the identity on 1 is just the empty diagram which we can stick next to any diagram we like. Isotopic diagrams correspond to the same processes [35, Theorem 3.3] so that topological moves are allowed algebraic manipulations.

Definition 1.5 (States and costates)**.** *Given a system A, a state of A is a morphism $y : 1 \to a$. A costate (or effect) of a is a morphism $x : a \to 1$. In the diagrammatic language we draw states and costates respectively:*

Remark The cartesian product $\times$ in *Sets* satisfies the universal properties of a categorical product, in the sense that we have projections $p_1 : A \times B \to A$ and $p_2 : A \times B \to B$ such that if f and g are maps from some set C there is a unique function h making the following diagram commute:



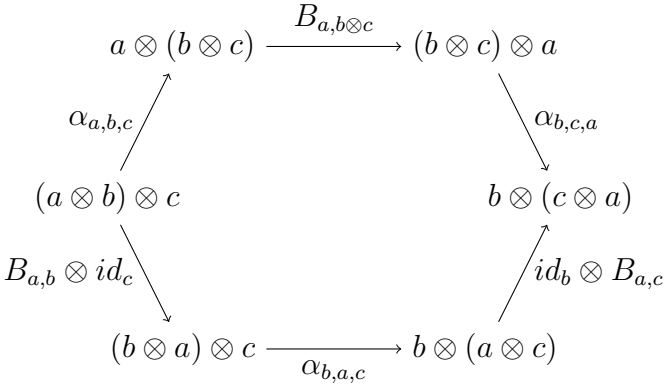
This implies that all states in $(Sets, \times)$ are separable, in the sense that any state c of $A \times B$ is a product state $a \times b$. The product $\times$ in *Sets* is cartesian whereas $\otimes$ in $Vect$ is not as $\otimes$ is not the categorical product. The categorical product in $Vect$ is the biproduct and it is defined in the appendix.

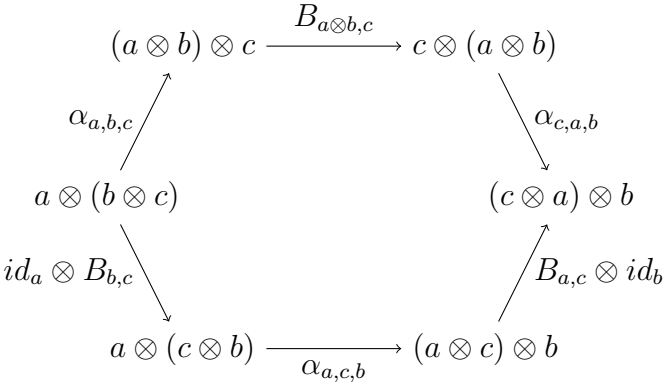
Example 1.3. *In $Vect_k$ states are vectors and costates are functionals. Note that the diagrammatic notation provides a two-dimensional generalisation of Dirac's notation. Note that the tensor of vector spaces $\otimes$ is not a categorical product, and in fact we can have non-separable (entangled) states.*

Definition 1.6 (Scalars)**.** *Scalars in a monoidal category are morphisms $1 \to 1$.*

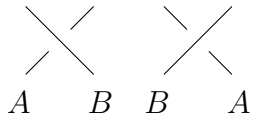
The category *Sets* has only one scalar. *Rel* has two scalars forming the booleans $\mathbb{B}_2$ under composition. $Vect_k$ has scalars from k. Given a vector and a functional we obtain a scalar by composing them.

Definition 1.7 (BMC)**.** *A braided monoidal category is a monoidal category $\mathcal{C}$ equipped with a natural isomorphism $B_{a,b} : a \otimes b \to b \otimes a$ called braiding, subject to the following compatibility conditions (called hexagon equations):*

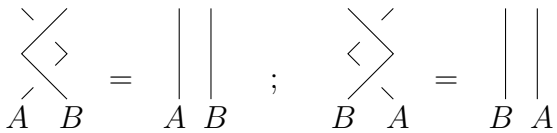




In the diagrammatic language this means we have braidings:



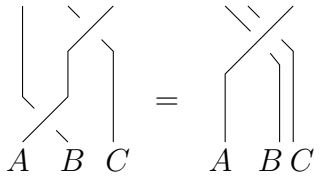
for any A and B, satifying:



(1.1)

The compatibility conditions are obvious statements in the diagrammatic calculus,

for instance the first hexagon equation just says:



(1.2)

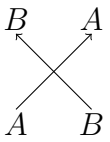
Both *Sets* and *Hilb* are examples of symmetric monoidal categories in the following sense.

Definition 1.8 (SMC)**.** *A braided monoidal category is symmetric if the braiding* $B_{a,b}$ *satisfies*

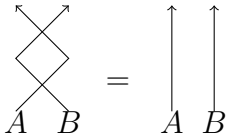
$$B_{a,b} \circ B_{b,a} = id_{a \otimes b}$$

For all objects a, b*.*

In a SMC the braiding is called symmetry morphism and is denoted



It satisfies:



We will now describe some new classes of examples of monoidal categories. These are of a different nature to the categories we have seen so far.

Definition 1.9 (PRO)**.** *A PRO (products category) is a strict monoidal category where every object is of the form* $x^{\otimes n}$ *for a single object* x *and* $n \geq 0$*.*

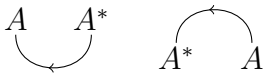
Definition 1.10 (PROPs)**.** *A PROP (products and permutations category) is a symmetric strict monoidal category where every object is of the form* $x^{\otimes n}$ *for a single object* x *and* $n \geq 0$*.*

This means that we are only allowed one type of wire when drawing diagrams about *PROPs* but we can use as many copies as we like and we can make swaps with them. Categories satisfying these properties are useful syntactic tools as we will see. One way to think of a *PROP* A is as an abstract algebraic structure carrying some axioms, we can then instantiate those axioms in some other symmetric monoidal category $\mathcal{C}$ by considering symmetric monoidal functors $F : A \to \mathcal{C}$. We call such functors algebras or models of A in $\mathcal{C}$. If A is defined in terms of generators and

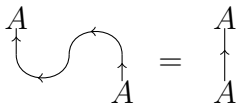
relations (as is most often done), the choice of such functor corresponds to the choice of one object from $\mathcal{C}$ and morphisms on that object respecting the defining relations of A. On its own A has no clear interpretation, it just defines a syntax, but if $\mathcal{C}$ is a semantic category (i.e one with a clear interpretation) then F is a 'filling' of the syntax with meaning. This reasoning was first proposed in Lawvere's Phd thesis in 1963 [24].

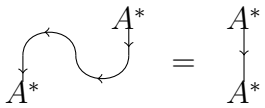
Remark The semantic categories we will use the most are $Sets$, $Vect_{\mathbb{C}}$ and $FVect_{\mathbb{C}}$. In fact we will only consider vector spaces over $\mathbb{C}$ so from now on let us denote $Vect := Vect_{\mathbb{C}}$ and $FVect := FVect_{\mathbb{C}}$. One important difference between $Sets$ and $FVect$ is that $FVect$ exhibits duality.

Definition 1.11 (Rigidity)**.** *Let $\mathcal{C}$ be a monoidal category and $A \in obj(\mathcal{C})$. A left-dual of A is an object A^* with morphisms*



Satisfying the snake equations:

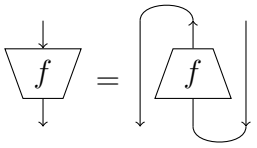




If every object has a left-dual, we say that $\mathcal{C}$ is left-rigid. Similarly we can define right-duals and right-rigid categories by interchanging the roles of A and A^* in the definition.

Given a (left/right) rigid structure we can define (left/right)transpose as follows.

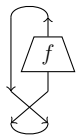
Definition 1.12 (Transpose)**.** *Given a (left/right) rigid category $\mathcal{C}$ and any process $f : A \to B$ the (left/right) transpose f^* (or left transpose f^l, right transpose f^r if it is not clear from context) is:*



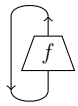
(1.3)

Given the above definition we can define the (left/right) transpose endofunctor on a (left/right) rigid category $\mathcal{C}$ as the funcotr $(-)^* : \mathcal{C} \to \mathcal{C}$ taking objects to their (left/right) duals and morphisms to their (left/right) transpose.

Definition 1.13 (Trace)**.** *In a symmetric monoidal category $\mathcal{C}$, if A has a left dual A^*, the trace of some morphism $f : A \to A$ is defined as the following scalar:*



A pivotal structure on a rigid monoidal category $\mathcal{C}$ is a natural isomorphism $id_{\mathcal{C}} \Rightarrow (-)^{**}$. It allows to define traces without using the symmetry. Most categories we will consider have both sided duals (in the sense that left and right duals coincide), and therefore a trivial (identity) pivotal structure. Given a pivotal structure we can define left pivotal traces as:

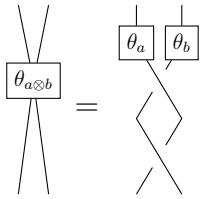


Where we have hidden the pivotal natural isomorphism. Similarly we can define right pivotal traces on endomorphisms in the obvious way.

Definition 1.14. *A rigid monoidal category with a pivotal strucutre is spherical if left and right traces coincide. In a spherical category, if a is an object, the trace $tr : End(a) \to End(1)$ is well defined and $tr(id_a)$ is called the categorical (or quantum) dimension of a.*

For a braided monoidal category, giving a spherical structure is equivalent to giving a ribbon structure [31] where:

Definition 1.15. *A ribbon (or twist) structure on a braided monoidal category with left duality $\star$ is a natural isomorphism $\theta : id_{\mathcal{C}} \Rightarrow id_{\mathcal{C}}$ satisfying:*



(1.4)

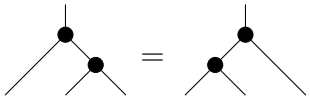
and compatible with the rigid structure $(\theta_a)^\star = \theta_{a^\star}$

1.2 Hopf algebras

Now that we have set in place a diagrammatic machinery based on monoidal categories, let us make use of it. In this section we will meet some mathematical structures which have been used by mathematicians to describe symmetry. The

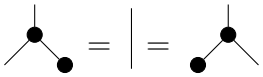
notion of Hopf algebras is a powerful generalization of that of a group. Since their discovery in the 1940s, Hopf algebras have been used in various fields of pure mathematics (such as number theory, algebraic geometry, and representation theory) and have found applications in Quantum mechanics. Most of the results of this section can be found in [26].

Definition 1.16 (Monoid)**.** Δ *is a PRO generated by morphisms (*  *,* *) satisfying associativity:*



(1.5)

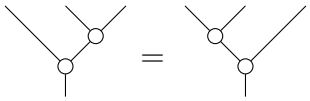
and the unit law:



(1.6)

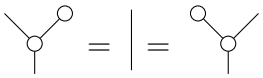
Models of Δ in monoidal categories are called monoids and they are very well known, examples include the natural numbers under addition, lists of some alphabet under concatenation and any group. Taking the opposite category Δ^{op} corresponds to flipping all the diagrams.

Definition 1.17 (Comonoid)**.** Δ^{op} *is a PRO generated (*  *,* *) satisfying coassociativity:*



(1.7)

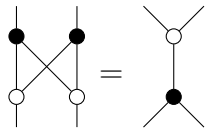
and the counit law:



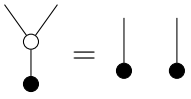
(1.8)

Models of these are comonoids, the most common example is the copy map on any set with 'delete' as counit. Monoids and comonoids are simple structures that we can stick together to form more complicated ones. Bialgebras arise from one type of interaction of a monoid and comonoid.

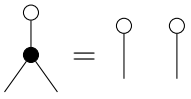
Definition 1.18 (Bialg)**.** *Bialg is a PROP generated by (* *,* *,* *,* *), where* *and* *form a monoid,* *and* *a comonoid and the morphisms additionally satisfy the following laws:*



(1.9)



(1.10)



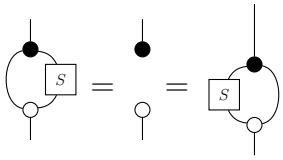
(1.11)

(1.12)

where the empty diagram is the identity on the tensor unit.

Models of $Bialg$ in $Vect$ are bialgebras. We leave examples for later as we are now ready to introduce one of the main topics of this thesis.

Definition 1.19 (Hopf)**.** *$Hopf$ is a PROP generated by (*  *), where (* *) is a bialgebra and the antipode S satisfies the Hopf law:*



(1.13)

We will argue that $Hopf$ is a good syntax to talk about symmetry. Let us start by instantiating $G : Hopf \to Sets$. This corresponds to choosing a set G, with an associative binary operation $G \times G \to G$ (or multiplication) with a unit. Using the counit rule it is easy to see that the comultiplication in $Sets$ must be the copy map $g \mapsto (g, g)$ so that the antipode is the morphism $g \mapsto g^{-1}$ and G forms a group. Since the 19^{th} century groups have been used by mathematicians and physicists to describe symmetry.

Example 1.4 (Finite groups)**.** *We will only make use of the following classes of finite groups:*

- *$\mathbb{Z}_n$ the cyclic group with n elements.*
- *S_n the symmetric group, can be seen as the group of permutations of a set with n elements, has order $n!$. S_3 is the smallest non-abelian group up to isomorphism.*

Example 1.5 (Groups of matrices)**.** *Here we will fix some notation on the infinite groups of matrices we will meet. All matrices we will consider are over the complex numbers. $GL(n)$ is the group of invertible n by n complex matrices. $U(n)$ is the group of unitary $n \times n$ matrices (i.e such that $U^\dagger U = UU^\dagger = I$). The special unitary group $SU(n)$ is the subgroup of $U(n)$ consisting of matrices with determinant 1. The representation theory of $SU(n)$ is widely used in particle physics, for instance representations of $SU(2)$ model the behaviour of spin-$\frac{1}{2}$ particles.*

If we take a model of $H : Hopf \to Vect$ we obtain what is known as a Hopf Algebra.The most common example of Hopf algebras are group algebras.

Example 1.6 (Group algebras)**.** *If G is a group with unit e, the group algebra $\mathbb{C}G$ (of dimension $|G|$) is a hopf algebra with multiplication linearly generated by $|g\rangle \otimes |h\rangle \to |gh\rangle$, unit $|e\rangle$, comultplication generated by $|g\rangle \to |g\rangle \otimes |g\rangle$ and counit $\sum_g \langle g|$.*

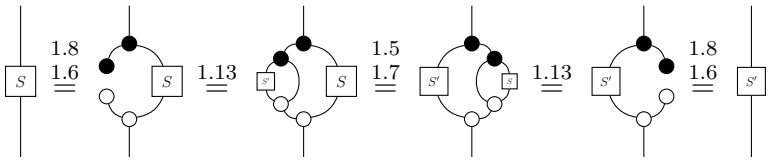
The previous example gives the usual definition of a group algebra which, for finite sets and finite dimensional vector spaces is just the composition $Q \circ G$ (as shown in the diagram) where $Q : FSets \to FVect$ is the free vector space functor. It is easy to see that Q preserves the monoidal structure as well as the symmetry morphisms (we say Q is a symmetric monoidal functor) so that the composition is also symmetric monoidal and $Q \circ G$ is a model of $Hopf$.

$$\begin{array}{ccc} & Hopf & \\ {\scriptstyle G}\swarrow & & \searrow{\scriptstyle \mathbb{C}G} \\ FSets & \xrightarrow[Q]{} & FVect \end{array}$$

In this case the comultiplication in $FVect$ is the linearisation of the copy map (the copy map on some basis extended linearly to the whole vector space) which is co-commutative. For a general $H : Hopf \to Vect$ this doesn't have to be the case. Hopf algebras provide a broader framework to talk about symmetry, as we can have non co-commutative Hopf algebras. We can see it as a quantization of the notion of symmetry, it will allow us to describe symmetries of quantum systems. The following two propositions are simple but important results about the antipode of a hopf algebra.

Proposition 1.1. *The antipode of a Hopf algebra is unique, i.e being a Hopf algebra is a property of bialgebras.*

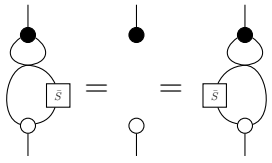
Proof. Suppose S and S' are two antipodes for some Hopf algebra, then:



□

Some bialgebras have a skew antipode instead of an antipode.

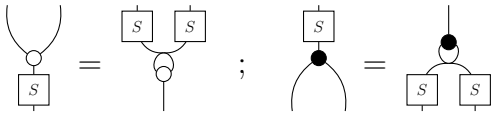
Definition 1.20 (Skew antipode)**.** *Given a bialgebra (, , ,). $\bar{S}$ is a skew antipode if it satisfies:*



(1.14)

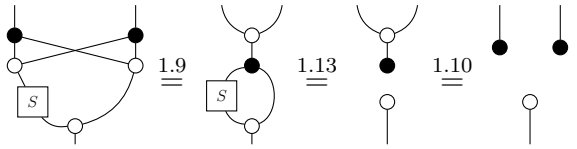
It is possible to replicate the proof above for skew antipodes to see that it is a property of bialgebras too. Most of the Hopf algebras we will meet also have a skew antipode. In particular, it is a well-known fact that Hopf algebras with invertible antipode S always have a skew antipode defined by $\bar{S} = S - S^{-1}$. Note that this argument applies to Hopf algebras, as they are models of $Hopf$ in $Vect$ where addition in the hom-sets is well-defined ($Vect$ is an abelian category, see the appendix) but it doesn't hold in general for models of the $Hopf\ PROP$.

Proposition 1.2. *The antipode is an anti-(co)algebra homomorphism.*

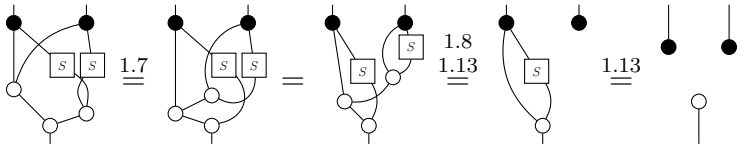


(1.15)

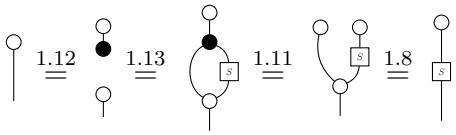
Proof. We recall the proof given in Figure 4.6 of [27].
First note that:



So that  is a left convolution inverse to  .
Also:

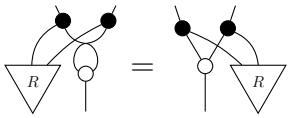


So that  is a right convolution inverse to  . And it is easy to see using associativity and co-associativity that right and left convolution inverses must coincide.
Also note that:

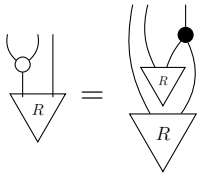


We deduce that the antipode is an anti-coalgebra homomorphism. For a proof that the antipode is an anti-algebra morphism simply flip all the diagrams and interchange white with black. □

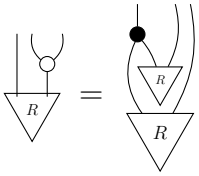
Definition 1.21 (Quasitriangularity)**.** *A Hopf algebra H is quasitriangular if there is an invertible element $R \in H \otimes H$ satisfying the following equations:*



(1.16)



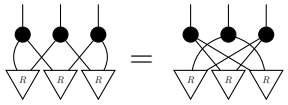
(1.17)



(1.18)

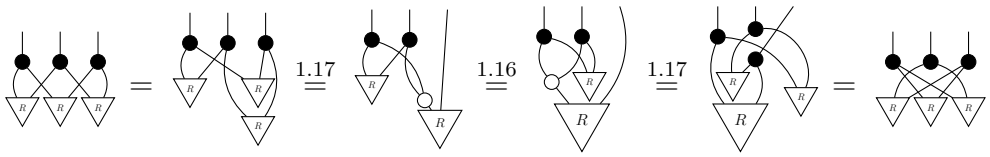
R is called the 'universal R-matrix', and it can be thought as controlling the non-cocommutativity of the Hopf algebra. Quasitriangular Hopf algebras are sometimes called Quantum groups. We will see that they exhibit topological behaviour, as the following proposition hints to.

Proposition 1.3. *The universal R-matrix satisfies the Quantum Yang-Baxter equation:*



(1.19)

Proof. Making use of isotopy invariance we get:



□

Example 1.7. *The most trivial example of quasitriangular Hopf algebras are the cocommutative ones. It is easy to check that if H is cocommutative, it is quasitriangular with* *as R-matrix.*

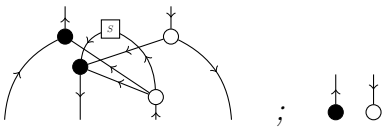
We will only be considering finite dimensional Hopf Algebras, as for finite dimensional vector spaces, these always have duals.

Definition 1.22 (Dual Hopf Algebra)**.** *For a finite dimensional Hopf Algebra H the dual Hopf algebra is the vector space $H^\star$ of linear functionals on H with Hopf Algebra structure given by transposing all of the structure.*

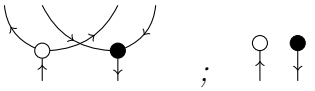
Given any finite dimensional Hopf algebra H with invertible antipode there is a standard way of constructing a Quasitriangular Hopf Algebra first introduced by Drinfeld [11]. It will be implicit from now on that all Hopf algebras (and vector spaces) are finite-dimensional unless stated otherwise.

Definition 1.23 (Quantum double of a Hopf algebra)**.** *The quantum double of a finite dimensional Hopf algebra* $(H, \mu, 1, \Delta, \epsilon, S)$ *with invertible antipode is the vector space* $H \otimes H^\star$*, with the following structure:*

- *multiplication and unit:*

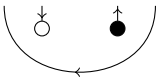


- *comultiplication and counit:*



- *antipode:*

It is shown in [26] that this is indeed a Hopf algebra and that it is quasitriangular with universal R-matrix:

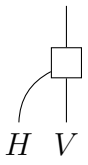


Physically we will see that quasitriangular Hopf algebras allow to talk about local symmetries and exchange statistics on the same footing [37]. In particular if the R-matrix is entangled (as for the quantum double above), the exchange statistics can be highly non-trivial, in which case they will describe the braiding of anyons.

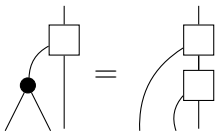
1.3 Representations of Hopf algebras

Recall that a group describes the symmetries of some space X when it acts on it (classically X is taken to be a set). If we apply the same reasoning to Hopf Algebras we have to make H act on some quantum state space (i.e Hilbert space). So our object of study is not H on its own but rather a module (or representation) of H.

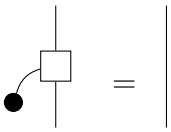
Definition 1.24 (Module)**.** *Let* H *be a bialgebra, a (left)* H*-module (or representation of* H*) is a finite dimensional vector space* V *together with a (left) action of* H *on* V*,*



satisfying the following conditions:



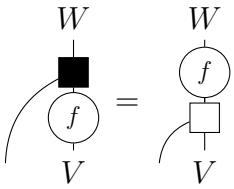
(1.20)



(1.21)

A right H-module is defined similarly with a right H-action.

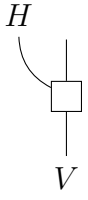
Suppose V and W are representations of H, then we say $f : V \to W$ (a linear map) is a H-module homomorphism (or intertwiner) if:



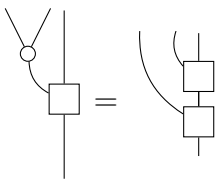
(1.22)

Where the black square denotes the action of H on W.
Dually we can define H-comodules and H-comodule homorphisms as follows.

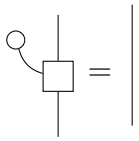
Definition 1.25 (Comodule)**.** *Let H be bialgebra, an H-comodule is a finite dimensional vector space V together with a coaction of H on V,*



satisfying the following conditions:



(1.23)



(1.24)

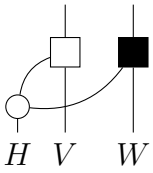
A right H-comodule is defined similarly with a right H-coaction. And H-comodule homomorphisms are linear maps which commute with the H-coaction.

Remark Although Hopf algebras $H : Hopf \to Vect$ are allowed to be infinite dimensional, we will only consider finite dimensional modules and comodules as defined above. Also we do not distinguish between representations and modules.

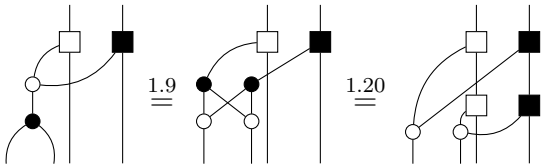
Let us consider, the category $Rep(H)$ where objects are representations of H and morphisms intertwiners. It is easy to see that the axioms of a category are satisfied, composition is just lifted from vector spaces. This category has really nice structure induced from the defining axioms of hopf algebras.

Proposition 1.4. *$Rep(H)$ is a monoidal category for any bialgebra H with tensor unit the trivial one-dimensional representation $(\mathbb{C}, \varphi)$.*

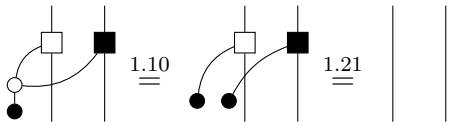
Proof. Given H-modules V and W (with white and black actions respectively), $V \otimes W$ has natural H-module structure induced by the comultiplication:



And $V \otimes W$ with this action is indeed a module as:



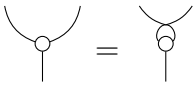
Also:



Showing that $(\mathbb{C}, \varphi)$ is the tensor unit is a trivial application of the counit law. □

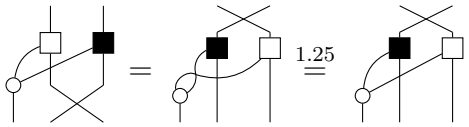
Proposition 1.5. *If H is cocommutative, then $Rep(H)$ is symmetric.*

Proof. Cocommutativity means:



(1.25)

So the symmetry morphism on $V \otimes W$ from $Vect$ is an intertwiner:

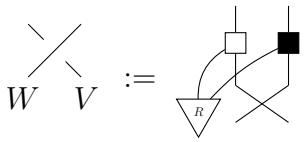


□

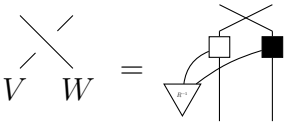
Recall that when H is cocommutative, it is trivially quasitriangular. The following is an important generalisation of the previous result.

Proposition 1.6. *If H is quasitriangular, then $Rep(H)$ is braided.*

Proof. For any H-modules V and W, using the symmetry morphism from $Vect$ define:

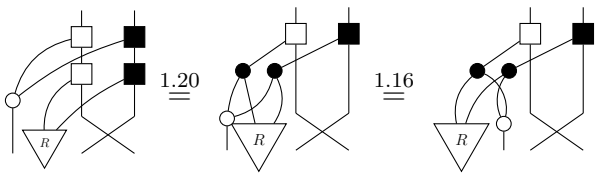


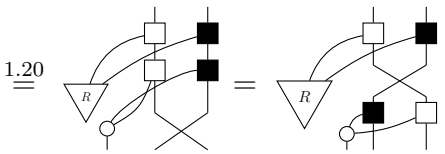
(1.26)



(1.27)

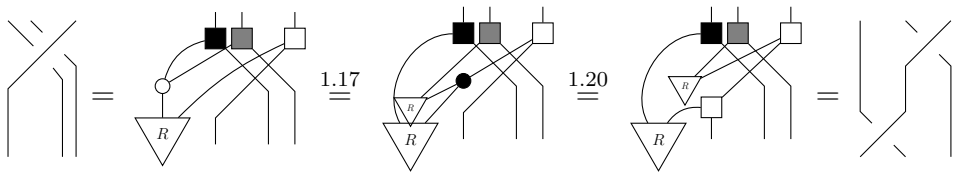
It is easy to see these are inverses of each other, we first need to check they are intertwiners.





And a similar proof works for the inverse. Now we need to show that the coherence conditions (i.e the hexagon axioms) are satisfied. The first hexagon equation follows

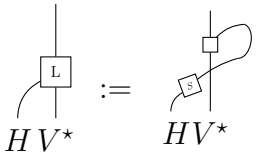
from 1.17:



And similarly the second hexagon equation follows from (1.18). □

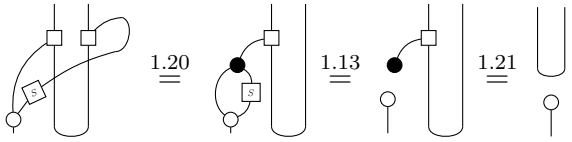
Proposition 1.7. *If H is a Hopf algebra, then $Rep(H)$ is left-rigid.*

Proof. For any H-module V let $V^\star$ be its dual in $Vect$, we can define a dual H-action on $V^\star$ using the antipode:



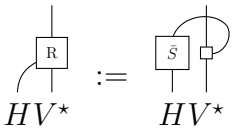
(1.28)

Then the usual cups and caps from $FVect$ are intertwiners.



A similar derivation holds for the cap. □

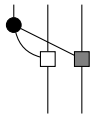
We can see that the proof relies on the existence of the antipode. If a skew-antipode $\bar{S}$ exists, $Rep(H)$ is right-rigid, where the right dual is defined:



(1.29)

The proof that this choice works is very similar to the one above. In particular, $\bar{S}$ exists when the antipode is an invertible morphism as we can define $\bar{S} = S - S^{-1}$ If the antipode coincides with the skew antipode then $Rep(H)$ then left and right duals in $Rep(H)$ coincide, we say it is rigid.

In the following chapters we will frequently use the notion of a comodule. We have only talked about the structure of the category of H-modules but all this structure can be shown to apply to H-comodules as well. For instance the category of H-comodules is monoidal with tensor product given by:



Here we give a brief argument. Note that if we flip the axioms of a Hopf algebra upside down we obtain the axioms back. An H-comodule is precisely the flipped version of an H-module, therefore all the structure of the category of H modules is also the structure of H-comodules. The definitions and proofs are simply obtained by flipping the diagrams and interchanging white with black.

Tannaka duality

Here we give a very brief exposition of Tannaka reconstruction meant as a motivation for the study of Hopf Algebras. We wont prove the reconstruction theorems, surveys on Tannaka duality are given by [38] and [19].
Reconstruction results are recipes which produce all the examples of a class of categories (i.e categories with some fixed structure) from simpler mathematical objects. As we have seen in the previous sections, the structure of categories of H-modules is induced from the axioms of the Hopf Algebra H. It is surprising that Hopf algebras underly most of the categories with this structure.

Theorem 1.8 (Tannaka reconstruction)**.**

- *Any monoidal category $\mathcal{C}$ equipped with a fiber functor (i.e a strict monoidal functor) $U : \mathcal{C} \to Vect$ is equivalent to $Rep(B)$ where B is a bialgebra.*

- *Any (braided) rigid monoidal category equipped with a fiber functor (here this means strict (braided) rigid monoidal functor) to $Vect$ is equivalent to $Rep(H)$ for some (quasitriangular) Hopf algebra H.*

Note that any category can be seen as a process theory in the sense of [1] and [10]. Objects are systems and morphisms are their possible physical evolutions. The tensor product of a monoidal category can then be regarded as a way of forming composed systems. Quantum systems usually exhibit duality (particle, antiparticle pairs) and entanglement which are captured by the rigid structure of the category. From this perspective, this reconstruction result has an interesting physical interpretation. It says that any physical theory based on vector spaces (monoidal category with fiber functor) is completely determined by the symmetries of the systems under consideration (the algebra structure). In the next section we will take this reasoning further to study physical theories of certain topological quantum systems.

Chapter 2

The Algebra of Anyons

In this chapter we introduce the physics of Anyons using the framework previously developed to define categorical models for theories of these particles. In section 2.1 we introduce the physics and make the link with braided fusion categories, in section 2.2 we develop the categorical formalism and in section 2.3 we characterise the Drinfeld center construction as a quantization procedure. For an introduction to the physics of anyons consider the foundational paper [22] or Simon's notes [36], for a categorical presentation [32] and for a survey of the mathematical aspects of anyons [34].

To understand how anyons arise physically, let us consider n indistinguishable particles evolving in space. The quantum amplitude for a space-time evolution of the system will depend on the topology of the particle word-lines and not on the detailed geometry. This means that isotopic space-time evolution will yield the same amplitude. To formalize the situation suppose we have n indistinguishable particles in D dimensions, the configuration space can be written as:

$$\mathrm{C} = (\mathbb{R}^{nD} - \Delta)/S_n$$

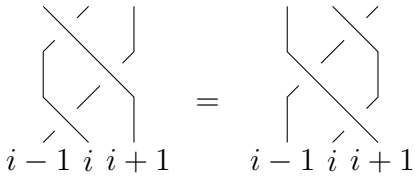
where Δ is the space of coincidences (where at least two of the n particles occupy the same position in $\mathbb{R}^D$). We are quotienting the space by S_n to account for the indistinguishability of the particles (i.e we do not care about the order of the n coordinates in D dimensions). Let us fix the starting and endpoint in the configuration space, the space of paths from starting to endpoint divides C into topologically distinct classes, described by the fundamental group $\pi_1(\mathrm{C})$. These classes account for the different possible exchange statistics of the particles.
We can then describe the evolution of the wave function for the system via unitary transformations induced from the element of the fundamental group corresponding to particles word-lines. In mathematical terms this corresponds to a representation

of $\pi_1(\mathrm{C})$.
If space-time has $D = 3+1$ dimensions, the topological class of paths is completely determined by the corresponding permutation of the particles, because there are no knots in 4 dimensions. Therefore the evolution of the system under particle exchanges will be described by a representation of the symmetric group S_n. In $2+1$ dimensions we have more exotic behaviour, as the paths in configuration space can braid. The time evolution of the wave function is then described by a representation of the braid group on n strands, denoted B_n.

Definition 2.1 (Braid group)**.** *The braid group on n strands B_n is the group generated by $\{\sigma_i : i = 1, .., n-1\}$ satisfying the following relations:*

- $\sigma_i\sigma_j = \sigma_j\sigma_i$ *for* $i + 1 < j$
- $\sigma_i\sigma_{i+1}\sigma_i = \sigma_{i+1}\sigma_i\sigma_{i+1}$ *for* $1 < i < n$.

The second relation is called Yang-Baxter equation and can be drawn as follows:



(2.1)

- Abelian case

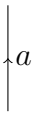
We say the system is abelian if the wave function lives in a one-dimensional representation of the group of paths in configuration space. In $3+1$ dimensions, this means we have to consider the one-dimensional representations of S_n. Note that there are only two possibilities (namely the trivial and the sign representations) corresponding to the two possible types of particle statistics in $3+1$ dimensions (Bose and Fermi statistics respectively). In $2+1$ dimensions we have many more possibilities as the evolution of the wave function will be described by a one-dimensional representation of the braid group B_n. There are infinitely many one dimensional representations of the braid group connecting the fermions and bosons case. These are described by a single parameter θ. Indeed any of the σ_i must be represented by a phase and it is easy to show, using 2.1, that all n phases have to be the same. Systems described by one-dimensional representations of B_n are called abelian anyons.

- Non-abelian case

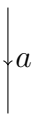
In the non-abelian case, the wave function lives in a higher-dimensional representation of $\pi_1(\mathrm{C})$. In $3+1$ dimensions again there only two types of exchange statistics. In $2+1$ dimensions it is much harder to obtain a classification of all possible behaviours under exchange. We will instead construct a procedure to obtain general theories of anyons. This means we will build a class of categories that model anyonic behaviour and induce representations of B_n. As hinted by Tannaka duality these will be induced by quasitriangular Hopf algebras and the universal R-matrix will play an important role in the description of the exchange statistics.

2.1 Models of anyons

This section is inspired from [32] and appendix E of [22]. We motivate the study of braided fusion categories, showing how they arise naturally as models of anyons.
We want to construct a category $\mathcal{C}$ that models the behaviour of anyons. Ojects of $\mathcal{C}$ will correspond to quantum systems and morphisms to their possible evolutions, or to the processes we can perform on them.
Let us first set a finite set of labels $I = \{a, b, c...\}$ of distinct particle types, these will be objects of $\mathcal{C}$. In our theory we must be able to consider many particles at the same time, so $\mathcal{C}$ must be monoidal [10]. The unit of the tensor $\mathbf{1}$ corresponds to the vacuum particle type (or "no-particle") and must be within our labels. So for the moment our theory is a monoidal category $\mathcal{C}$ and we can already use the diagrammatic language. A particle of type a evolving trivially in time is denoted:



Where we have adopted the convention that time flows upwards. Any particle a comes with its antiparticle a^* which we can picture as a particle of type a travelling backwards in time.



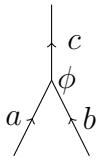
It has the property of fusing to the vacuum when it encounters a. Dually the vacuum can yield a particle-antiparticle pair, so we have cups and caps morphisms

Categorically this corresponds to a rigid structure on $\mathcal{C}$, where we have assumed that every object has two-sided duals. We will also assume the category is well behaved:

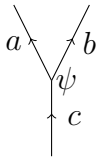
it is spherical and $1^* = 1$. This allows us to define the quantum numbers for each particle type a to be the following scalar:

$$d_a := tr(id_a) = \quad a \tag{2.2}$$

At this point we need to linearise the theory to take superpositions into account. This means we make $\mathcal{C}$ into a rigid tensor category (see appendix). We have biproducts $\oplus$ to account for superpositions. Two particles of types a and b can fuse to a third particle of type c. So we have fusion morphisms:



Similarly a particle c can split to give two particles a and b. And $\mathcal{C}$ contains splitting morphism:



In order for the fusions to behave well with superpositions we must require that our labels for particle types be simple objects in the category and all objects decompose as direct sums of simple ones ; we say $\mathcal{C}$ is semisimple (see the appendix).

Definition 2.2 (Fusion category)**.** *A fusion category is a finite semisimple $\mathbb{C}$-linear tensor category with two sided duals.*

At this point, our category $\mathcal{C}$ is a spherical fusion category and the fusion rules look like this:

$$a \otimes b \simeq \oplus_c N_{ab}^c c \tag{2.3}$$

Where $N_{ab}^c \in \mathbb{N}$. This defines a matrix for any a simple indexed by simple objects $i, j \in I$:

$$(N_a)_{i,j} = N_{ij}^a$$

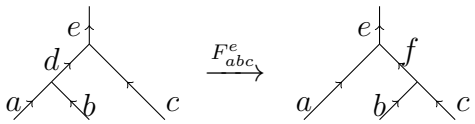
We can also define the dimension of the theory $\mathcal{C}$ as the following scalar:

$$dim(\mathcal{C}) = \sum_{i \in I} d_i^2$$

Now we know that hom-sets in our category form vector spaces. The fusion morphisms (ϕ above) are states of the fusion vector spaces V_{ab}^c and the splitting morphisms (ψ above) are states of the splitting spaces denoted V_c^{ab}. In the case of

abelian anyons all these spaces are one-dimensional, if they have higher dimension the anyons are non-abelian. In section 3.1 we will see that these spaces are where topological quantum computation takes place. The information of the theory can be captured by few matrices which are defined on the fusion or splitting spaces. We start by the F-matrix which contains the information of the fusions interacting with the quasi-associativity of the tensor product.

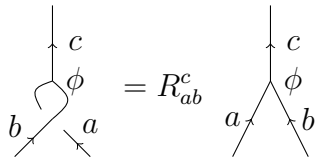
Definition 2.3 (F-matrix)**.** *Given particles types a,b,c we have two ways of fusing them to obtain particle type e, the matrix* $F^e_{abc} : \oplus_d V^d_{ab} \otimes V^e_{dc} \to \oplus_f V^e_{af} \otimes V^f_{bc}$ *is the canonical morphism:*



(2.4)

Physically the domain and codomain of the F-matrix correspond to a single fusion space V^e_{abc}. In all interesting cases (see Definition 3.2), the F-matrix is a unitary and corresponds to the canonical change of basis for V^e_{abc}.The possibilities for the F-matrices are constrained by the pentagon axiom of a monoidal category, it corresponds to a matrix representation of the associators.
We still have one important question to ask to the theory, what happens when the position of two particles is exchanged? To answer this question the theory must have a braided structure and we obtain a braided fusion category. The braided structure determines the long-distance, topological interactions between particles. Braided fusion categories induce representations of the braid group B_n, given our discussion at the beginning of this chapter, we see that they are very good candidates for describing theories of anyons. The braided structure is captured by the following piece of data:

Definition 2.4 (R-matrix)**.** *Given particle types a, b and c the matrix* $R^c_{ab} : V^c_{ba} \to V^c_{ab}$ *is the map defined by:*



(2.5)

From the first section we know that sphericality of the theory, interacting with the braided structure yields a ribbon structure. The twist θ has physical significance, it can be seen as a rotation of the particle and in most interesting cases it will be non-trivial.
In the case of abelian anyons, the twist is just a global phase, if we denote by h_a

the topological spin of the particle then $\theta_a = e^{2\pi i h_a}$ is the twist factor of a. In this scenario, the R-matrices are scalars and it is easy to see, using the definition of the twist, that the R coefficients and the twist factors are related by:

$$R^c_{ab} R^c_{ba} = \frac{\theta_c}{\theta_a \theta_b}$$

These coefficients are also constrained by the hexagon axiom of braided monoidal categories. One way to build theories of anyons, is to start by choosing fusion coefficients given by the N matrices, then constructing R matrices, twist factors and F matrices which satisfy both hexagon and pentagon axioms. However, these constraints do not fix R and F uniquely.

Example 2.1 (G-graded vector spaces)**.** *Suppose we start from a set of labels and define the fusions to form a group.* 1 *is the identity particle type, for any particle type* a*,* a^* *will be its inverse. We have defined the skeleton of a spherical fusion category, which we obtain by linearising, i.e taking a fiber functor to* $Vect$*. We obtain the category* Vec_G*, of* G *graded vector spaces over* $\mathbb{C}$*. The category* Vec_G *for* G *a group is a symmetric spherical fusion category. Linearity and tensor are given by the underlying* $Vect$ *structure, simple objects* V_g *are one-dimensional and indexed by elements* $g \in G$*, duality is proved by using the group inverse and fusions are given by the group multiplication.*

$$V_g \otimes V_h \simeq V_{gh}$$

In this case both the F *and* R *matrices are trivial.*
Tannaka duality hints that this should be a category of representations and indeed it is easy to show that $Vec_G \simeq Rep(Func(G))$ *where* $Func(G)$ *is the function algebra on* G*.*
For $G = \mathbb{Z}_2$ *we have two irreducible representations* τ_+ *and* τ_-*, both one dimensional with the obvious fusion rules given by the cyclic group of order* 2*.*

Proposition 2.1. *If* H *is a finite dimensional, semisimple, quasitriangular Hopf algebra, then* $Rep(H)$ *is a braided fusion category and* $dim(Rep(H)) = dim(H)$*.*

Proof. $Rep(H)$ is a fusion category from Theorem A.1 and from Proposition 1.6 we know $Rep(H)$ is braided. □

In 1.3, we discussed how a representation V of a Hopf algebra H can be understood as a quantum state space restricted by symmetries captured by H. When H is quasitriangular, Proposition 2.1, implies that the information of the braiding of

our quantum systems is fully contained in H. This means that H not only describes the symmetries of the system V under isolation but also its long-range interaction with other systems under exchange. The proof of Proposition 1.6 contains the form of such interaction, obtained from the universal R-matrix from the definition of quasitriangularity which perfectly matches the R-matrix of definition 2.4 . We see that when the R-atrix is separable, the interaction only results in a global phase. If the R-matrix is entangled more interesting interactions can arise from braiding. We will take this discussion further in section 2.3.

Proposition 2.1 gives us a way of building theories of anyons from Hopf algebras. The first example that comes to mind is that of a group algebra $\mathbb{C}G$. So let us suppose the theory is described by the category $RepG$. First consider the object $V = \mathbb{C}G$. It is known that $\mathbb{C}G \simeq \oplus_{i\in I} X_i \otimes X_i^*$. Simple objects X_i correspond to particle types so V can be seen as the completely mixed state. This object (i.e the vector space with it's G-action) carries all the information of the theory (by Tannaka duality) and indeed we could study the theory by just considering this algebra. We can think of elements of G as particle subtypes, particle types correspond to conjugacy classes, a state $v \in V$ is a superposition of particle subtypes. The action of G permutes the basis vectors, and corresponds to fusion. So acting with $g \in G$ on a state $v \in V$ corresponds to fusing a particle of type g with one that is in a superposition v of particle types.
In the case where G is abelian all irreducible representations are one dimensional, each corresponding to an element of the group. So really $Rep(G) \simeq Vec_G$ and behaves exactly like $\mathbb{C}G$ (without distinction between particle types and subtypes). This case is perhaps interesting philosophically as the representations of our symmetries have the same structure as the symmetries themselves [cite majid self-duality]. From a computational perspective it is a trivial situation, as only classical processes can be performed (no entanglement is possible).
If G is not abelian we must have a higher dimensional irreducible representation of G. So we could obtain more interesting processes but from a topological quantum perspective it remains a trivial case as no computational power can be obtained from the braided structure. This is because $RepG$ is symmetric as we have seen in the first section. Physically, we have seen that symmetric exchange of particles applies to fermions and bosons, from a topological perspective those types of particles can be seen as degenerate cases of anyons. Groups are therefore not enough to describe interesting anyon theories. In the next section we pin down a smaller class of categories which correspond to non-degenerate theories of anyons.

2.2 Modular categories

In this section we define modular categories and state a few results that we will use in the next section. As we have seen, braided fusion categories are well suited to describe theories of anyons. These form a big class of categories, some of which are uninteresting from the physical point of view. To distinguish between them we can place braided fusion categories in a spectrum by asking what their symmetric center Z_2 is.

Definition 2.5 (Symmetric center)**.** *If $\mathcal{C}$ is a monoidal category, the symmetric center $Z_2(\mathcal{C})$ is the full subcategory of $\mathcal{C}$ defined by:*

$$obj\, Z_2(\mathcal{C}) = \{X \in \mathcal{C} : c_{X,Y} \circ c_{Y,X} = id_{Y \otimes X} \quad \forall Y \in \mathcal{C}\}$$

It is easy to see that $\mathcal{C}$ is symmetric iff $Z_2(\mathcal{C}) = \mathcal{C}$.

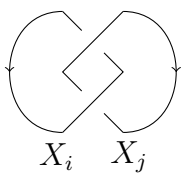
Definition 2.6 (Modular categories)**.** *A braided fusion category is:*

- *pre-modular if it is spherical,*
- *non-degenerate if $Z_2(\mathcal{C})$ is trivial (i.e it only contains direct sums of the tensor unit as objects, i.e every simple object is isomorphic to the tensor unit)*
- *modular if it is pre-modular and non-degenerate.*

The two opposite ends of this spectrum are symmetric fusion categories on one side (such that $Z_2(\mathcal{C}) = \mathcal{C}$) and modular tensor categories (as defined). In the first case, we have only symmetric exchange of quantum systems which means all particles in the theory are either bosons or fermions. Such categories exhibit no topological behaviour. Modular categories are the opposite situation, the theory doesn't contain any bosons or fermions but only non-degenerate anyons (i.e anyons with non-trivial twist). Modular categories are very well-behaved theories as we can assign to them the so called modular S-matrix which will contain all the information on fusion rules as well as the braided structure.

Definition 2.7 (S-matrix)**.** *Let $\mathcal{C}$ be a spherical braided fusion category and let I be the set of isomorphism classes of simple objects in $\mathcal{C}$. We define $S_{i,j}$ for $i, j \in I$ to be the following:*

$$S_{i,j} := tr(B_{X_j,X_i} \circ B_{X_i,X_j}) =$$



(2.6)

Remark Note that it doesn't matter on which side we take the trace by sphericality.

The following matrix contains the information of the twist structure on our modular category as it can be shown that the twist factors θ_i defined in the previous section are precisely the diagonal entries of the T-matrix.

Definition 2.8 (T-matrix)**.** *Let $\mathcal{C}$ be a spherical braided fusion category, we define the T matrix (indexed by I) given by*

$$T_{i,j} := \delta_{i,j} tr(B_{X_i,X_j}) = \delta_{i,j} \quad \underset{X_i}{\cap} \quad \underset{X_j}{\cap} \tag{2.7}$$

Remark Categories of this type are called modular as it can be shown that S and T satisfy the same relations as the generators of the modular group $SL(2,\mathbb{Z})$, so that any modular category induces a representation of $SL(2,\mathbb{Z})$. It is a conjecture that the S and T matrices determine modular categories up to ribbon equivalence.

Definition 2.9 (Mueger centralizer)**.** *If $\mathcal{D}$ is a full (tensor) subcategory of $\mathcal{C}$ can define $C_{\mathcal{C}}(D)$ to be the full subcategory such that*

$$obj(C_{\mathcal{C}}(D)) = \{X \in obj(\mathcal{C}) : B_{X,Y} \circ B_{Y,X} = id_{X\otimes Y}\}$$

It is easy to check this is indeed a monoidal subcategory and it is replete (i.e closed under isomorphisms) [29]. Also note that $Z_2(\mathcal{C}) = C_{\mathcal{C}}(\mathcal{C})$. The following result is one of the most important pure category theoretic results on modular categories and we will need it for the discussion on the Drinfeld center. We also state other important results demonstrating the importance of the S-matrix in modular categories.

Theorem 2.2 (Mueger decomposition)**.** *[29, Theorem 4.2] Let $\mathcal{C}$ be a modular category and $\mathcal{K}$ a semisimple full tensor subcategory, then there is an equivalence of braided fusion categories:*

$$\mathcal{C} \simeq \mathcal{K} \boxtimes C_{\mathcal{C}}(\mathcal{K})$$

Theorem 2.3. *$\mathcal{C}$ is modular iff the S-matrix is invertible.*

Proof. Suppose $\mathcal{C}$ is not modular, then $Z_2(\mathcal{C})$ is non-trivial $\implies$ there is a non-trivial simple object a such that its braiding is the symmetry. Therefore $S_{a,i} = d_a d_i$ for all $i \in I$, but also $S_{1,i} = d_i$ and so the first and ath rows of S are proportional $\implies$ S is not invertible.
The other direction is less easy and can be found in [5] and [29]. □

Proposition 2.4. *[5, Proposition 3.1.12] The modular S-matrix diagonalises the N-matrix.*

Therefore the S-matrix contain all the information of the fusion rules, and with some algebraic manipulation (which can be found in [5]) we obtain the well-known Verlinde formula for the fusion coefficients:

$$N_{ij}^{k} = \sum_{r \in I} \frac{S_{ir} S_{jr} S_{kr}}{S_{1r}} \tag{2.8}$$

Remark Given any modular category we can use the Turaev-Viro construction [6], which yields a $2+1$ topological quantum field theory. For our purposes we only need to view the modular category as a process theory of anyons in the sense of [1] and we won't introduce the $TQFT$ formalism.

Example 2.2 (Fibonacci anyons)**.** *The category Fib of Fibonacci anyons is one of the most popular examples of modular categories as it has a purely algebraic formulation. Anyons of this type are non-abelian and complete for topological quantum computation [32]. We will meet them again in the next chapter.*
Fib has only two simple objects: τ and the vacuum type 1. The fusion rules are given by:

$$1 \otimes \tau = \tau = \tau \otimes 1$$

$$\tau \otimes \tau = 1 \oplus \tau$$

It turns out that those equations together with the hexagon and pentagon constraints completely determine a modular category [36].

2.3 The Drinfeld center

In this section we introduce a general construction that turns braided fusion categories into modular categories, and we show its relationship with the Quantum double construction on a Hopf Algebra introduced in the first chapter. The proofs and results of this section were developed with Amar Hadzihasanovic.
Topological dependencies between objects in fusion categories are captured by the braided structure. Let us fix some definitions before discussing the Drinfeld construction.

Definition 2.10 (Half-braiding)**.** *A half-braiding on some object X in a monoidal category $\mathcal{C}$ is a natural isomorphism*

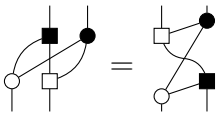
$$e^{X} : X \otimes (-) \Rightarrow (-) \otimes X$$

satisfying the compatibility condition:

$$e^{X}_{Y \otimes Z} = (id_Y \otimes e^{X}_{Z}) \circ (e^{X}_{Y} \otimes id_Z)$$

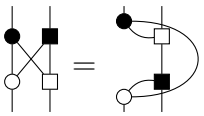
Definition 2.11 (Drinfeld center)**.** *The braided (Drinfeld) center of a monoidal category $\mathcal{C}$ is the category $Z(\mathcal{C})$ with objects pairs (X, e^X) where $X \in \mathcal{C}$ and e^X is a half-braiding, and with morphisms given by the morphisms of $\mathcal{C}$ which commute with the half-braiding.*

Definition 2.12 (Yetter-Drinfeld modules)**.** *Let H be a bialgebra, the category $\mathcal{D}_H^{lr}$ is the category of left-right Yetter-Drinfeld modules where objects are left H-modules which are simultaneously right H-comodules satisfying the following compatibility condition:*



(2.9)

where the white box denotes the right H-coaction and the black box denotes the left H-action. Morphisms of $\mathcal{D}_H^{lr}$ are both H-module and H-comodule morphisms. Left-left Yetter-Drinfeld modules are defined in the obvious way and form a category $\mathcal{D}_H^{ll}$. The compatibility condition is then the following:



(2.10)

Proposition 2.5. *Let $\mathcal{C}$ be a monoidal category, then $Z(\mathcal{C})$ is braided monoidal.*

Proof. It is easy to check that defining the tensor as $(X \otimes Y, e_Z^{X\otimes Y} = (e_Z^X \otimes id_Y) \circ (id_X \otimes e^Y(Z))$ and the braiding as e_Y^X yields a braided monoidal structure on $Z(\mathcal{C})$. □

The following proposition hints to the relationship between the Drinfeld center and the quantum double.

Proposition 2.6. *[31, Theorem 7.1] The Drinfeld center of a spherical fusion category is modular. And $dim(Z(\mathcal{C})) = dim(\mathcal{C})^2$*

In general $Z(\mathcal{C})$ is not symmetric as we will see, but in the case of $Vect$ the Drinfeld construction is trivial.

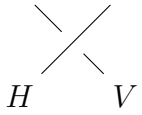
Proposition 2.7. $Z(Vect) \simeq Vect$

Proof. Using the Mueger decomposition (Theorem 2.2), note that $Z(Vect)$ is modular and $Vect$ is a full fusion subcategory of $Z(Vect)$, therefore

$$Z(Vect) \simeq Vect \boxtimes C_{Z(Vect)}(Vect)$$

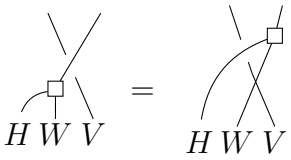
But if (A, e^A) is an object of $C_{Z(Vect)}(Vect)$ then any component e_B^A must be the inverse of the symmetry morphism on $A \otimes B \implies$ it must be the symmetry morphism $\implies C_{Z(Vect)}(Vect) \simeq Vect$ and so $Z(Vect) \simeq Vect$. □

We now aim to prove theorems 2.12 and 2.13 that relate Yetter-Drinfeld modules and the Drinfeld center construction.
Fix a bialgebra H and suppose $\left(V, \rightharpoonup\right) \in obj(Rep(H))$ and (V, e_V) is in $Z(Rep(H))$. Note that H has a natural H-module structure given by right multiplication. Consider the component of the half-braiding of H at V.

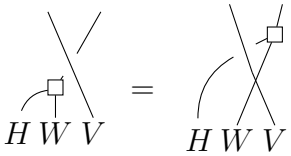


In the arguments that follow we will use repeatedly the following trick which we state as a Lemma, it exploits the copy of $Vect$ which lives inside any category of representations.

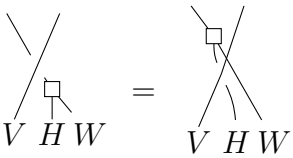
Lemma 2.8. *For any W object of $Rep(H)$ with white action and V as above.*



(2.11)



(2.12)



(2.13)

Proof. Note that $\left(H \otimes W, \rightharpoonup |\right)$ is in $Rep(H)$ and

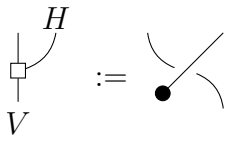
$$\rightharpoonup\square : \left(H \otimes W, \rightharpoonup |\right) \to \left(W, \rightharpoonup\square\right)$$

is an intertwiner by the module law. Also it is easy to check that the symmetry morphism lifted from $Vect$

$$\left(W, |\right) \otimes V \to V \otimes \left(W, |\right)$$

is an intertwiner. And it follows from $Z(Vect) = Vect$ that it must be the W-component (where W has the trivial action) of the half braiding on V as W lives in the copy of $Vect$ in $Rep(H)$. The equations then follow from naturality of the half -braiding. □

Define a right coaction of H on V:

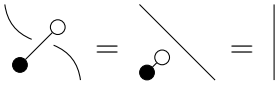


(2.14)

Note that, from the bialgebra laws,  and  (seen as morphisms on the H-module H) are intertwiners in $Rep(H)$. Therefore by naturality of the half braiding we get:

(2.15)

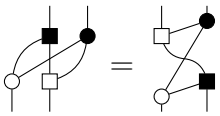
and



(2.16)

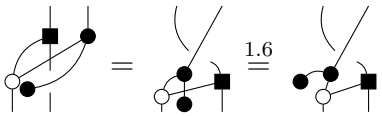
So that the coaction indeed defines a left H-comodule.

Claim 1.



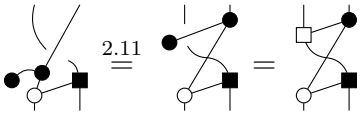
(2.17)

Proof. As the braiding is an intertwiner, it commutes with the action of H on $V \otimes H$, therefore:



(2.18)

Therefore by Lemma 2.8 with $W := H$ (with the module structure given by multiplication):

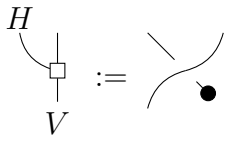


(2.19)

□

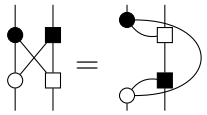
We have defined a functor $F_1 : Z(Rep(H)) \to \mathcal{D}_H^{lr}$ which is the identity on arrows (hence faithful) and sends (V, e_V) to the left-right Yetter-Drinfeld module with black H action and white H coaction. To see this, note that if an H-module morphism f is in $Z(Rep(H))$ then it commutes with the half-brading, in particular it commutes with the H-component of the half-braiding and therefore it commutes with the H-coaction as defined.
Similarly we can define a functor $F_2 : Z(Rep(H)) \to \mathcal{D}_H^{ll}$ by considering the H

component of the half braiding on V and defining the following left-coaction:



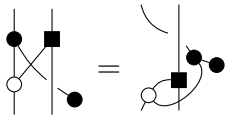
(2.20)

Claim 2.



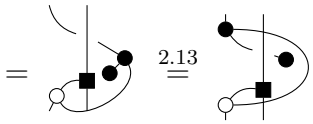
(2.21)

Proof. The proof is very similar to that of the previous claim. Using the fact that the braid is an intertwiner we obtain



(2.22)

Then using the unit law and the same trick as before we see that



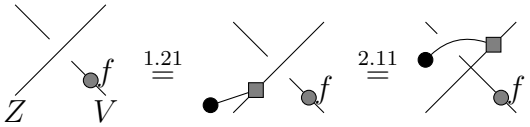
(2.23)

□

For the same reasons as for F_1, F_2 is faithful. To show F_1 and F_2 are equivalences of categories we still need to show they are full and essentially surjective.

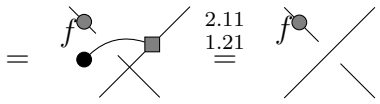
Proposition 2.9. *If H is a bialgebra, then $F_1 : Z(Rep(H)) \to \mathcal{D}_H^{lr}$ and $F_2 : Z(Rep(H)) \to \mathcal{D}_H^{ll}$ are full and faithful.*

Proof. Note that in all previous claims and definitions we have only assumed that H is a bialgebra. We have already shown that F_1 and F_2 are faithful, it remains to show they are full.
Suppose f is a morphism $V \to W$ in $\mathcal{D}_H^{lr}$, then using the Lemma we see that for any Z in $Rep(H)$ with gray H-action:



(2.24)

And by definition of f, it commutes with the coaction so that:



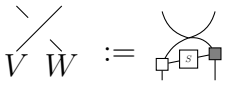
(2.25)

So f commutes with the coaction $\implies$ it is a morphism in $Z(Rep(H))$. Therefore F_1 is full. And a similar proof applies to F_2. □

Until now we have only assumed H is a bialgebra.

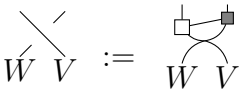
Proposition 2.10. *If H is a Hopf algebra, F_1 is essentially surjective.*

Proof. To prove this we construct a half braiding for any object V of $\mathcal{D}_H^{lr}$ which yields the coaction of the form (2.14).
Fix any object V with white right H-coaction and for any $(W, \blacksquare)$ define



(2.26)

(2.26) is an isomorphism as

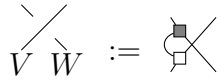


(2.27)

is an inverse by the hopf law. It is natural in W as all morphisms are intertwiners (so they commute with the H-action on W). And it satisfies the compatibility condition by definition of H-comodule. Clearly setting $W = H$ in (2.27) with the natural left-multiplication action, and inserting $\bullet$ on the left of the tensor yields the H-coaction (2.14). □

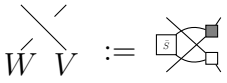
Proposition 2.11. *If H has a skew antipode, F_2 is essentially surjective.*

Proof. Define



(2.28)

The same argument as the previous proposition applies defining the inverse using the skew antipode $\bar{S}$:



(2.29)

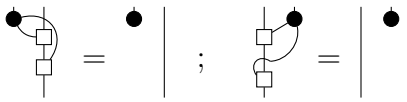
□

We have proved the following theorem.

Theorem 2.12. *Let H be a bialgebra. If H is a Hopf algebra then $Z(RepH) \simeq \mathcal{D}_H^{lr}$. If H has a skew antipode then $Z(RepH) \simeq \mathcal{D}_H^{ll}$.*

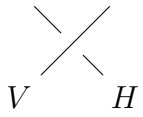
We now want to obtain a generalization of Theorem 2.12 to the case where H is only a bialgebra. To do this we need to make the two kinds of Yetter Drinfeld modules interact with one another.

Definition 2.13 (Interacting Yetter-Drinfeld modules)**.** *For a bialgebra H, the category $\mathcal{YD}_H$ has objects given by left H-modules which are also left H-comodules and right H-comodules such that both the left-left and left-right Yetter Drinfeld compatibility conditions are satisfied and the comodules structures additionally satisfy:*

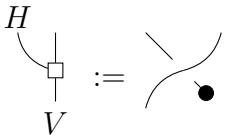


(2.30)

We now define a functor $F : Z(Rep(H)) \to \mathcal{YD}_H$ in a similar fashion to F_1 and F_2 as above. Fix any object V of $Z(Rep(H))$ and consider the component e_H^V of the half braiding on V at H.

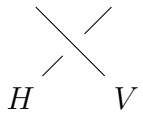


Define a left H-comodule structure on V by:

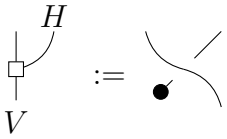


(2.31)

Note that this is precisely the left H-comodule structure in the definition of F_2. As the half-braiding on V is a natural isomorphism we can consider the inverse of the H component:



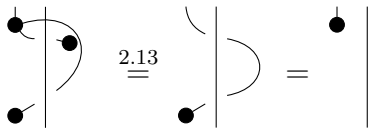
And define the right H-comodule structure on V by:



(2.32)

In general this is not the same coaction as in the definition of F_1. (2.31) and (2.32) define a mapping $F : Z(Rep(H)) \to \mathcal{YD}_H$ for a bialgebra H. We now show F is an equivalence of categories.

- To see that F is well-defined on objects note that (2.31) and (2.32) satisfy left-left and left-right Yetter-Drinfeld compatibility conditions (resp.) by claim 2 and an adaptation of claim 1 where overcrossing is replaced by undercrossing (resp.). The additional compatibility conditions of $\mathcal{YD}_H$ are easily seen to hold.Indeed using the Lemma 2.8 we have

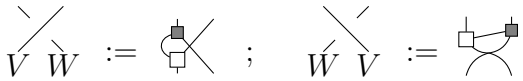


(2.33)

- As for F_1 and F_2 we can define F to be identity on arrows. This is because if some linear map commutes with the half braiding then it commutes with both left and right H-coactions. So F is well-defined on arrows and it is faithful.

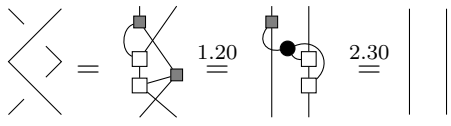
- The proof that F is full is the same as for Proposition 2.9, except it uses the (2.12) and (2.13) instead of (2.11) from Lemma 2.8.

- To see F is essentially surjective. Fix any object V of $\mathcal{YD}_H$ (with white left and right comodule structures). Then for any $(W, \blacksquare)$ in $Rep(H)$ define:



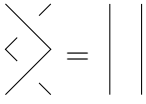
(2.34)

  These are inverses of each other as:



(2.35)

  and similarly



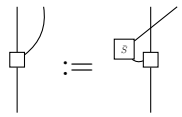
(2.36)

  from the second compatibility condition of interacting Yetter-Drinfeld modules. Naturality is easy to check. And setting $W = H$ and plugging units $\bullet$ in the H-components we recover the left and right H-coactions defined.

This proves the following result.

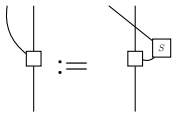
Theorem 2.13. *If H is a bialgebra then $Z(Rep(H)) \simeq \mathcal{YD}_H$.*

If H is a hopf algebra or has a skew antipode we recover Propositions 2.10 and 2.11. Indeed given a skew antipode and left H-comodule structure we can define



(2.37)

to obtain a right H-comodule structure. And similarly given an antipode and a right H-comodule structure we obtain a left H-comodule structure by defining:

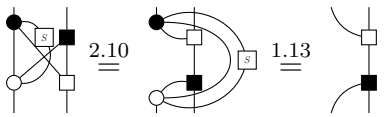


(2.38)

When H is a finite-dimensional Hopf algebra we obtain the better known results below.

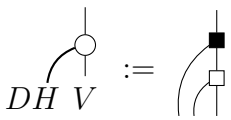
Theorem 2.14. *If H is a finite dimensional Hopf algebra with invertible antipode $Z(RepH) \simeq RepDH$*

Proof. Note that defining DH requires the antipode to be invertible (it is used in the definition of the antipode for DH), so such condition is inevitable. As H has invertible antipode it also has a skew antipode. Therefore we already know $Z(Rep(H)) \simeq \mathcal{D}_H^{ll} \simeq \mathcal{D}_H^{lr}$ From theorem 2.12. So it is sufficient to show that $Rep(DH) \simeq \mathcal{D}_H^{ll}$. Fix an object of $\mathcal{D}_H^{ll}$ with black H-action and white H-coaction. Making use of the antipode and the compatibility condition we obtain:



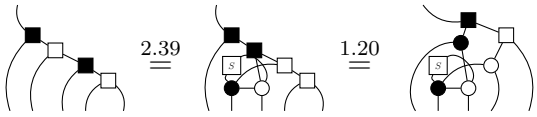
(2.39)

As H is finite dimensional, we can define the action of DH on V as follows (where thick wires carry DH and thin wires carry H)



(2.40)

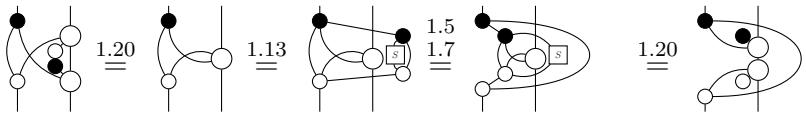
This action gives V a DH-module structure as:



(2.41)

Now it is easy to see that morphisms commute with the DH-action iff they commute

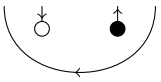
with the H-action and H-coaction. So we have defined a fully faithful embedding of $\mathcal{D}_H^{ll}$ into $Rep(DH)$. To see it is essentially surjective note that, given a DH-action on V, we can recover the H-action by plugging the counit in the $H^\star$ component of the DH-action and the H-coaction by plugging the unit in the H-component and bending the $H^\star$ wire up. It remains to check that those indeed define a left-left Yetter-Drinfeld module in all cases, i.e that the compatibility condition is satisfied. And it is indeed the case:



(2.42)

□

Recall our discussion on the universal R-matrix of quasitriangular Hopf algebras from section 2.1. We noted that when the universal R-matrix is entangled it induces non-trivial interactions when braiding two representations. Now by the definition of the quantum double, we know the R-matrix looks like this:



It can be shown that using this R-matrix as in the proof of Proposition 1.6 we recover the half braiding of $Z(Rep(H))$. Also note that the R-matrix is entangled, so that the interaction under braiding is necessarily non-trivial. This means the braiding is not the symmetry corroborating the fact that $Z(Rep(H))$ is modular.
We have found many equivalent ways of constructing non-degenerate theories of anyons. In the next section we will use the simplest examples induced by groups, justified by the following proposition

Proposition 2.15. *If G is a finite non-abelian group then $Rep(D(G))$ is modular.*

Proof. This follows from the fact that $Z(Rep(G))$ is modular. A direct proof is given in the third chapter of [5]. □

Chapter 3

Quantum Computation

In the previous chapter we saw that categories can be interpreted as physical process theories. In a similar way, we can interpret objects as data types and morphisms as computational processes, so that any category corresponds to a theory of computation. Quantum computation is a model in which data is encoded and processed on quantum systems. A computation consists of the preparation of some quantum states, the implementation of some unitary transformation of the system followed by measurement. Usually this procedure is repeated in order to collect statistics and approximate density distributions. The unit of information in quantum computation is called qubit by analogy with the classical bit. A qubit is a two-level quantum system, that is a Hilbert space of dimension 2 which is denoted by $\mathbb{C}^2$. Similarly a qudit is a d dimensional system $\mathbb{C}^d$.

In section 3.1 we introduce a model of quantum computation based on anyons. The model, first introduced by Kitaev [21], is obtained by starting with a lattice of particles and imposing some dynamics (in the form of a Hamiltonian). We will see that this physical procedure illustrates the categorical construction (Drinfeld center) that was presented in section 2.3. Quantum computation by anyons has the advantage of being fault-tolerant and is therefore the object of applied research by Microsoft [15].

In section 3.2 we discuss the less known model of Permutational Quantum Computing (PQC) discovered by Jordan [18]. It is by nature a very restricted model, which was introduced as an argument for the supremacy of quantum over classical computation. Indeed, despite its restrictiveness, it can solve classically hard problems in polynomial time [18]. Instead of focusing on computational complexity, we will extract the salient features of PQC into a categorical formalism to identify in what sense it is 'restricted'.

Section 3.3 is an illustration of functorial semantics in the context of quantum computation. We use the same modular categories which arise in Kitaev's construction

as syntax instead of semantic categories. This results in a description of quantum gates consisting in braids.

3.1 Topological Quantum Computation

Modular categories are models for Topological Quantum Computation (TQC) in the sense of [34] or [21]. In TQC, data is encoded in non-abelian anyons and quantum gates are obtained by braiding those particles. Topologically equivalent braids implement the same quantum process so that small perturbations of particle world-lines do not affect the computation and gates (and quantum information) are topologically protected from decoherence. Another reason for studying topological quantum computation is that some TQC models allow to efficiently approximate the Jones polynomial, a problem that is believed to be untractable classically.

Definition 3.1 (TQC)**.** *A topological quantum computer runs as follows [34]:*

1. *Creation of anyon pairs from the vacuum to encode the information as a quantum state.*

2. *Braiding those anyons performs a quantum gate on the state.*

3. *Fusing neighbouring anyons and observing the resulting anyon type corresponds to a projective measurement on the system.*

The computation result is the approximation to a probability distribution (over measurement outcomes) obtained by repeating the procedure polynomially many times and recording the output anyon types. Note that if we postselct on the vacuum sector to be the output anyon type we are effetively approximating an invariant of links. Indeed any process in TQC starting and ending in the vacuum sector is a link, formed by the particle trajectories in space-time (i.e the briading process). The operations we can perform on a system V are unitaries, so that any braiding process on n particles induces the evaluation of some unitary representation $\beta \to U_\beta$ of the braid group B_n.
In order to make sure the braiding process is a unitary transformation of the state space we will assume one further constraint on our categorical model of computation: unitarity of the braids in the modular category in question.

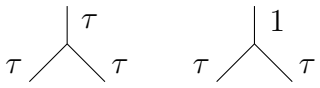
Definition 3.2 (UMC)**.** *A unitary modular category (UMC) is a modular category where each component of the braiding and of the associators is a unitary.*

This is important for quantum computation as unitaries are the only processes we can physically implement. Given a UMC $\mathcal{C}$, we have a finite set of data types I given by the isomorphism classes of simple objects in $\mathcal{C}$. The topological qudit is usually encoded in some fusion space as follows. We fix some data type a and consider the fusion of n copies of a. We then choose some output type b on which to post-select in order to obtain a fusion space $V := V^b_{a^{\otimes n}} = Hom(a^{\otimes n}, b)$ of dimension d. We can picture standard basis as a labelled binary fusion trees with b at the root and a on the n leaves. Each binary tree shape (with n leaves) corresponds to a different bracketing of $a^{\otimes n}$ and usually yields distinct bases related to each other via F-moves. Braiding a pair of type a anyons before fusing is an R-move. All computational processes on V can be decomposed in sequences of F-moves (re-bracketing) and R-moves (braiding).

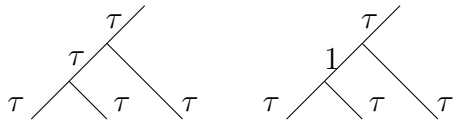
Example 3.1 (Fibonacci anyons)**.** *We now look back at example [ref] on Fibonacci anyons and show how to compute in the model. Recall we have only two particle types: the vacuum sector* 1 *and the non-trivial* τ *such that:*

$$\tau \otimes \tau = 1 \oplus \tau$$

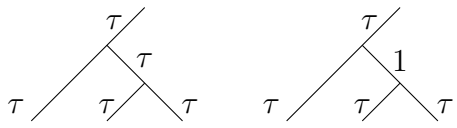
τ *and* 1 *are their own anti-particles so we don't need to distinguish particles and antiparticles by wrinting arrows on wires. We can write the basis of the two dimensional space* $\tau \otimes \tau$ *as:*



The fusion space $V^\tau_{\tau^{\otimes 3}}$ *is two dimensional, we will take it as our computational space. Practically, this corresponds to considering three* τ *particles with overall charge (type)* τ*. This space is our topological qubit and we write the computational basis as:*



Let's denote by $|0\rangle, |1\rangle$ *these basis states. Another basis is given by fusing the left-most two anyons first:*



And we denote them by $|+\rangle, |-\rangle$. *These two bases are linked by a unitary* 2×2 *transformation* $F := F^{\tau}_{\tau^{\otimes 3}}$ *given by the solution of the following system:*

$$|0\rangle = F_{0+} |+\rangle + F_{0-} |-\rangle$$

$$|1\rangle = F_{1+} |+\rangle + F_{1-} |-\rangle$$

To derive the form of the F*-matrix we need to consider the pentagon axiom. It turns out that for the Fibonacci model the pentagon is enough to derive the* F*-matrix but it is not the case in general. The resulting* F*-matrix is [36]:*

$$F^{\tau}_{\tau^{\otimes 3}} = \begin{bmatrix} \phi^{-1} & \phi^{-\frac{1}{2}} \\ \phi^{-\frac{1}{2}} & -\phi^{-1} \end{bmatrix} \tag{3.1}$$

where $\phi = \frac{\sqrt{5}-1}{2}$. *Given the* F*-matrix and the two hexagon axioms for braided monoidal categories the possibilities for the* R*-matrix are few. In this case there is only one possibility and we obtain the* R*-matrix:*

$$R^{\tau}_{\tau^{\otimes 3}} = \begin{bmatrix} e^{-4\pi i/5} & 0 \\ 0 & -e^{-2\pi i/5} \end{bmatrix} \tag{3.2}$$

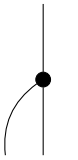
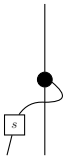
In [32] it is shown that the Fibonacci model allows universal quantum computation. This is done by first noting that polynomially many R *and* F *matrices as above can approximate any unitary on one qubit, and then by constructing a CNOT gate on two topological qubits.*

Kitaev's quantum double model

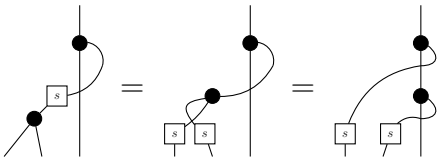
Kitaev's quantum double models originate in [21] and are induced by finite groups . As we discussed at the beginning of the chapter it will illustrate the Drinfeld center construction studied in 2.3 and in particular the $Z(Rep(G)) \simeq Rep(DG)$ for a group G. In 2.3 we obtained many variations of this result working with the more general framework of Hopf algebras. In view of these generalisations we will present analogous variations of Kitaev's quantum double construction.

Suppose we have particles living in state space H where H is a Hopf algebra (with black multiplication, white comultiplication and antipode S as usual). We can define two canonical types of left H-module structures on H given by the right and left

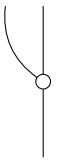
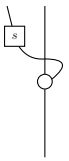
multiplication as follows:

$L_+ =$  ; $L_- =$  (3.3)

Right multiplication defines a module by associativity, left multiplication also works because the antipode is an algebra anti-morphism:

 (3.4)

Similarly there are two canonical left H-comodule structures on H given by left and right comultiplication.

$T_+ =$  ; $T_- =$  (3.5)

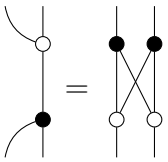
And the proofs that these are H-comodules are dual to the previous ones. Kitaev considers the case where $H = \mathbb{C}G$ for some group G. The above module and comodule structures then yield 4 types of linear operators on $\mathbb{C}G$: $L^g_\pm$, $T^h_\pm$ (using the notation from [21]), indexed by elements $g, h \in G$, and defined as follows:

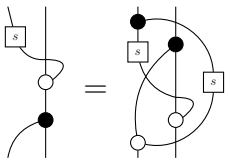
$$\begin{aligned} L^g_+ |z\rangle &= |gz\rangle \quad L^g_- |z\rangle = |zg^{-1}\rangle \\ T^h_+ |z\rangle &= \delta_{h,z} |z\rangle \quad T^h_- |z\rangle = \delta_{h^{-1},z} |z\rangle \end{aligned} \tag{3.6}$$

Note that L operators commute with each other and T operators too. The non-trivial commutation relations are the following:

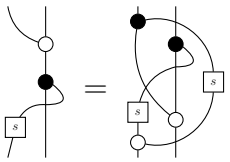
$$\begin{aligned} L^g_+ T^h_+ &= T^{gh}_+ L^g_+ \quad L^g_+ T^h_- = T^{hg^{-1}}_- L^g_+ \\ L^g_- T^h_+ &= T^{hg^{-1}}_+ L^g_- \quad L^g_- T^h_- = T^{gh}_- L^g_- \end{aligned} \tag{3.7}$$

In the general Hopf algebra case those commutation relations correspond to the following equalities of diagrams, which are easily obtained using the bialgebra law (1.9) and the fact that the antipode is an algebra and coalgebra anti-morphism (Proposition 1.2).

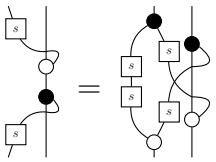
 (3.8)



(3.9)



(3.10)



(3.11)

We then consider a lattice with oriented edges embedded in some $2D$ oriented manifold (e.g a sphere or a torus) with particles on the edges. For any vertex s and adjacent plaquette p, a site is defined as the pair $a = (s, p)$. Let $star(a)$ be the set of edges adjacent to s and $bound(p) = \{j_1, ..j_k\}$ the ordered set of edges adjacent to plaquette p starting and ending at vertex s. Every edge $j \in star(s)$ on the lattice has an orientation and in [21] is defined $L^g(j, s)$ to be L^g_- applied to vertex j when s is the origin of j and L^g_+ otherwise. Similarly $T^h(j, p)$ is defined to be T^h_+ (repectively T^h_-) if j is on the right (resp. on the left) of j. The following operators are then defined at each site a of the lattice:

$$\begin{aligned} A_g(a) &= \prod_{j \in star(s)} L^g(j, s) \\ P_h(a) &= \sum_{h_1 ... h_k = h} \prod_{m=1}^{k} T^{h_m}(j_m, p) \end{aligned} \tag{3.12}$$

We will write the general diagrammatic form of these operators in the next section. For the moment let us go through Kitaev's reasoning [21].

From a physical point of view P_h operators can be understood as measuring the magnetic flux of the system at some site and A_g are local symmetry transformations on the charge. Flux measurements are projection $P_h \in \mathbb{C}G^*$ onto flux sector h. The allowed residual global symmetry transformations are then implemented via A_g for $g \in N(h)$.

The projectors form a Von Neumann family and satisfy

$$P_h P_{h'} = \delta_{h,h'} P_h.$$

Operators A_g are global symmetry transformation

$$A_g A_h = A_g h$$

and affect the fluxes via conjugation:

$$A_g P_h = P_{ghg^{-1}} A_g \tag{3.13}$$

(this was shown was shown by Kitaev [21]). Operators A_g and P_h generate the algebra DG. So the quantum double construction allows to capture both global symmetry transformations and projective measurements in one algebraic structure. It is easy to check, rewriting the definition, that the following is true.

Proposition 3.1. *For any finite group G, its quantum double $D(G)$ is the algebra generated by $\{P_h A_g\}_{h,g\in G}$ with multiplication induced by (3.13), comultiplication and antipode as defined in [first section].*

$D(G)$ has a natural quasi-triangular structure witnessed by the universal R-matrix $R = \sum_{g,h\in G} P_h e \otimes P_h g$, making $RepDG$ braided.
Kitaev then builds a Hamiltonian for the system and shows that the ground state of the Hamiltonian is an irreducible representations of DG. Here we will skip this part of the reasoning and rely on the intuition that the operators A_g and P_h correspond to the symmetries of the system, i.e the dynamics which are 'constantly being applied'. So the allowed processes of the systems are processes that commute with all of those operators, i.e the system lives in a representation of DG. The ground state of the Hamiltonian has degeneracy 4^g where g is the genus of the surface in which we embedded the lattice.
In the case of a sphere, $g = 0$, so there is no degeneracy and the overall system lives in a one dimensional (trivial) representation of DG, the vacuum sector. An excitation can arise at some site on the lattice when the constraints given by the Hamiltonian are violated. In representation theoretic terms this corresponds to the creation of a state of some higher dimensional irreducible representation of DG. Those excitations (or quasi-particles) are anyons and can only be created in pairs (particle-antiparticle pairs). When the lattice is 'layered' enough (i.e contains many particles) we can move those excitations on the lattice (practically this is done by applying charge and flux operators at given sites on the lattice). All the excitations can then be fused pairwise to end back in the vacuum sector and obtain fusion results. There are different possible types of excitations (anyon flavours) we can create on the lattice, corresponding to different possible violations of the constraints. These are precisely indexed by the irreducible representations of DG. We can see that we have obtained a physical setting giving rise to anyons whose behaviour is modeled

by the modular category $Rep(DG)$.
In order to understand the possible anyon types in the model, we must study the irreducible representations of the quantum double finite group algebra DG. This has been done by Gould [16], who showed that irreducible representations of DG are obtained in the following way.
Let $\{C_i\}_{i=1}^n$ be the distinct conjugacy classes in G. To each of those conjugacy classes corresponds a centralizer subgroup N_i (two choices of representatives for C_i yield isomorphic centralizer subgroups). Then for any irreducible representation (α, V_α^i) of N_i with basis elements v_j^α, let $V_{i,\alpha} = \mathbb{C}C_i \otimes V_\alpha^i$, this has basis $\{|k, v_j^\alpha\rangle\}_{j=1,\ldots,dim\alpha}^{k\in C_i}$ and forms an irreducible representation of $D(G)$ under the action

$$P_h g\,|k, v_j^\alpha\rangle = \delta_{h,gkg^{-1}}\,|h, \alpha(h^{-1}gk)v_j^\alpha\rangle \tag{3.14}$$

and the $\{V_{i,\alpha}\}$ is the complete set of irreducible representations. When G is abelian, all irreducible representations are one-dimensional and we only have abelian anyons which are very unlikely to be universal for quantum computation. It was shown by Kitaev that if $G = S_5$ the model is universal for quantum computation. In [23] the $G = S_3$ model was described thoroughly but not shown to be universal.

Example 3.2 (Quantum memory)**.** *The case where $G \simeq \mathbb{Z}_2$ gives rise to Kitaev's toric code. Note that $D(\mathbb{Z}_2) \simeq \mathbb{C}(\mathbb{Z}_2 \times \mathbb{Z}_2^\star)$, so that there are 4 irreducible representations, all of which are 1-dimensional. Each of those corresponds to a different type of excitation. Let x and y be the generators of the group. The trivial representation is the trivial excitation (or 'no excitation'). The other irreducible representations are obtained by mapping x and y to order 2 elements of $\mathbb{C}$. We obtain two bosons, when both get sent to -1 or i and one fermions when $x \mapsto -1$ and $y \mapsto i$.*
If we implement the construction on a lattice embedded on a torus, we obtain a model for a topologically protected quantum memory. Consider a (layered enough) lattice on a torus with spins on the edges. Let C_1 and C_2 be two cycles. States of the system are generated by labellings of the lattice with elements of $\mathbb{Z}_2$. As shown by Kitaev, the ground state degeneracy has dimension 4, so that we can think of the system as storing two qubits of information. If a particle-antiparticle pair is created at some site on the lattice it will re annihilate at some other site. The world-lines will form a loop on the torus and we have three possible behaviours. If the loop can be shrinked to a point, this won't affect the underlying information otherwise we obtain two non-trivial operations T_1 and T_2 affecting the ground state when the world-lines loop around cycle C_1 and C_2. If we initialise the lattice in some ground state it will remain in that state unless a T_1 or T_2 operation is implemented. If the lattice is layered enough, it is very unlikely that such processes occur spontaneously, and therefore the quantum information is protected.

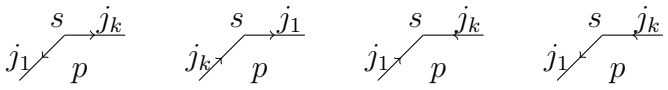
Generalising the model

We will now try to generalize the above construction to the cases where the Hopf algebra H is not a group algebra. We will wait before 'bending cables' (i.e using the dual Hopf algebra) to see how far we can go without making too many assumptions on H. This will provide an interesting illustration of the proof given in section 2.3. Interpreted in this physical context, the Drinfeld construction can be understood as imposing global (or topological) dependencies on the particles under consideration (e.g in the form of a Hamiltonian as the one considered by Kitaev) giving rise to anyonic behaviour.

As above, we have an oriented lattice on a $2D$ oriented manifold with particles on the edges taking values in H. For simplicity we will assume the manifold is a sphere and that the lattice has no loops. Let $\mathcal{L}_a$ be the state space of particles at some site $a = (s, p)$ (i.e the particles on edges adjacent to s or p with some order that we give below). We will define a left H-module and a left H-comodule structure on $\mathcal{L}_a$ and show those satisfy the left-left Yetter-Drinfeld module compatibility condition.

Analogously to [21] we first define an H-module structure L, for $j \in star(s) \cup bound(p)$ given by L_- from (3.3) if s is the origin of $j \in star(s)$, by L_+ if j is not the origin and $j \in star(s)$ and the trivial H-module otherwise. The H-comodule structure is given by the T_+ action from (3.5) if p is on the right of j, by T_- if it is on the left and by the trivial H-comodule otherwise. In his model, Kitaev only needed to order the edges in $bound(p)$ because the comultiplication of $\mathbb{C}G$ is just the copy map, here we will need some more conventions on the ordering of the edges.

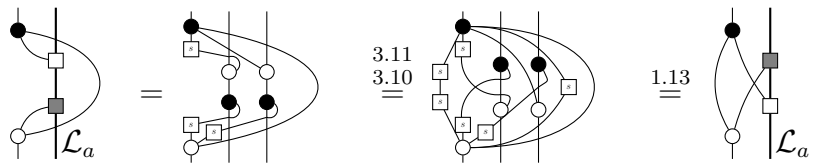
We have $bound(p) = \{j_1, j_2, ..., j_k\}$ starting and ending at vertex s, then order $star(s) = \{i_1, i_2, ..., i_n\}$ where $i_1 = j_1$ and $i_n = j_k$ also note that we have 4 possible configurations of vertex s adjacent to plaquette p and we can choose which of the edges is j_1 and which is j_k. We choose as follows:



Then we can define $\mathcal{L}_a = H_{j_1} \otimes ... \otimes H_{j_{k-1}} \otimes H_{i_2} \otimes ... \otimes H_{i_n}$ where H_m is the copy of H corresponding to edge m. Each of the H_m's carries a left H-module and left H-comodule structure as defined above so that $\mathcal{L}_a$ inherits the tensor product H-module (given by using the comultiplication of H) and tensor product H-comodule structure (using the multiplication). We obtain the following result.

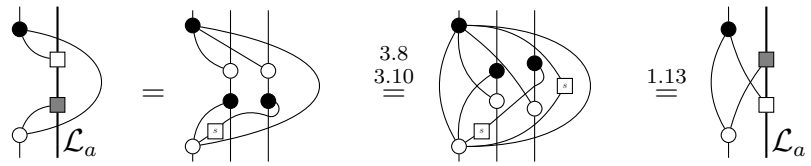
Theorem 3.2. *If the antipode of H is involutive (i.e $S \circ S = id_H$), then $\mathcal{L}_a$ is a left-left Yetter Drinfeld module.*

Proof. We need to check the compatibility condition. First note that for all components of $\mathcal{L}_a$ except the first and last one, the H-action and H-coaction commute (as one of them is trivial). In order to keep our diagrams tidy we will only prove this for the case where $\mathcal{L}_a = H_{j_1} \otimes H_{j_k}$ (i.e $j = n = 2$), but it is easy to generalize the proof as all other components would trivially commute.
For the first configuration we have:



(3.15)

Where the last step also uses the fact that S is an involution. For the second configuration gives:



(3.16)

For a proof of the remaining two cases flip the two proofs above and interchange white with black. □

Note that this does not require the Hopf algebra to be finite dimensional, but the construction on the lattice with H infinite-dimensional does not seem physically meaningful. First of all because it would require having observables with infinitely many degrees of freedom in a lattice configuration. And second because it would produce a theory with infinitely many anyon types, violating one of the assumption of models for TQC. We will discuss this further in relation to permutational quantum computation in section 3.2. For finite-dimensional H, the above result yields a generalisation of Kitaev's original model from groups to Hopf algebras which was already hinted in his original paper [21] and further developed by several authors (see for instance the recent conference at the Perimeter institute for theoretical physics: 'Hopf algebras in Kitaev's quantum double models', Waterloo, Canada, July 31 2017). When $H = \mathbb{C}G$ we recover the DG-module structure defined by Kitaev from the equivalence seen in section 2.3.

3.2 Permutational Quantum Computing

This section is about a model of quantum computation introduced by Jordan [18]. It is a highly restricted model of quantum computation which still seems to yield ad-

vantage over classical computation. For instance, Jordan showed the model allows to compute some irreducible representations of S_n in polynomial time [18], problem for which no classical polynomial time algorithm is known. This section is the beginning of a collaboration with Vojtech Havlicek and consists in a categorical presentation of permutational quantum computing (PQC). The categorical perspective puts PQC on a broader picture and highlights its relationship to TQC but hasn't yielded interesting results yet.

Jordan's model

Here we give an exposition of PQC as it appears in [18] and [17]. Let $\mathcal{L}$ be an n-qubit quantum system. Basis states of an n-qubit quantum systems are often specified by listing eigenvalues of Pauli-Z operators applied to each qubit, which is known as computational basis. Permutational quantum computing (PQC) works with another choice of basis states: eigenstates of complete set of commuting spin measurements on qubit subsets. Let us fix a finite set $I = \{1, 2, 3..., n\}$ indexing the qubits. With a convention that $\hbar = 1$, the spin of the k-th qubit is defined by a triple:

$$\vec{S}_k = \frac{1}{2}\left(X_k,\, Y_k,\, Z_k\right),$$

where X_k, Y_k and Z_k denote the Pauli X,Y and Z operators on the k-th qubit. The total spin operator of a qubit subset A is given by:

$$S_A^2 = \left(\sum_{k\in A} \vec{S}_k\right) \cdot \left(\sum_{k\in A} \vec{S}_k\right),$$

and we will use S^2 to denote the spin operator on the set of all qubits. Let

$$Z_A = \frac{1}{2}\sum_{k\in A} Z_k$$

denote the total Z-spin operator on qubit subset A and we label by Z the total Z-operator applied to all qubits (i.e $Z = Z_I$). Z and S^2 commute and stabilize an eigenspaces labeled by quantum numbers J and M:

$$S^2\left|J,\, M\right\rangle = J(J+1)\left|J,\, M\right\rangle, Z\left|J,\, M\right\rangle = M\left|J,\, M\right\rangle, \tag{3.17}$$

where J is the total spin of all qubits and M takes values $-J \leq M \leq J$ in an integer steps. There are therefore $2J+1$ Z-operator eigenstates for each J and we will refer to this degeneracy as M-degeneracy.

Now, the operators S_A^2 and S_B^2 on sets A, B commute if and only if A and B are disjoint or one is subset of the other. We can then give a complete set of commuting operators on I:

$$S^2_{\{12\}},\ S^2_{\{123\}},\ \dots S^2,\ Z \tag{3.18}$$

In practice, this means that if we have n qubits, measuring each of those operators yields a a sequence of outcomes $j_{12}, j_{123}, ..., J, M$ (the eigenvalues of each operator) which tests for some state of $\mathcal{J}$. Dually, allowing superselection on the outcomes of each measurement we have also defined a preparation recipe. This choice of basis states is known as *sequential coupling*.

The j-quantum numbers on sets of qubits A, B combine according to the angular addition rules [40]:

$$|j_A - j_B| \leq j_{A\cup B} \leq j_A + j_B,$$
$$j_{A\cup B} + j_A + j_B \in \mathbb{Z},$$

For example if $n = 3$, there are two ways to obtain $J = \frac{1}{2}$ eigenstate of three spins - either by adding a qubit to a two-qubit singlet ($J = 0$) state, or by adding a qubit to a triplet ($J = 1$) [33]. We can picture those states as labeled binary trees with n leaves, which we refer to as labeled recoupling diagrams. For instance, for $n = 3$ we have:

Note that the shape of those binary trees is induced by the choices (3.18). Every rooted binary tree shape with n leaves (which we will refer to as recoupling diagram) yields a different choice of complete set of commuting observables, and therefore a different choice of basis for $\mathcal{L}$. And there are 2^n labelled recoupling diagrams for every recoupling diagram, one for each basis state. A computation in PQC is given by the following procedure:

Definition 3.3 (PQC)**.** *Given a permutation π:*

1. *Prepare a simultaneous eigenstate $|\lambda\rangle = |j_{12}, j_{123}, \dots, J, M\rangle$ of $S^2_{12},\ S^2_{123},\ \dots S^2, Z$. Such basis (ie. the sequentially coupled basis) plays the role of computational basis .*

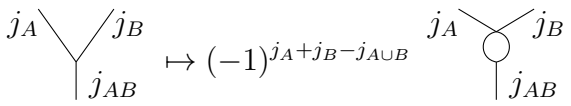
2. *Measure the following set of observables: $S^2_{\pi(1)\pi(2)},\ S^2_{\pi(1)\pi(2)\pi(3)},\ \dots S^2, Z$. This is equivalent to applying a sequence of SWAP gates U_π in the quantum circuit model and measuring a J-spin eigenstate $|x\rangle = |j'_{12}, j'_{123}, \dots, J', M'\rangle$ in the sequentially coupled basis.*

3. *The computing result is obtained by repeating steps 1 and 2 polynomially many times to yield an approximation of the probability distribution $P_\pi(x|\lambda) = |\langle x| U_\pi |\lambda\rangle|^2$.*

In his paper [18], Jordan shows that PQC can approximate the irreducible representations of the symmetric group in polynomial tyme. This is a relatively surprising result as this problem no classical polynomial time algorithm is known that solves the same problem. This hints that although the the PQC model seems trivial in comparison with other quantum computation models it is still superior to classical computation. Any PQC computation (3.3), corresponds to a sequence of phase and racah moves.

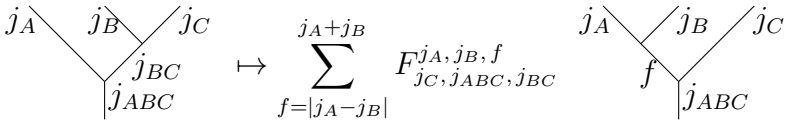
Definition 3.4 (Phase and Racah moves)**.** *Using a convention where $AB := A \cup B$ we define the following*

- *A phase move is obtained by swapping adjacent particles, diagrammatically we picture it as:*



(3.19)

- *Racah moves (or F-moves):*



(3.20)

Where $F^{a,b,c}_{d,e,f}$ is the Wigner 6j-symbol.

Theorem 3.3 (Biedenharn-Louck)**.** *[9, Topic 12] Let A, B, C be disjoint sets of qubits. Any quantum state corresponding to a labelled recoupling diagram can be transformed to a superposition of sequentially coupled labelled recoupling diagram states using a poly(n) sequence of Racah and Phase moves.*

For our purposes the actual evaluation of Wigner $6j$-symbols to compute F-moves won't be important. More importantly, phase and Racah moves have a general categorical description as we will see in the next section.

Categorical PQC

The theory of permutational quantum computing is based on the following abstract ingredients:

1. A tensor product to model many-body quantum systems
2. A direct product to model superpositions of particle types.

3. A set of labels of particle types (with antiparticle for each type) generating all other systems together with fusion rules which account for coupling of those particle types.

4. A permutational structure, i.e the possibility to permute particle positions, i.e phase moves

5. The Racah or F moves which models changes of basis.

6. Underlying Hilbert spaces which account for the quantum mechanical nature of the model.

Let us build a class of categories which account for all those ingredients. As already argued in the previous section we need the structure of a tensor category in order to model many-body quantum systems together with superpositions. We then require the category to contain a simple object for each particle type and to be semisimple so that we obtain fusion rules (see appendix). Note that we do not require there to be finitely many simple objects as in the anyonic case. Indeed note that if we want a theory to reproduce Jordan's model for any chosen number of particles (n), the theory must contain infinitely many particle types, one for each half-integer value (value of angular momentum). We must also require the category to be rigid so that we have antiparticles for each particle type. A tensor category is monoidal so it comes with associators which precisely model the equivalent of the Racah moves. For the permutational structure we require the theory to have a symmetric structure. And finally, if we want to recover finite dimensional Hilbert spaces underlying the objects of our theory we can impose the existence of a forgetful functor to $FHilb \simeq FVect$. Putting it all together we have obtained a rigid semisimple symmetric tensor category $\mathcal{C}$ equipped with a fiber functor $F : \mathcal{C} \to FVect$. We will call those categories Tannakian for our purposes.
The following theorem is a variation of Tannaka reconstruction which shows that any model for permutational quantum computation is induced by a group or a supergroup. Here by supergroup we mean a supercommutative Hopf algebra, i.e a model of $Hopf$ in the category of $\mathbb{Z}_2$-graded vector spaces (see example 2.1) that is cocommutative (i.e (1.25) holds).

Theorem 3.4 (Doplicher-Roberts)**.** *[30, Theorem 2.18] If $\mathcal{C}$ is a rigid semisimple symmetric tensor category equipped with a fiber functor to $Vect$ then $\mathcal{C}$ is symmetrically monoidally equivalent to Rep(G) for G some group (if the twist is trivial) or some supergroup (if the twist is -1).*

Now, we recognize Jordan's model as the theory of representations of the special unitary group.

Proposition 3.5. *Jordan's qubit model $\mathcal{J}_2$ is the category of representations of $SU(2)$.*

Proof. Irreducible representations of $SU(2)$ are precisely indexed by half-integer values and the fusion rules given by angular addition rules [40]. □

We can easily see that defining $\mathcal{J}_d := Rep(SU(d))$ we obtain the corresponding qu*d*it model for permutational quantum computation. The permutational structure of the categories under observation, is tightly linked to the symmetric group S_n. In his model, Jordan builds an algorithm to compute representations of S_n, this can be done in any PQC category.

Proposition 3.6. *Any Tannakian category $\mathcal{C}$ induces representations of the symmetric group S_n for any $n \in \mathbb{N}$.*

Proof. Fix $n \in \mathbb{N}$ and a simple object $a \in obj(\mathcal{C})$ then S_n acts on $a^{\otimes n}$ by permutations, and this clearly defines a module as we can consider a as a vector space using the fiber functor. □

Example 3.3 (S_3 PQC)**.** *The PQC model based on S_3 is rather trivial. Recall the group $S_3 = \{e, g, g^2, \sigma, \sigma g, \sigma g^2\}$. The category $Rep(S_3)$ is a fusion category. By the known representation theory of S_3, $Rep(S_3)$ has three simple objects: the trivial representation 1, the sign representation -1 and the geometric two dimensional representation τ:*

$$\begin{aligned} \tau: \quad \sigma &\mapsto \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \\ g &\mapsto \begin{pmatrix} \omega & 0 \\ 0 & \bar{\omega} \end{pmatrix} \end{aligned}$$

These satisfy the following fusion rules $\forall X$ simple object:

$$\begin{aligned} 1 \otimes X &\simeq X \simeq X \otimes 1 \\ -1 \otimes -1 &\simeq 1 \\ -1 \otimes \tau &\simeq \tau \simeq \tau \otimes -1 \\ \tau \otimes \tau &\simeq 1 \oplus -1 \oplus \tau \end{aligned} \tag{3.21}$$

$Rep(S_3)$ is symmetric so that phase moves are trivial. Note that

$$\tau \otimes \tau \otimes \tau = 1 \oplus -1 \oplus 3\tau$$

So the fusion space $V_{\tau\tau\tau}^{\tau}$ can serve as a qutrit. The only processes allowed by PQC are the swap and the F-matrix. We can choose whether to use the trivial swap or

the one inducing a -1 phase. The F-moves are more interesting and can be shown to yield the equivalent of the Hadamard gate for qutrits adapting the argument in [23]..

Comparing PQC and TQC

In the previous sections we have shown that modular categories are models for TQC whereas PQC is modelled by Tannakian categories. These categorical models have many similarities: they both are examples of semisimple braided tensor categories. At the beginning of the chapter we stated that the PQC model is restricted. Given the categorical formalism this restrictiveness becomes evident. Indeed let $\mathcal{C}_T$ and $\mathcal{C}_P$ be models for TQC and PQC respectively. Then $Z_2(\mathcal{C}_T)$ is trivial, whereas $Z_2(\mathcal{C}_P) = \mathcal{C}_P$. This is because $\mathcal{C}_P$ is symmetric whereas $\mathcal{C}_T$ is modular. Similarly to our discussion in section 2.2, we see that PQC is restricted in comparison to TQC as it exhibits less computational power in the braiding process. This is echoed, at the level of the symmetries (algebra structure), by the passage from groups (or cocommutative Hopf algebras to include the supergroup case) to non-cocommutative Hopf algebras.

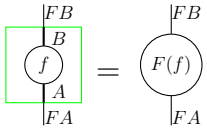
Now, we have seen that Jordan's model corresponds to the interpretation of $Rep(SU(2))$ as a theory of computation. Note that $SU(2)$ is an infinite dimensional Hopf algebra with involutive antipode. In view of the generalization of Kitaev's model in section 3.1, it is a natural question to ask whether we can apply a similar reasoning here, to obtain a model induced by Jordan's model which exhibits topological dependencies. This means applying the Drinfeld center construction to J_2 or in other words pairing the $SU(2)$-modules of Jordan's model with an $SU(2)$ coaction to obtain a theory of $SU(2)$ Yetter-Drinfeld modules. If we want to reproduce Kitaev's construction this could be done by initialising a lattice labeled by representations of $SU(2)$ (i.e spin $1/2$ particles, for instance initialised in sector $1/2$ as in the PQC model) and define similar vertex operators given by the $SU(2)$ action on the tensor and plaquette operators measuring the total angular momentum on the plaquette (yielding the $SU(2)$ coation). The theoretical development and possible implications of this reasoning is left for future work. It is not known to the author whether such model is implementable in practice. It should be noted though that this is very unlikely as it gives rise to a theory of anyons with infinitely many particle types and it is one of the assumptions of TQC that such setting is not possible in nature.

This brings us back to our initial discussion on the restrictiveness of the PQC model. Indeed we have glossed over an important distinction between Tannakian categories and modular categories. Modular categories have only finitely many simple objects whereas Tannakian categories are allowed to have infinitely many (and J_2 is an

example). It is not clear whether the possibility of infinitely many particle types brings computational advantage to PQC.

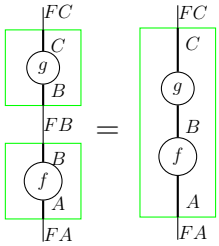
3.3 A braided representation of quantum computation

Recall our discussion on functorial semantics in 1.1. We talked about categories representing syntax (such as $PROs$ and $PROPs$) and semantic categories (such as $Vect$ or $Sets$). The category $Rep(DG)$ is a category which we filled with meaning and we have used it as a semantic category so far. Syntax and semantics are relative notions, in this section we forget all the meaning we associated to the category $Rep(DG)$ (e.g as a theory of anyons, as a model for Kitaev's lattice construction, as a boosting of PQC etc...) and we just see it as a syntax for diagrams which we will interpret in $Vect$. When drawing processes in some monoidal category $\mathcal{D}$, a functorial box [28] is a diagrammatic tool to depict the application of a functor $F : \mathcal{C} \to \mathcal{D}$ to some diagram in the monoidal category $\mathcal{C}$. We can think of the functorial box as a separation between the inside world (the source category $\mathcal{C}$) and the outside world (the target category $\mathcal{D}$). If $f : A \to B$ is a morphism in $\mathcal{C}$ we have:



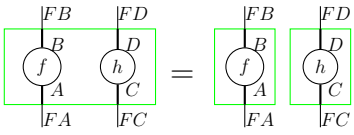
(3.22)

If $g : B \to C$, the defining equation $F(g \circ f) = F(g) \circ F(f)$ is depicted as follows:



(3.23)

If the functor is strict monoidal this means the following holds for $h : C \to D$:

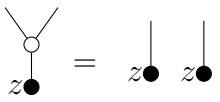


(3.24)

We will use functorial boxes to map the braided pictures in $\mathcal{C} := Rep(DG)$ down to $Vect$ and obtain a braided representation of quantum gates. This means that our ambient world will be the category of vector spaces, but we will borrow pictures from

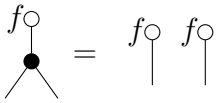
$\mathcal{C}$ using a functorial box. Note that in section 2.3 all our diagrams were drawn in $Vect$ and we slightly abused the diagrammatic notation when equating braids from $Rep(DH)$ to linear maps in $FVect$. We were making implicit use of the forgetful functor $U : Rep(DH) \to FVect$. This was not harmful but here we will be careful to depict U as a green box. First let us fix some definitions.

Definition 3.5 (Copyable states and cocopyable costates)**.** *Let H be a Hopf algebra. A state $z \in H$, also denoted z●, is copyable if:*



(3.25)

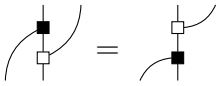
A costate $f : H \to \mathbb{C}$, also denoted f⚲ is cocopyable if



(3.26)

The unit ● is always copyable and the counit ⚲ always cocopyable by definition. In the case of group algebras copyable states are elements of the standard basis and they are in bijective correspondence with elements of the group G. The cocopyable costates are usually called characters. Let $G := \mathbb{Z}_n$ be the cyclic group with n elements (generated by 1), it is a well-known result that there are exactly n characters of G, each determined by sending 1 to one of the n^{th} roots of unity. In what follows we will denote by ${}^{\alpha}\!\!\Qweierp$ the character of $\mathbb{Z}_n$ obtained by sending 1 to $e^{\frac{i2\pi\alpha}{n}}$.

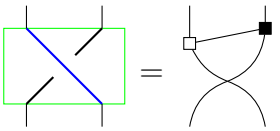
We know from section 2.3 that $\mathcal{C} = Rep(DG) \simeq \mathcal{D}_G^{lr}$, so specifying an object of $Rep(DG)$ just corresponds to choosing a vector space V with a left G-module structure and right G-comodule structure satisfying the compatibility conditions.
To illustrate the idea let us start by picking $G := \mathbb{C}\mathbb{Z}_2$ and let us denote the standard basis of $\mathbb{C}\mathbb{Z}_2$ by $\{|0\rangle, |1\rangle\}$. Fix an object a left-right Yetter-Drinfeld module V over G (equivalently $V \in obj(Rep(DG))$) with black G-action and white G-coaction. As $\mathbb{C}\mathbb{Z}_2$ is both commutative and cocommutative it is easy to see (using the antipode) that the left-right Yetter-Drinfeld compatibility condition is equivalent to the following:



(3.27)

In order to represent quantum computation, we first should be able to reproduce a CNOT gate. The CNOT quantum gate arises from the interaction of complementary observables, such as X and Z observables (see [10, Chapter 9]). It is then natural to consider the Z G-action (given by $0 \mapsto id$ and $1 \mapsto Z$ the Pauli Z operator) and X

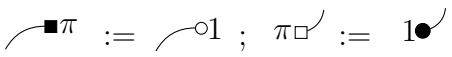
G action (defined similarly with the Pauli X operator) on $\mathbb{C}^2$. It is easy to see that these yield G-module structures on $\mathbb{C}^2$ and as objects in $FVect$ are self-dual they can turned into G-coactions which yield G-comodule structures. Note that X and Z do not commute, so the combination of a Z action (coaction) and an X coaction (action) doesn't satisfy (3.27). For this reason we will need to choose two two-dimensional objects of $\mathcal{C}$ representing qubits equipped with Z and X complementary observables. Let V_Z be $\mathbb{C}^2$ equipped with the Z-action and Z-coaction and V_X be $\mathbb{C}^2$ equipped with X-action and X-coaction. These trivially satisfy (3.27). We will denote V_Z with black wires (and Z (co)action with black boxes) and V_X with blue wires (and X (co)action with white boxes) in $\mathcal{C}$. Recall that the forgetfull functor $U : \mathcal{C} \to FVect$ picks the underlying vector space of each object and the underlying linear map each morphism. So $U(V_X) = U(V_Z) = \mathbb{C}^2$ that we will denote with thin black wires. Also it can be checked that U is a strict monoidal functor, we will depict it as a green box. Now from (2.27) we obtain the following equality:



(3.28)

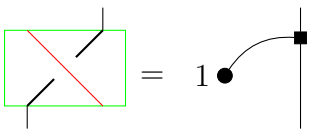
And by definition the two equivalent diagrams above correspond to a SWAP gate followed by a CNOT gate.

We now want a way to represent phases, we will do this using ancillary systems which will implement a phase when braiding around V_X and V_Z. Define π as the one-dimensional representation with action and coaction given by:



(3.29)

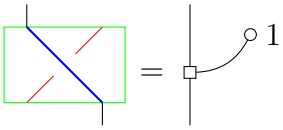
This is are well defined $\mathbb{C}\mathbb{Z}_2$-(co)module structures as $1\bullet = |1\rangle$ is copyable and $1\circ = \langle 0| - \langle 1|$ is cocopyable. The Yetter-Drinfeld compatibility condition is trivially satisfied. In $\mathcal{C}$, π is not the unit of the tensor and we will denote it by a red wire. Then braiding V_Z with π gives:



(3.30)

which, by definition of the action on V_Z, is the Pauli Z operator applied to V_Z, i.e the Z π phase. Note that the blue wire denoting π 'disappears' outside the box as it is mapped to the identity on the tensor unit in $FVect$. Similarly braiding π with

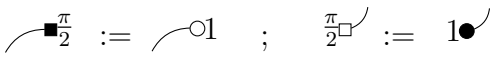
V_X gives:



(3.31)

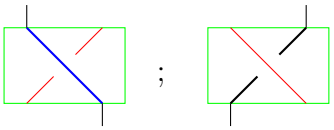
which precisely yields an X π phase. Now π phases are not enough to obtain interesting computations. For instance we would wish to perform a $\pi/2$ phase in order to get stabilizer quantum mechanics. For this reason we will need to choose a bigger group and again consider functorial semantics onto $FVect$, but the ideas to describe gates as braids will be the same as above.

Pick the group $\mathbb{Z}_4 = \{0, 1, 2, 3\}$ and consider $Rep(D\mathbb{Z}_4)$. First of all note that any $\mathbb{Z}_2$-module is also a $\mathbb{Z}_4$-module by precomposing with the parity homomorphism $p : \mathbb{Z}_4 \to \mathbb{Z}_2$. Similarly any $\mathbb{Z}_2$-comodule is a $\mathbb{Z}_4$-comodule. Therefore there is a copy of $Rep(D\mathbb{Z}_2)$ living inside $Rep(D\mathbb{Z}_4)$. Again we pick Q_Z and Q_X to be the two-dimensional modules with G action generated by the $\frac{\pi}{2}$ Z-phase and $\frac{\pi}{2}$ X-phase respectively. Then the braid on $Q_Z \otimes Q_X$ is a SWAP gate followed by a CNOT as above. Now consider the one-dimensional object $\frac{\pi}{2}$ of $Rep(D\mathbb{Z}_4)$ defined by the following action and coaction:



(3.32)

Using red wires to denote $\frac{\pi}{2}$, we have that the braids:



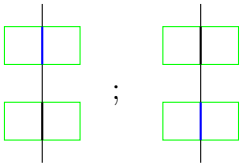
;

(3.33)

are respectively a $\frac{\pi}{2}$ X-phase and a $\frac{\pi}{2}$ Z-phase.

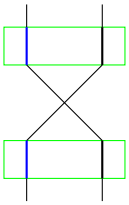
By now we have formed a language internal to the $\mathcal{C}$ consisting of all possible braids using black, red and blue wires. This language is closed under composition and tensoring so it forms a monoidal subcategory $\mathcal{L}$ of $\mathcal{C}$. Applying the green functor U to any diagram in $\mathcal{L}$ interprets it in the context of quantum computation by associating to it a quantum gate on qubits. In $\mathcal{L}$ we can describe various gates such as CNOT$\circ$SWAP but it is clearly not universal for stabilizer quantum computation. Indeed it is not possible to express the composition of X and Z phases on the same system while staying in the inner world of $\mathcal{C}$. In fact the main limitation is that we cannot turn a black wire into a blue one inside $\mathcal{C}$. The advantage of using functorial boxes is that we can work in both source and target categories at the same time. We will now add ingredients to our language, using the outside world of $FVect$ to obtain a representation of the stabilizer segment of quantum computation. We will

first need the following diagrams:



(3.34)

i.e we need to be able to get outside the box and look at our system from the complementary perspective. We can change the colour of wires by going outside the boxes and therefore we can now compose X and Z phases. It is well-known that X and Z $\pi/2$-phases and their compositions yield any one-qubit stabilizer transformation. In order to capture all multi-qubit quantum gates we need the possibility of swapping two systems without making them interact. As allowed by the following diagram:

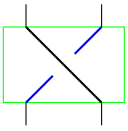


(3.35)

Note that the order of the colours is insignificant here, as we can pre/post compose with the diagrams in (3.34) to change colour.

Proposition 3.7. *Any stabilizer quantum gate can be expressed as a composition (or tensoring) of boxed braids from $\mathcal{L}$ together with the diagrams in (3.34) and (3.35).*

Proof. This is immediate from the fact that CNOT and $\pi/2$ phases form a universal gate set for stabilizer quantum computation. Phases were discussed above and the CNOT gate is simply obtained by composing (3.35) with the braid:



□

A quantum computation also requires the preparation of states and measurements. As we can already express any stabilizer quantum gate, we only need separable states and costates. Any choice of one Z or X standard one-qubit basis state and costate suffices: all other (co)states are achieved by tensoring and applying gates. This description of states is analogous to the one in measurement-based quantum computing [10, Section 12.3] and an the syntax described here could be adapted to this model of quantum computation.

Chapter 4

Conclusion and Future Works

The main outcome of this work was to demonstrate the power of diagrammatic reasoning, using it as the main tool in casting the theory of representations of Hopf algebras and their applications to quantum computation.
In the first chapter we have seen how diagrammatic languages arise from monoidal categories and have used them to study Hopf algebras and their representations. The content of this chapter is well-known [26], but the diagrammatic proofs are not commonly found in the literature.
In sections 2.1 and 2.2 we have developed the categorical framework of Modular categories (MCs) for describing theories of anyons following similar formulations provided in [22, 36, 32, 29, 5]. This has endowed the diagrammatic notation with physical significance as drawing diagrams in Modular categories corresponds to drawing anyon trajectories in $2+1$ dimensional space-time. Quasitriangular Hopf algebras capture the symmetries of certain topological quantum systems and give rise to most BFCs as categories of representations. Starting with a finite dimensional Hopf algebra H the quantum double construction produces a quasitriangular Hopf algebra DH. At the categorical level this quantization procedure is captured by the Drinfeld center construction, which we have studied in detail in section 2.3. The results of this section are known but the diagrammatic proofs are not present in the literature.
In chapter 3 we have started by recalling Kitaev's models for Topological Quantum Computation (TQC). In [21], Kitaev constructs a Hamiltonian capturing the dynamics of the lattice system exhibiting anyonic behaviour. Theorem 3.2 is our own contribution, giving a diagrammatic generalization of the reasoning in [21] to the Hopf algebra framework. A categorical formulation of the Hamiltonian formalism would provide an interesting follow up and is material for future works. We have then introduced Jordan's model for Permutational Quantum Computing (PQC) [18] and contributed with a categorical perspective on the model. This hasn't been very fruitful yet but has opened many directions for future research. Indeed, discussing ways of relating PQC to TQC has raised various questions: can we boost Jordan's

model for PQC to obtain a model with topological dependencies between particles? Is it true that only finitely many anyon types are possible in nature? If we interpret the category of Yetter-Drinfeld modules over $SU(2)$ as a theory of computation, how powerful is it?

Finally, we have explored functorial semantics in the context of quantum computation, using the diagrams in Modular categories to construct a syntax for stabilizer quantum gates. The use of functorial boxes allowed us to formulate a diagrammatic language on two levels: the underlying commutative world of vector spaces and the braided world of Yetter-Drinfeld modules. The study of this syntax from a type-theoretic perspective could result in a (non-commutative) programming language for measurement-based (stabilizer) quantum computing ; although it is likely to incur in complexity issues.

Bibliography

[1] S. Abramsky and B. Coecke. A categorical semantics of quantum protocols. *Proceedings of the 19th IEEE conference on Logic in Computer Science*, 2004.

[2] S. Abramsky and N. Tzevelekos. *Introduction to Categories and Categorical Logic*, volume 813 of *New Structures for Physics, Lecture Notes in Physics*. Springer-Verlag Berlin Heidelberg, 2011.

[3] L. Auslander. An account of the theory of crystallographic groups. In *Proceedings of the American Mathematical Society*, pages 1230–1236, 1965.

[4] J. Baez and D. James. Categorification. *eprint arXiv:math/9802029*, 1998.

[5] B. Bakalov and A. Kirillov. *Lectures on tensor categories and modular functors*, volume 21 of *University Lectures Series*. American Mathematical Society, 2000.

[6] B. Balsam and A. Kirillov. Kitaev's lattice model and turaev-viro tqfts. *eprint arXiv:1206.2308*, 2012.

[7] B. Bartlett. Categorical aspects of topological quantum field theories. Master's thesis, Utrecht University, 2005.

[8] B. Bartlett. Fusion categories via string diagrams. *eprint arXiv:1502.02882*, 2015.

[9] L. C. Biedenharn and J. D. Louck. *The Racah-Wigner Algebra in Quantum Theory*, volume 9 of *Encyclopedia of Mathematics and its Applications*. Addison Wesley Publ. Co., 1981.

[10] B. Coecke and A. Kissinger. *Picturing Quantum Processes*. Cambridge University Press, 2017.

[11] V. Drinfeld. *Quantum groups*. American Mathematical Society, 1987.

[12] P. Etingof, S. Gelaki, D. Nikshych, and V. Ostrik. *Tensor Categories*, volume 205 of *Mathematical Surveys and Monographs*. American Mathematical Society, 2015.

[13] P. Etingof, D. Nikshych, and V. Ostrik. On fusion categories. *eprint arXiv:math/0203060*, 2002.

[14] P. Freyd. *Abelian Categories*. Harper & Row, 1966.

[15] Elizabeth Gibney. Inside microsoft's quest for a topological quantum computer.

[16] M. Gould. Quantum double finite group algebras and their representations. *Bulletin of the Australian Mathematical Society*, 1993.

[17] Vojtech Havlicek. Search for computational advantage in permutational quantum computing. unpublished (Term paper), April 2017.

[18] S. P. Jordan. Permutational quantum computing. *Arxiv e-prints*, 2009.

[19] A. Joyal and R. Street. An introduction to tannaka duality and quantum groups. In *Category Theory*, volume 1488 of *Lecture Notes in Mathematics*, pages 413–492. Springer, Berlin, 1991.

[20] G. M. Kelly. Many variable functorial calculus. In *Coherence in Categories*, Lecture Notes in Mathematics, pages 66–105. Springer, 1972.

[21] A. Kitaev. Fault-tolerant quantum computation by anyons. *Annals Phys. 303*, pages 2–30, 2003.

[22] A. Kitaev. Anyons in an exactly solved model and beyond. *Annals Phys. 30*, pages 2–111, 2006.

[23] Ville Lahtinen. Topological quantum computation: an analysis of an anyon model based on quantum double symmetries. Master's thesis, University of Helsinki, 2006.

[24] F: W. Lawvere. *Functorial semantics of algebraic theories*. PhD thesis, Columbia University, 1963.

[25] S. Mac Lane. *Categories for the working mathematician*. Springer Verlag, 1971.

[26] Sh. Majid. *Foundations of Quantum Group Theory*. Cambridge University Press, 1995.

[27] Sh. Majid. Quantum groups and braided algebra. In *Quantum Physics and Linguistics: A Compositional, Diagrammatic Discourse*. Oxford Scholarship Online, 2013.

[28] P. Mellies. Functorial boxes in string diagrams. In Springer Verlag, editor, *Computer Science Logic*, pages 1–30, 2006.

[29] M. Mueger. On the structure of modular categories. *eprint arXiv:math/0201017*, 2002.

[30] M. Mueger. Abstract duality theory for symmetric tensor *-categories. In *Algebraic Quantum field Theory*, Handbook of the Philosophy of Physics, chapter Appendix. J. Butterfield & J. Earman (eds), 2006.

[31] M. Mueger. Modular categories. http://www.math.ru.nl/ mueger/PDF/oxford.pdf, 2012.

[32] P. Panangaden and E. Paquette. *A categorical presentation of quantum computation with anyons*, volume 813 of *Lecture Notes in Physics*. Springer Berlin Heidelberg, 2011.

[33] R. Pauncz. *Alternant Molecular Orbital Method*. Studies in physics and chemistry, no. 4. Saunders, 1967.

[34] E. Rowell and W. Zhenghan. Mathematics of topological quantum computing. *eprint arXiv:1705.06206*, 2017.

[35] P. Selinger. A survey of graphical languages for monoidal categories. *eprint arXiv:0908.3347*, 2009.

[36] S. Simon. Toplogical quantum. http://oxfordtopquantum.tiddlyspot.com/, 2016.

[37] J. K. Slingerland. *Hopf symmetry and its breaking: braid statistics and confinement in planar physics*. PhD thesis, Universiteit van Amsterdam, 2002.

[38] J. Vercruysse. Hopf algebras, variant notions and reconstruction theorems. *eprint arXiv:1202.3613*, 2012.

[39] J. Vicary and C. Heunen. Lectures on categorical quantum mechanics, 2012. https://www.cs.ox.ac.uk/files/4551/cqm-notes.pdf.

[40] Peter Woit. *Quantum Theory, Groups and Representations: An Introduction (Final draft version)*. 2017.

Appendix A

Abelian categories

Abelian categories are frameworks for theories that have similar behaviour to linear algebra. Results and definitions related to the theme of this appendix can be found in [12], [8] and [14]. We will give a short description of abelian categories inspired by [32], for this we will need a few preliminary notions. We start by generalising the notions of injection and surjection from set theory.

Definition A.1 (Mono)**.** *A morphism $f : a \to b$ is a monomorphism if for any $g, h : c \to a$*

$$f \circ g = f \circ h \implies h = g$$

Definition A.2 (Epi)**.** *A morphism $f : a \to b$ is an epimorphism if for any $g, h : b \to c$*

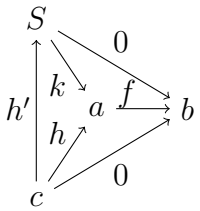
$$g \circ f = h \circ f \implies h = g$$

One important feature of linear algebra is the presence of the following object.

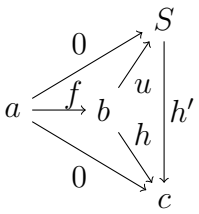
Definition A.3 (Zero object)**.** *We say $0 \in obj(\mathcal{C})$ is a zero object if it is both initial and terminal, i.e for any object a in $\mathcal{C}$ there are unique arrows $0 \to a$ and $a \to 0$.*

Such object gives rise to the presence of zero morphisms in any hom-set as the unique morphism $a \to 0 \to b$ for any $a, b \in obj(\mathcal{C})$. Note that *Sets* has no zero object. We obtain generalizations of the notions of kernels from vector spaces.

Definition A.4 (Kernel and cokernel)**.** *The kernel of a morphism $f : a \to b$ in a category with zero object 0 is a morphism $k : S \to a$ such that $f \circ k = 0$ and for any $h : c \to a$ such that $h \circ f = 0$ there is a unique $h' : c \to S$ such that the following diagram commutes:*



Dually the cokernel of a morphism $f : a \to b$ is a morphism $u : b \to S$ making the following commute:



Definition A.5 (Ab category)**.** *The category $\mathcal{C}$ is Ab if it is enriched over abelian groups. That is all hom-sets have abelian group structures and composition of morphisms is a group homomorphism.*

Definition A.6 (Direct sums)**.** *A category $\mathcal{C}$ has direct sums if it has a monoidal structure with tensor $\oplus$ and such that $\oplus$ is the categorical product and coproduct. This means for any objects $A, B \in obj(\mathcal{C})$ the direct product $A \oplus B$ comes with projections p_A, p_B and injections i_A, i_B satisfying the universal properties of the categorical product and coproduct (respectively).*

For $\oplus$ to be the categorical product means that for any morphisms $f : C \to A$ $g : C \to B$ there is a unique arrow $h : C \to A \oplus B$ such that $p_A \circ h = f$ and $p_B \circ h = g$. The universal property of the coproduct is the dual notion where all arrows are flipped and projections are replaced by injections.

Definition A.7 (Additive category)**.** *An Ab-category $\mathcal{C}$ is additive if it has zero object and every pair of objects has a direct sum $\oplus$.*

It can be shown that the zero morphism of any hom-set is then the unit of the abelian group structure [39]. We can now define abelian categories.

Definition A.8 (Abelian category)**.** *An abelian category is an additive category where every morphism has a kernel and a cokernel and every monic (epic) is a kernel (cokernel).*

Note that this is precisely the behaviour of kernels and cokernels from vector spaces. In an abelian category hom-sets are abelian groups, but we can explicitly require hom-sets to be vector spaces over some field k.

Definition A.9 (k-linearity)**.** *Let k be field, we say $\mathcal{C}$ is k-linear if all hom-sets are k-vector spaces and composition is bilinear.*

We will assume throughout the thesis that $k = \mathbb{C}$ so in particular the field is algebraically closed.
As explained in the introduction, categorification is the process of replacing sets by

categories, functions by functors and weakening equalities to natural isomorphisms. Tensor categories are the categorification of rings. Multiplication becomes a tensor product $\otimes$ and addition becomes a direct product $\oplus$. Monoidal categories are defined in the first chapter of the thesis. They are obtained by categorifying the notion of a monoid. We also defined rigidity, as the property that any object has a right dual and a left dual. The notion of tensor categories is obtained by considering rigid monoidal structures on abelian categories.

Definition A.10 (Tensor category)**.** *A tensor category is an abelian rigid monoidal category.*

Fusion categories are special types of tensor categories where the objects are generated under $\oplus$ by a finite set of simple objects.

Definition A.11 (Simple object)**.** *An object X in a $\mathbb{C}$-linear category is called simple if $EndX = kid_X$.*

Definition A.12 (Semisimplicity)**.** *$\mathcal{C}$ is semisimple if every object is isomorphic to a direct sum of simple objects. $\mathcal{C}$ is finite if there are finitely many isomorphism classes of simple objects.*

Definition A.13 (Fusion category)**.** *A $\mathbb{C}$-linear tensor category is a fusion category if it has finite-dimensional hom-spaces, is semisimple with finitely many isomorphism classes of simple objects and the unit $\mathbf{1}$ is simple.*

The following is a folklore result, for which we refer to [13].

Theorem A.1. *If H is a finite dimensional semisimple Hopf algebra then $Rep(H)$ is a fusion category.*